
Linux Standard Base (LSB) —
Part 1-3:
Desktop specification generic part

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021



STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by the Linux Foundation as Linux Standard Base (LSB): Desktop specification generic part and drafted in accordance with its editorial rules. It was assigned to Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 22, *Programming languages, their environments and system software interfaces*, and adopted by National Bodies.

This first edition of ISO/IEC 23360-1-3 cancels and replaces ISO/IEC 23360-1:2006, which has been technically revised.

This document is based on “The GNU Free Documentation License, version 1.1”. The license is available at <https://www.gnu.org/licenses/old-licenses/fdl-1.1.html>.

A list of all parts in the ISO/IEC 23660 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Contents

Foreword	iii
Introduction	viii
I Introductory Elements	1
1 Scope	2
2 References	3
2.1 Normative References	3
2.2 Informative References/Bibliography	6
3 Requirements	7
3.1 Relevant Libraries	7
4 Terms and Definitions	9
5 Documentation Conventions	11
II Graphic Libraries	12
6 Libraries	13
6.1 Interfaces for libX11	13
6.2 Data Definitions for libX11	23
6.3 Interface Definitions for libX11	180
6.4 Interfaces for libSM	180
6.5 Data Definitions for libSM	181
6.6 Interfaces for libICE	185
6.7 Data Definitions for libICE	187
6.8 Interface Definitions for libICE	194
6.9 Interfaces for libXt	196
6.10 Data Definitions for libXt	201
6.11 Interface Definitions for libXt	235
6.12 Interfaces for libXext	236
6.13 Data Definitions for libXext	239
6.14 Interface Definitions for libXext	257
6.15 Interfaces for libXi	257
6.16 Data Definitions for libXi	258
6.17 Interfaces for libXtst	288
6.18 Data Definitions for libXtst	289
6.19 Interfaces for libxcb	290
6.20 Data Definitions for libxcb	298
6.21 Interface Definitions for libxcb	373
III OpenGL Libraries	374
7 Libraries	375
7.1 Interfaces for libGL	375
7.2 Data Definitions for libGL	384
7.3 Interfaces for libGLU	438
7.4 Data Definitions for libGLU	439
IV PNG12 library	445
8 Libraries	446
8.1 Interfaces for libpng12	446
8.2 Data Definitions for libpng12	448
8.3 Interface Definitions for libpng12	462
V JPEG library	527
9 Libraries	528
9.1 Interfaces for libjpeg	528

9.2 Data Definitions for libjpeg	529
9.3 Interface Definitions for libjpeg	539
VI Fontconfig library	561
10 Libraries.....	562
10.1 Interfaces for libfontconfig.....	562
10.2 Data Definitions for libfontconfig	565
11 Commands and Utilities.....	574
11.1 Commands and Utilities.....	574
11.2 Command Behavior	574
VII Freetype library	577
12 Libraries.....	578
12.1 Interfaces for libfreetype.....	578
12.2 Data Definitions for libfreetype.....	580
12.3 Interface Definitions for libfreetype.....	610
VIII Xft library	612
13 Libraries.....	613
13.1 Interfaces for libXft	613
13.2 Data Definitions for libXft	614
13.3 Interface Definitions for libXft	619
IX Xrender library.....	620
14 Libraries.....	621
14.1 Interfaces for libXrender	621
14.2 Data Definitions for libXrender	622
14.3 Interface Definitions for libXrender	629
X Cairo Vector Graphics library	630
15 Libraries.....	631
15.1 Interfaces for libcairo.....	631
15.2 Data Definitions for libcairo.....	638
15.3 Interface Definitions for libcairo.....	665
XI tiff library	666
16 Libraries.....	667
16.1 Interfaces for libtiff.....	667
16.2 Data Definitions for libtiff.....	669
XII GTK+ Stack Libraries	683
17 Libraries.....	684
17.1 Introduction	684
17.2 Interfaces for libglib-2.0	685
17.3 Data Definitions for libglib-2.0	726
17.4 Interface Definitions for libglib-2.0	795
17.5 Interfaces for libgmodule-2.0.....	796
17.6 Data Definitions for libgmodule-2.0.....	796
17.7 Interfaces for libgobject-2.0	797
17.8 Data Definitions for libgobject-2.0.....	810
17.9 Interface Definitions for libgobject-2.0.....	848
17.10 Interfaces for libgthread-2.0	849
17.11 Interfaces for libgio-2.0	850
17.12 Data Definitions for libgio-2.0	887
17.13 Interface Definitions for libgio-2.0	1048
17.14 Interfaces for libatk-1.0	1048
17.15 Data Definitions for libatk-1.0.....	1054

17.16	Interface Definitions for libatk-1.0.....	1079
17.17	Interfaces for libpango-1.0.....	1079
17.18	Data Definitions for libpango-1.0.....	1088
17.19	Interfaces for libpangocairo-1.0.....	1111
17.20	Data Definitions for libpangocairo-1.0.....	1113
17.21	Interfaces for libpangoft2-1.0.....	1114
17.22	Data Definitions for libpangoft2-1.0.....	1115
17.23	Interfaces for libpangoft2-1.0.....	1117
17.24	Data Definitions for libpangoft2-1.0.....	1119
17.25	Interfaces for libgdk_pixbuf-2.0.....	1121
17.26	Data Definitions for libgdk_pixbuf-2.0.....	1125
17.27	Interfaces for libgdk_pixbuf_xlib-2.0.....	1133
17.28	Data Definitions for libgdk_pixbuf_xlib-2.0.....	1134
17.29	Interfaces for libgdk-x11-2.0.....	1136
17.30	Data Definitions for libgdk-x11-2.0.....	1149
17.31	Interfaces for libgtk-x11-2.0.....	1210
17.32	Data Definitions for libgtk-x11-2.0.....	1272
17.33	Interface Definitions for libgtk-x11-2.0.....	1479
XIII	Qt Libraries.....	1482
18	Libraries.....	1483
18.1	Introduction.....	1483
18.2	Interfaces for libQtCore.....	1484
18.3	Data Definitions for libQtCore.....	1578
18.4	Interface Definitions for libQtCore.....	1620
18.5	Interfaces for libQtGui.....	1628
18.6	Data Definitions for libQtGui.....	2163
18.7	Interface Definitions for libQtGui.....	2209
18.8	Interfaces for libQtXml.....	2267
18.9	Data Definitions for libQtXml.....	2288
18.10	Interfaces for libQtOpenGL.....	2289
18.11	Data Definitions for libQtOpenGL.....	2300
18.12	Interface Definitions for libQtOpenGL.....	2301
18.13	Interfaces for libQtSql.....	2302
18.14	Data Definitions for libQtSql.....	2325
18.15	Interface Definitions for libQtSql.....	2327
18.16	Interfaces for libQtSvg.....	2328
18.17	Data Definitions for libQtSvg.....	2334
18.18	Interfaces for libQtNetwork.....	2335
18.19	Data Definitions for libQtNetwork.....	2356
18.20	Interface Definitions for libQtNetwork.....	2360
XIV	ALSA sound library.....	2362
19	Libraries.....	2363
19.1	Interfaces for libasound.....	2363
19.2	Data Definitions for libasound.....	2386
XV	Desktop Environment.....	2429
20	Desktop Environment.....	2430
20.1	Desktop Base Directory.....	2430
20.2	Desktop Entries.....	2430
20.3	Desktop Menu Specification.....	2430
20.4	Icon Theme Specification.....	2430

21 Desktop Commands.....	2431
21.1 Xdg-utils.....	2431
XVI Package Format and Installation	2432
22 Software Installation.....	2433
22.1 Package Dependencies	2433
Annex A Alphabetical Listing of Interfaces by Library.....	2434
A.1 libGL.....	2434
A.2 libGLU.....	2444
A.3 libICE	2445
A.4 libSM	2446
A.5 libX11	2447
A.6 libXext	2458
A.7 libXft	2460
A.8 libXi.....	2461
A.9 libXrender	2462
A.10 libXt.....	2463
A.11 libXtst.....	2468
A.12 libcairo	2468
A.13 libcairo-gobject	2475
A.14 libcairo-script-interpreter.....	2476
A.15 libfontconfig.....	2476
A.16 libfreetype	2479
A.17 libjpeg.....	2481
A.18 libpng12	2482
A.19 libtiff.....	2485
A.20 libxcb.....	2487
A.21 libatk-1.0	2496
A.22 libgdk-x11-2.0	2500
A.23 libgdk_pixbuf-2.0.....	2510
A.24 libgdk_pixbuf_xlib-2.0	2513
A.25 libgio-2.0	2514
A.26 libglib-2.0.....	2546
A.27 libgmodule-2.0	2570
A.28 libgobject-2.0.....	2570
A.29 libgthread-2.0	2578
A.30 libgtk-x11-2.0	2578
A.31 libpango-1.0.....	2628
A.32 libpangocairo-1.0	2635
A.33 libpangoft2-1.0	2636
A.34 libpangoxft-1.0	2637
A.35 libQtCore	2637
A.36 libQtGui.....	2687
A.37 libQtNetwork.....	2887
A.38 libQtOpenGL	2898
A.39 libQtSql	2904
A.40 libQtSvg.....	2913
A.41 libQtXml.....	2915
A.42 libasound	2925

Introduction

The LSB defines a binary interface for application programs that are compiled and packaged for LSB-conforming implementations on many different hardware architectures. A binary specification must include information specific to the computer processor architecture for which it is intended. To avoid the complexity of conditional descriptions, the specification has instead been divided into generic parts which are augmented by one of several architecture-specific parts, depending on the target processor architecture; the generic part will indicate when reference must be made to the architecture part, and vice versa.

This document should be used in conjunction with the documents it references. This document enumerates the system components it includes, but descriptions of those components may be included entirely or partly in this document, partly in other documents, or entirely in other reference documents. For example, the section that describes system service routines includes a list of the system routines supported in this interface, formal declarations of the data structures they use that are visible to applications, and a pointer to the underlying referenced specification for information about the syntax and semantics of each call. Only those routines not described in standards referenced by this document, or extensions to those standards, are described in the detail. Information referenced in this way is as much a part of this document as is the information explicitly included here.

The specification carries a version number of either the form $x.y$ or $x.y.z$. This version number carries the following meaning:

1. The first number (x) is the major version number. Versions sharing the same major version number shall be compatible in a backwards direction; that is, a newer version shall be compatible with an older version. Any deletion of a library results in a new major version number. Interfaces marked as deprecated may be removed from the specification at a major version change.
2. The second number (y) is the minor version number. Libraries and individual interfaces may be added, but not removed. Interfaces may be marked as deprecated at a minor version change. Other minor changes may be permitted at the discretion of the LSB workgroup.
3. The third number (z), if present, is the editorial level. Only editorial changes should be included in such versions.

Since this specification is a descriptive Application Binary Interface, and not a source level API specification, it is not possible to make a guarantee of 100% backward compatibility between major releases. However, it is the intent that those parts of the binary interface that are visible in the source level API will remain backward compatible from version to version, except where a feature marked as "Deprecated" in one release may be removed from a future release. Implementors are strongly encouraged to make use of symbol versioning to permit simultaneous support of applications conforming to different releases of this specification.

LSB is a trademark of the Linux Foundation. Developers of applications or implementations interested in using the trademark should see the Linux Foundation Certification Policy for details.

I Introductory Elements

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

1 Scope

The Linux Standard Base (LSB) defines a system interface for compiled applications and a minimal environment for support of installation scripts. Its purpose is to enable a uniform industry standard environment for high-volume applications conforming to the LSB.

These specifications are composed of two basic parts: a common part describing those parts of the interface that remain constant across all implementations of the LSB, and an architecture-specific part describing the parts of the interface that vary by processor architecture. Together, the common part and the relevant architecture-specific part for a single hardware architecture provide a complete interface specification for compiled application programs on systems that share a common hardware architecture.

The LSB contains both a set of Application Program Interfaces (APIs) and Application Binary Interfaces (ABIs). APIs may appear in the source code of portable applications, while the compiled binary of that application may use the larger set of ABIs. A conforming implementation provides all of the ABIs listed here. The compilation system may replace (e.g. by macro definition) certain APIs with calls to one or more of the underlying binary interfaces, and may insert calls to binary interfaces as needed.

The LSB is primarily a binary interface definition. Not all of the source level APIs available to applications may be contained in this specification.

This is the common part of the Desktop module of the Linux Standard Base (LSB). This module provides the fundamental system interfaces, libraries, and runtime environment upon which all conforming applications and libraries depend requiring the LSB Desktop module depend.

The common part of LSB Desktop should be used in conjunction with an architecture-specific part. Whenever a section of the common part is supplemented by architecture specific information, the common part includes a reference to the architecture-specific part. Architecture-specific parts of LSB Desktop may also contain additional information that is not referenced in the common part.

Interfaces described in this part of LSB Desktop are mandatory except where explicitly listed otherwise. Interfaces described in the LSB Desktop module supplement those described in the LSB Core module. They do not depend on other LSB modules.

2 References

2.1 Normative References

The specifications listed below are referenced in whole or in part by the LSB Desktop specification. Such references may be normative or informative; a reference to specification shall only be considered normative if it is explicitly cited as such. The LSB Desktop specification may make normative references to a portion of these specifications (that is, to define a specific function or group of functions); in such cases, only the explicitly referenced portion of the specification is to be considered normative.

Table 2-1 Normative References

Name	Title	URL
ATK 2.2.0 Reference Manual	ATK 2.2.0 Reference Manual	http://developer.gnome.org/atk/2.2/index.html
Double Buffer Extension Library	Double Buffer Extension Library - Protocol Version 1.0	http://refspecs.linuxfoundation.org/X11/dbelb.pdf
Fontconfig Developers Reference	Fontconfig Developers Reference, Version 2.6.0	http://refspecs.linuxfoundation.org/fontconfig-2.6.0
Gdk 2.10.14 Reference Manual	Gdk 2.10.14 Reference Manual	http://library.gnome.org/devel/gdk/2.10/
Gdk-pixbuf 2.26.0 Reference Manual	Gdk-pixbuf 2.26.0 Reference Manual	http://developer.gnome.org/gdk-pixbuf/2.26
Gio 2.32 Reference Manual	Gio 2.32 Reference Manual	http://developer.gnome.org/gio/2.32
Glib 2.32 Reference Manual	Glib 2.32 Reference Manual	http://developer.gnome.org/glib/2.32
Gobject 2.32 Reference Manual	Gobject 2.32 Reference Manual	http://developer.gnome.org/gobject/2.32
Gtk+ 2.10.14 Reference Manual	Gtk+ 2.10.14 Reference Manual	http://library.gnome.org/devel/gtk/2.10/
ISO C (1999)	ISO/IEC 9899:1999 - Programming Languages -- C	
ISO/IEC 14882: 2003 C++ Language	ISO/IEC 14882: 2003 Programming languages --C++	
Itanium™ C++ ABI	Itanium™ C++ ABI (Revision 1.86)	http://refspecs.linuxfoundation.org/cxxabi-1.86.html
Libtiff 4.0.2 Reference Manual	Libtiff 4.0.2 Reference Manual	http://www.libtiff.org/man/index.html

Name	Title	URL
Libxcb API	Libxcb API	http://xcb.freedesktop.org/XcbApi/
OpenGL 2.1	The OpenGL® Graphics System: A Specification (Version 2.1)	http://www.opengl.org/registry/doc/glspec2.1.20061201.pdf
OpenGL ABI	OpenGL® Application Binary Interface for Linux	http://www.opengl.org/registry/ABI/
OpenGL Extensions	OpenGL® Graphics with the X Window System® (Version 1.3)	http://opengl.org/documentation/specs/gl/glx1.3.pdf
OpenGL Utilities	The OpenGL Graphics System Utility Library (Version 1.3)	http://www.opengl.org/documentation/specs/glu/glu1.3.pdf
Pango 1.30.1 Reference Manual	Pango 1.30.1 Reference Manual	http://developer.gnome.org/pango/1.30/index.html
POSIX 1003.1-2008 (ISO/IEC 9945-2009)	Portable Operating System Interface (POSIX®) 2008 Edition / The Open Group Technical Standard Base Specifications, Issue 7	http://www.unix.org/version4/
QtCore 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtcore.html
QtGui 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtgui.html
QtNetwork 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtnetwork.html
QtOpenGL 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtopengl.html
QtSql 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtsql.html
QtSvg 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtsvg.html
QtXml 4.2.0	Qt 4.2.0 Reference Manual	http://doc.qt.digia.com/4.2/qtqml.html
The MIT Shared Memory Extension	MIT-SHM - The MIT Shared Memory Extension - X version 11, Release 5	http://refspecs.linux-foundation.org/X11/mit-shm.pdf

Name	Title	URL
X Display Power Management Signaling	X Display Power Management Signaling (DPMS) Extension - Library Specification - Version 1.0	http://refspecs.linux-foundation.org/X11/DPMslib.pdf
X Extended Visual Interface Extension	Extended Visual Information Extension - Version 1.0	http://refspecs.linux-foundation.org/X11/evi.pdf
X Nonrectangular Window Shape Extension Library	X Nonrectangular Window Shape Extension Library - Version 1.0	http://refspecs.linux-foundation.org/X11/shapelib.pdf
X Record Extension Library	X Record Extension Library - Version 1.13	http://refspecs.linux-foundation.org/X11/recordlib.pdf
X Security Extension Specification	Security Extension Specification - Version 7.1	http://refspecs.linux-foundation.org/X11/security.pdf
X Synchronization Extension Library	X Synchronization Extension Library - Version 3.0	http://refspecs.linux-foundation.org/X11/syncnlib.pdf
X11 C Library	Xlib - C Language X Interface - X Version 11 Release 6.4	http://refspecs.linux-foundation.org/X11/xlib.pdf
X11 Input Library	X Input Device Extension Library - X Version 11, Release 6.4	http://refspecs.linux-foundation.org/X11/Xinput.pdf
X11 Inter-Client Exchange	Inter-Client Exchange Library - Version 1.0	http://refspecs.linux-foundation.org/X11/ICElib.pdf
X11 Keyboard Extension	X Keyboard Extension Library Specification - X Version 11, Release 6.4	http://refspecs.linux-foundation.org/X11/XKBlib.pdf
X11 Session Management	X Session Management Library - Version 1.0	http://refspecs.linux-foundation.org/X11/SMlib.pdf
X11 Toolkit Intrinsics	X Toolkit Intrinsics - C Language Interface - X Version 11, Release 6.4	http://refspecs.linux-foundation.org/X11/intrinsics.pdf
Xft Placeholder	Xft Specification Placeholder	
Xrender Placeholder	Xrender Specification Placeholder	http://refspecs.linux-foundation.org/X11/XRenderProtocol.html

Name	Title	URL
XTEST Extension Library	XTEST Extension Library - Version 2.2	http://refspecs.linux-foundation.org/X11/xtestlib.pdf

2.2 Informative References/Bibliography

In addition, the specifications listed below provide essential background information to implementors of this specification. These references are included for information only.

Table 2-2 Other References

Name	Title	URL
A description on how to use and modify libpng	A description on how to use and modify libpng	http://www.libpng.org/pub/png/libpng-1.2.5-manual.html
ALSA Library API Reference	ALSA Library API Reference	http://www.alsa-project.org/alsa-doc/alsa-lib/
Base Directory Spec	XDG Base Directory Specification Version 0.6	http://standards.freedesktop.org/basedir-spec/basedir-spec-0.6.html
Cairo API Reference	Cairo Vector Graphics API Specification for 1.12.4	http://cairographics.org/manual-1.12.4
Desktop Entry Spec	Desktop Entry Specification Version 1.0	http://standards.freedesktop.org/desktop-entry-spec/desktop-entry-spec-1.0.html
Desktop Menu Spec	Desktop Menu Specification Version 1.0	http://standards.freedesktop.org/menu-spec/menu-spec-1.0.html
FreeType 2.2 Reference	FreeType 2.2.1 API Reference	http://refspecs.linuxfoundation.org/freetype/freetype-doc-2.2.1/docs/reference/ft2-toc.html
Icon Theme Spec	Icon Theme Specification Version 0.11	http://standards.freedesktop.org/icon-theme-spec/icon-theme-spec-0.11.html
Independent JPEG Group	Independent JPEG Group	http://www.iijg.org/
xdg-utils reference	Portland Project XDG Utilities Reference 1.0	http://portland.freedesktop.org/xdg-utils-1.0/

3 Requirements

3.1 Relevant Libraries

The libraries listed in Table 3-1 shall be available on a Linux Standard Base - Desktop system, with the specified runtime names. This list may be supplemented or amended by the architecture-specific specification.

Table 3-1 Standard Library Names

Library	Runtime Name
libGL	libGL.so.1
libGLU	libGLU.so.1
libICE	libICE.so.6
libQtCore	libQtCore.so.4
libQtGui	libQtGui.so.4
libQtNetwork	libQtNetwork.so.4
libQtOpenGL	libQtOpenGL.so.4
libQtSql	libQtSql.so.4
libQtSvg	libQtSvg.so.4
libQtXml	libQtXml.so.4
libSM	libSM.so.6
libX11	libX11.so.6
libXext	libXext.so.6
libXft	libXft.so.2
libXi	libXi.so.6
libXrender	libXrender.so.1
libXt	libXt.so.6
libXtst	libXtst.so.6
libasound	libasound.so.2
libatk-1.0	libatk-1.0.so.0
libcairo	libcairo.so.2
libcairo-gobject	libcairo-gobject.so.2
libcairo-script-interpreter	libcairo-script-interpreter.so.2
libfontconfig	libfontconfig.so.1
libfreetype	libfreetype.so.6
libgdk-x11-2.0	libgdk-x11-2.0.so.0
libgdk_pixbuf-2.0	libgdk_pixbuf-2.0.so.0
libgdk_pixbuf_xlib-2.0	libgdk_pixbuf_xlib-2.0.so.0

Library	Runtime Name
libgio-2.0	libgio-2.0.so.0
libglib-2.0	libglib-2.0.so.0
libgmodule-2.0	libgmodule-2.0.so.0
libgobject-2.0	libgobject-2.0.so.0
libgthread-2.0	libgthread-2.0.so.0
libgtk-x11-2.0	libgtk-x11-2.0.so.0
libjpeg	libjpeg.so.62
libpango-1.0	libpango-1.0.so.0
libpangocairo-1.0	libpangocairo-1.0.so.0
libpangoft2-1.0	libpangoft2-1.0.so.0
libpangoxft-1.0	libpangoxft-1.0.so.0
libpng12	libpng12.so.0
libtiff	libtiff.so.5
libxcb	libxcb.so.1

These libraries will be in an implementation-defined directory which the dynamic linker shall search by default.

4 Terms and Definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 2382, ISO 80000-2, and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <http://www.electropedia.org/>

4.1

archLSB

Some LSB specification documents have both a generic, architecture-neutral part and an architecture-specific part. The latter describes elements whose definitions may be unique to a particular processor architecture. The term archLSB may be used in the generic part to refer to the corresponding section of the architecture-specific part.

4.2

Binary Standard, ABI

The total set of interfaces that are available to be used in the compiled binary code of a conforming application, including the run-time details such as calling conventions, binary format, C++ name mangling, etc.

4.3

Implementation-defined

Describes a value or behavior that is not defined by this document but is selected by an implementor. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence of the value or behavior. An application that relies on such a value or behavior cannot be assured to be portable across conforming implementations. The implementor shall document such a value or behavior so that it can be used correctly by an application.

4.4

Shell Script

A file that is read by an interpreter (e.g., awk). The first line of the shell script includes a reference to its interpreter binary.

4.5

Source Standard, API

The total set of interfaces that are available to be used in the source code of a conforming application. Due to translations, the Binary Standard and the Source Standard may contain some different interfaces.

4.6

Undefined

Describes the nature of a value or behavior not defined by this document which results from use of an invalid program construct or invalid data input. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence or validity of the value or behavior. An application that relies on any particular value or behavior cannot be assured to be portable across conforming implementations.

4.7

Unspecified

Describes the nature of a value or behavior not specified by this document which results from use of a valid program construct or valid data input. The value or behavior may vary among implementations that conform to this document. An application should not rely on the existence or validity of the value or behavior. An application that relies on any particular value or behavior cannot be assured to be portable across conforming implementations.

In addition, for the portions of this specification which build on IEEE Std 1003.1-2001, the definitions given in *IEEE Std 1003.1-2001, Base Definitions, Chapter 3* apply.

5 Documentation Conventions

Throughout this document, the following typographic conventions are used:

`function()`

the name of a function

command

the name of a command or utility

CONSTANT

a constant value

parameter

a parameter

variable

a variable

Throughout this specification, several tables of interfaces are presented. Each entry in these tables has the following format:

name

the name of the interface

(symver)

An optional symbol version identifier, if required.

[refno]

A reference number indexing the table of referenced specifications that follows this table.

For example,

forkpty(GLIBC_2.0) [SUSv4]

refers to the interface named `forkpty()` with symbol version `GLIBC_2.0` that is defined in the reference indicated by the tag `SUSv4`.

Note: For symbols with versions which differ between architectures, the symbol versions are defined in the architecture specific parts of of this module specification only. In the generic part, they will appear without symbol versions.

II Graphic Libraries

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

6 Libraries

The X Libraries should be built thread-safe.

6.1 Interfaces for libX11

Table 6-1 defines the library name and shared object name for the libX11 library

Table 6-1 libX11 Definition

Library:	libX11
SONAME:	libX11.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

[XKBlib] X11 Keyboard Extension

[Xlib] X11 C Library

6.1.1 X Window System Interface

6.1.1.1 Interfaces for X Window System Interface

An LSB conforming implementation shall provide the generic functions for X Window System Interface specified in Table 6-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-2 libX11 - X Window System Interface Function Interfaces

XActivateScreen Saver [Xlib]	XAddConnection Watch [Xlib]	XAddExtension [Xlib]	XAddHost [Xlib]
XAddHosts [Xlib]	XAddPixel [Xlib]	XAddToExtensionList [Xlib]	XAddToSaveSet [Xlib]
XAllPlanes [Xlib]	XAllocClassHint [Xlib]	XAllocColor [Xlib]	XAllocColorCells [Xlib]
XAllocColorPlanes [Xlib]	XAllocIconSize [Xlib]	XAllocNamedColor [Xlib]	XAllocSizeHints [Xlib]
XAllocStandardColorMap [Xlib]	XAllocWMHints [Xlib]	XAllowEvents [Xlib]	XAutoRepeatOff [Xlib]
XAutoRepeatOn [Xlib]	XBaseFontNameListOfFontSet [Xlib]	XBell [Xlib]	XBitmapBitOrder [Xlib]
XBitmapPad [Xlib]	XBitmapUnit [Xlib]	XBlackPixel [Xlib]	XBlackPixelOfScreen [Xlib]
XCellsOfScreen [Xlib]	XChangeActivePointerGrab [Xlib]	XChangeGC [Xlib]	XChangeKeyboardControl [Xlib]
XChangeKeyboardMapping [Xlib]	XChangePointerControl [Xlib]	XChangeProperty [Xlib]	XChangeSaveSet [Xlib]

XChangeWindowAttributes [Xlib]	XCheckIfEvent [Xlib]	XCheckMaskEvent [Xlib]	XCheckTypedEvent [Xlib]
XCheckTypedWindowEvent [Xlib]	XCheckWindowEvent [Xlib]	XCirculateSubwindows [Xlib]	XCirculateSubwindowsDown [Xlib]
XCirculateSubwindowsUp [Xlib]	XClearArea [Xlib]	XClearWindow [Xlib]	XClipBox [Xlib]
XCloseDisplay [Xlib]	XCloseIM [Xlib]	XCloseOM [Xlib]	XConfigureWindow [Xlib]
XConnectionNumber [Xlib]	XContextDependentDrawing [Xlib]	XContextualDrawing [Xlib]	XConvertCase [Xlib]
XConvertSelection [Xlib]	XCopyArea [Xlib]	XCopyColormapAndFree [Xlib]	XCopyGC [Xlib]
XCopyPlane [Xlib]	XCreateBitmapFromData [Xlib]	XCreateColormap [Xlib]	XCreateFontCursor [Xlib]
XCreateFontSet [Xlib]	XCreateGC [Xlib]	XCreateGlyphCursor [Xlib]	XCreateIC [Xlib]
XCreateImage [Xlib]	XCreateOC [Xlib]	XCreatePixmap [Xlib]	XCreatePixmapCursor [Xlib]
XCreatePixmapFromBitmapData [Xlib]	XCreateRegion [Xlib]	XCreateSimpleWindow [Xlib]	XCreateWindow [Xlib]
XDefaultColormap [Xlib]	XDefaultColormapOfScreen [Xlib]	XDefaultDepth [Xlib]	XDefaultDepthOfScreen [Xlib]
XDefaultGC [Xlib]	XDefaultGCOfScreen [Xlib]	XDefaultRootWindow [Xlib]	XDefaultScreen [Xlib]
XDefaultScreenOfDisplay [Xlib]	XDefaultString [Xlib]	XDefaultVisual [Xlib]	XDefaultVisualOfScreen [Xlib]
XDefineCursor [Xlib]	XDeleteContext [Xlib]	XDeleteModifiermapEntry [Xlib]	XDeleteProperty [Xlib]
XDestroyIC [Xlib]	XDestroyImage [Xlib]	XDestroyOC [Xlib]	XDestroyRegion [Xlib]
XDestroySubwindows [Xlib]	XDestroyWindow [Xlib]	XDirectionalDependentDrawing [Xlib]	XDisableAccessControl [Xlib]
XDisplayCells [Xlib]	XDisplayHeight [Xlib]	XDisplayHeightMM [Xlib]	XDisplayKeycodes [Xlib]
XDisplayMotionBufferSize [Xlib]	XDisplayName [Xlib]	XDisplayOfIM [Xlib]	XDisplayOfOM [Xlib]

XDisplayOfScreen [Xlib]	XDisplayPlanes [Xlib]	XDisplayString [Xlib]	XDisplayWidth [Xlib]
XDisplayWidthMM [Xlib]	XDoesBackingStore [Xlib]	XDoesSaveUnders [Xlib]	XDrawArc [Xlib]
XDrawArcs [Xlib]	XDrawImageString [Xlib]	XDrawImageString16 [Xlib]	XDrawLine [Xlib]
XDrawLines [Xlib]	XDrawPoint [Xlib]	XDrawPoints [Xlib]	XDrawRectangle [Xlib]
XDrawRectangles [Xlib]	XDrawSegments [Xlib]	XDrawString [Xlib]	XDrawString16 [Xlib]
XDrawText [Xlib]	XDrawText16 [Xlib]	XEHeadOfExtensionList [Xlib]	XESetBeforeFlush [Xlib]
XESetCloseDisplay [Xlib]	XESetCopyGC [Xlib]	XESetCreateFont [Xlib]	XESetCreateGC [Xlib]
XESetError [Xlib]	XESetErrorString [Xlib]	XESetEventToWire [Xlib]	XESetFlushGC [Xlib]
XESetFreeFont [Xlib]	XESetFreeGC [Xlib]	XESetPrintErrorValues [Xlib]	XESetWireToError [Xlib]
XESetWireToEvent [Xlib]	XEmptyRegion [Xlib]	XEnableAccessControl [Xlib]	XEqualRegion [Xlib]
XEventMaskOfScreen [Xlib]	XEventsQueued [Xlib]	XExtendedMaxRequestSize [Xlib]	XExtentsOfFontSet [Xlib]
XFetchBuffer [Xlib]	XFetchBytes [Xlib]	XFetchName [Xlib]	XFillArc [Xlib]
XFillArcs [Xlib]	XFillPolygon [Xlib]	XFillRectangle [Xlib]	XFillRectangles [Xlib]
XFilterEvent [Xlib]	XFindContext [Xlib]	XFindOnExtensionList [Xlib]	XFlush [Xlib]
XFlushGC [Xlib]	XFontsOfFontSet [Xlib]	XForceScreenSaver [Xlib]	XFree [Xlib]
XFreeColormap [Xlib]	XFreeColors [Xlib]	XFreeCursor [Xlib]	XFreeExtensionList [Xlib]
XFreeFont [Xlib]	XFreeFontInfo [Xlib]	XFreeFontNames [Xlib]	XFreeFontPath [Xlib]
XFreeFontSet [Xlib]	XFreeGC [Xlib]	XFreeModifiermap [Xlib]	XFreePixmap [Xlib]
XFreeStringList [Xlib]	XGContextFromGC [Xlib]	XGeometry [Xlib]	XGetAtomName [Xlib]
XGetAtomNames [Xlib]	XGetClassHint [Xlib]	XGetCommand [Xlib]	XGetDefault [Xlib]
XGetErrorDatabaseText [Xlib]	XGetErrorText [Xlib]	XGetFontPath [Xlib]	XGetFontProperty [Xlib]

XGetGCValues [Xlib]	XGetGeometry [Xlib]	XGetICValues [Xlib]	XGetIMValues [Xlib]
XGetIconName [Xlib]	XGetIconSizes [Xlib]	XGetImage [Xlib]	XGetInputFocus [Xlib]
XGetKeyboardControl [Xlib]	XGetKeyboardMapping [Xlib]	XGetModifierMapping [Xlib]	XGetMotionEvents [Xlib]
XGetNormalHints [Xlib]	XGetOCValues [Xlib]	XGetOMValues [Xlib]	XGetPixel [Xlib]
XGetPointerControl [Xlib]	XGetPointerMapping [Xlib]	XGetRGBColormaps [Xlib]	XGetScreenSaver [Xlib]
XGetSelectionOwner [Xlib]	XGetSizeHints [Xlib]	XGetStandardColormap [Xlib]	XGetSubImage [Xlib]
XGetTextProperty [Xlib]	XGetTransientForHint [Xlib]	XGetVisualInfo [Xlib]	XGetWMClientMachine [Xlib]
XGetWMColormapWindows [Xlib]	XGetWMHints [Xlib]	XGetWMIconName [Xlib]	XGetWMName [Xlib]
XGetWMNormalHints [Xlib]	XGetWMProtocols [Xlib]	XGetWMSizeHints [Xlib]	XGetWindowAttributes [Xlib]
XGetWindowProperty [Xlib]	XGetZoomHints [Xlib]	XGrabButton [Xlib]	XGrabKey [Xlib]
XGrabKeyboard [Xlib]	XGrabPointer [Xlib]	XGrabServer [Xlib]	XHeightMMOfScreen [Xlib]
XHeightOfScreen [Xlib]	XIMOfC [Xlib]	XIconifyWindow [Xlib]	XIfEvent [Xlib]
XImageByteOrder [Xlib]	XInitExtension [Xlib]	XInitImage [Xlib]	XInitThreads [Xlib]
XInsertModifiermapEntry [Xlib]	XInstallColormap [Xlib]	XInternAtom [Xlib]	XInternAtoms [Xlib]
XInternalConnectionNumbers [Xlib]	XIntersectRegion [Xlib]	XKeycodeToKeysym [Xlib]	XKeysymToKeycode [Xlib]
XKeysymToString [Xlib]	XKillClient [Xlib]	XLastKnownRequestProcessed [Xlib]	XListDepths [Xlib]
XListExtensions [Xlib]	XListFonts [Xlib]	XListFontsWithInfo [Xlib]	XListHosts [Xlib]
XListInstalledColormaps [Xlib]	XListPixmapFormats [Xlib]	XListProperties [Xlib]	XLoadFont [Xlib]
XLoadQueryFont [Xlib]	XLlocaleOfFontSet [Xlib]	XLlocaleOfIM [Xlib]	XLlocaleOfOM [Xlib]
XLockDisplay [Xlib]	XLookupColor [Xlib]	XLookupKeysym [Xlib]	XLookupString [Xlib]

XLowerWindow [Xlib]	XMapRaised [Xlib]	XMapSubwindows [Xlib]	XMapWindow [Xlib]
XMaskEvent [Xlib]	XMatchVisualInfo [Xlib]	XMaxCmapsOfScreen [Xlib]	XMaxRequestSize [Xlib]
XMinCmapsOfScreen [Xlib]	XMoveResizeWindow [Xlib]	XMoveWindow [Xlib]	XNewModifiermap [Xlib]
XNextEvent [Xlib]	XNextRequest [Xlib]	XNoOp [Xlib]	XOMOfOC [Xlib]
XOffsetRegion [Xlib]	XOpenDisplay [Xlib]	XOpenIM [Xlib]	XOpenOM [Xlib]
XParseColor [Xlib]	XParseGeometry [Xlib]	XPeekEvent [Xlib]	XPeekIfEvent [Xlib]
XPending [Xlib]	XPlanesOfScreen [Xlib]	XPointInRegion [Xlib]	XPolygonRegion [Xlib]
XProcessInternalConnection [Xlib]	XProtocolRevision [Xlib]	XProtocolVersion [Xlib]	XPutBackEvent [Xlib]
XPutImage [Xlib]	XPutPixel [Xlib]	XQLength [Xlib]	XQueryBestCursor [Xlib]
XQueryBestSize [Xlib]	XQueryBestStipple [Xlib]	XQueryBestTile [Xlib]	XQueryColor [Xlib]
XQueryColors [Xlib]	XQueryExtension [Xlib]	XQueryFont [Xlib]	XQueryKeymap [Xlib]
XQueryPointer [Xlib]	XQueryTextExtents [Xlib]	XQueryTextExtents16 [Xlib]	XQueryTree [Xlib]
XRaiseWindow [Xlib]	XReadBitmapFile [Xlib]	XReadBitmapFileData [Xlib]	XRebindKeysym [Xlib]
XRecolorCursor [Xlib]	XReconfigureWindow [Xlib]	XRectInRegion [Xlib]	XRefreshKeyboardMapping [Xlib]
XRegisterIMInstantiateCallback [Xlib]	XRemoveConnectionWatch [Xlib]	XRemoveFromSaveSet [Xlib]	XRemoveHost [Xlib]
XRemoveHosts [Xlib]	XReparentWindow [Xlib]	XResetScreenSaver [Xlib]	XResizeWindow [Xlib]
XResourceManagerString [Xlib]	XRestackWindows [Xlib]	XRootWindow [Xlib]	XRootWindowOfScreen [Xlib]
XRotateBuffers [Xlib]	XRotateWindowProperties [Xlib]	XSaveContext [Xlib]	XScreenCount [Xlib]
XScreenNumberOfScreen [Xlib]	XScreenOfDisplay [Xlib]	XScreenResourceString [Xlib]	XSelectInput [Xlib]
XSendEvent [Xlib]	XServerVendor [Xlib]	XSetAccessControl [Xlib]	XSetAfterFunction [Xlib]

XSetArcMode [Xlib]	XSetAuthorization [Xlib]	XSetBackground [Xlib]	XSetClassHint [Xlib]
XSetClipMask [Xlib]	XSetClipOrigin [Xlib]	XSetClipRectangles [Xlib]	XSetCloseDownMode [Xlib]
XSetCommand [Xlib]	XSetDashes [Xlib]	XSetErrorHandler [Xlib]	XSetFillRule [Xlib]
XSetFillStyle [Xlib]	XSetFont [Xlib]	XSetFontPath [Xlib]	XSetForeground [Xlib]
XSetFunction [Xlib]	XSetGraphicsExposures [Xlib]	XSetICFocus [Xlib]	XSetICValues [Xlib]
XSetIMValues [Xlib]	XSetIOErrorHandler [Xlib]	XSetIconName [Xlib]	XSetIconSizes [Xlib]
XSetInputFocus [Xlib]	XSetLineAttributes [Xlib]	XSetLocaleModifiers [Xlib]	XSetModifierMapping [Xlib]
XSetNormalHints [Xlib]	XSetOCValues [Xlib]	XSetOMValues [Xlib]	XSetPlaneMask [Xlib]
XSetPointerMapping [Xlib]	XSetRGBColormaps [Xlib]	XSetRegion [Xlib]	XSetScreenSaver [Xlib]
XSetSelectionOwner [Xlib]	XSetSizeHints [Xlib]	XSetStandardColormap [Xlib]	XSetStandardProperties [Xlib]
XSetState [Xlib]	XSetStipple [Xlib]	XSetSubwindowMode [Xlib]	XSetTSTOrigin [Xlib]
XSetTextProperty [Xlib]	XSetTile [Xlib]	XSetTransientForHint [Xlib]	XSetWMClientMachine [Xlib]
XSetWMColormapWindows [Xlib]	XSetWMHints [Xlib]	XSetWMIconName [Xlib]	XSetWMName [Xlib]
XSetWMNormalHints [Xlib]	XSetWMProperties [Xlib]	XSetWMProtocol [Xlib]	XSetWMSizeHints [Xlib]
XSetWindowBackground [Xlib]	XSetWindowBackgroundPixmap [Xlib]	XSetWindowBorder [Xlib]	XSetWindowBorderPixmap [Xlib]
XSetWindowBorderWidth [Xlib]	XSetWindowColormap [Xlib]	XSetZoomHints [Xlib]	XShrinkRegion [Xlib]
XStoreBuffer [Xlib]	XStoreBytes [Xlib]	XStoreColor [Xlib]	XStoreColors [Xlib]
XStoreName [Xlib]	XStoreNamedColor [Xlib]	XStringListToTextProperty [Xlib]	XStringToKeySym [Xlib]
XSubImage [Xlib]	XSubtractRegion [Xlib]	XSupportsLocale [Xlib]	XSync [Xlib]
XSynchronize [Xlib]	XTextExtents [Xlib]	XTextExtents16 [Xlib]	XTextPropertyToStringList [Xlib]

XTextWidth [Xlib]	XTextWidth16 [Xlib]	XTranslateCoordinates [Xlib]	XUndefineCursor [Xlib]
XUngrabButton [Xlib]	XUngrabKey [Xlib]	XUngrabKeyboard [Xlib]	XUngrabPointer [Xlib]
XUngrabServer [Xlib]	XUninstallColorMap [Xlib]	XUnionRectWithRegion [Xlib]	XUnionRegion [Xlib]
XUnloadFont [Xlib]	XUnlockDisplay [Xlib]	XUnmapSubwindows [Xlib]	XUnmapWindow [Xlib]
XUnregisterIMInstantiateCallback [Xlib]	XUnsetICFocus [Xlib]	XVaCreateNestedList [Xlib]	XVendorRelease [Xlib]
XVisualIDFromVisual [Xlib]	XWMGeometry [Xlib]	XWarpPointer [Xlib]	XWhitePixel [Xlib]
XWhitePixelOfScreen [Xlib]	XWidthMMOfScreen [Xlib]	XWidthOfScreen [Xlib]	XWindowEvent [Xlib]
XWithdrawWindow [Xlib]	XWriteBitmapFile [Xlib]	XXorRegion [Xlib]	XauDisposeAuth [Xlib]
XauFileName [Xlib]	XauGetBestAuthByAddr [Xlib]	XauReadAuth [Xlib]	XcmsAddColorSpace [Xlib]
XcmsAddFunctionSet [Xlib]	XcmsAllocColor [Xlib]	XcmsAllocNamedColor [Xlib]	XcmsCCCOOfColorMap [Xlib]
XcmsCIELabClipL [Xlib]	XcmsCIELabClipLab [Xlib]	XcmsCIELabClipab [Xlib]	XcmsCIELabQueryMaxC [Xlib]
XcmsCIELabQueryMaxL [Xlib]	XcmsCIELabQueryMaxLC [Xlib]	XcmsCIELabQueryMinL [Xlib]	XcmsCIELabToCIEXYZ [Xlib]
XcmsCIELabWhiteShiftColors [Xlib]	XcmsCIELuvClipL [Xlib]	XcmsCIELuvClipLuv [Xlib]	XcmsCIELuvClipuv [Xlib]
XcmsCIELuvQueryMaxC [Xlib]	XcmsCIELuvQueryMaxL [Xlib]	XcmsCIELuvQueryMaxLC [Xlib]	XcmsCIELuvQueryMinL [Xlib]
XcmsCIELuvToCIEuvY [Xlib]	XcmsCIELuvWhiteShiftColors [Xlib]	XcmsCIEXYZToCIELab [Xlib]	XcmsCIEXYZToCIEuvY [Xlib]
XcmsCIEXYZToCIExyY [Xlib]	XcmsCIEXYZToRGBi [Xlib]	XcmsCIEuvYToCIELuv [Xlib]	XcmsCIEuvYToCIEXYZ [Xlib]
XcmsCIEuvYToTekHVC [Xlib]	XcmsCIExyYToCIEXYZ [Xlib]	XcmsClientWhitePointOfCCC [Xlib]	XcmsConvertColors [Xlib]
XcmsCreateCCC [Xlib]	XcmsDefaultCCC [Xlib]	XcmsDisplayOfCCC [Xlib]	XcmsFormatOfPrefix [Xlib]
XcmsFreeCCC [Xlib]	XcmsLookupColor [Xlib]	XcmsPrefixOfFormat [Xlib]	XcmsQueryBlack [Xlib]

XcmsQueryBlue [Xlib]	XcmsQueryColor [Xlib]	XcmsQueryColors [Xlib]	XcmsQueryGreen [Xlib]
XcmsQueryRed [Xlib]	XcmsQueryWhite [Xlib]	XcmsRGBToRGBi [Xlib]	XcmsRGBiToCIEXYZ [Xlib]
XcmsRGBiToRGB [Xlib]	XcmsScreenNumberOfCCC [Xlib]	XcmsScreenWhitePointOfCCC [Xlib]	XcmsSetCCCOColorMap [Xlib]
XcmsSetCompressionProc [Xlib]	XcmsSetWhiteAdjustProc [Xlib]	XcmsSetWhitePoint [Xlib]	XcmsStoreColor [Xlib]
XcmsStoreColors [Xlib]	XcmsTekHVCClipC [Xlib]	XcmsTekHVCClipV [Xlib]	XcmsTekHVCClipVC [Xlib]
XcmsTekHVCQueryMaxC [Xlib]	XcmsTekHVCQueryMaxV [Xlib]	XcmsTekHVCQueryMaxVC [Xlib]	XcmsTekHVCQueryMaxVSamples [Xlib]
XcmsTekHVCQueryMinV [Xlib]	XcmsTekHVCToCIEuvY [Xlib]	XcmsTekHVCWhiteShiftColors [Xlib]	XcmsVisualOfCCC [Xlib]
XkbAllocClientMap [XKBlib]	XkbAllocCompatMap [XKBlib]	XkbAllocControls [XKBlib]	XkbAllocGeomColors [XKBlib]
XkbAllocGeomDoodads [XKBlib]	XkbAllocGeomKeyAliases [XKBlib]	XkbAllocGeomKeys [XKBlib]	XkbAllocGeomOutlines [XKBlib]
XkbAllocGeomOverlayKeys [XKBlib]	XkbAllocGeomOverlayRows [XKBlib]	XkbAllocGeomOverlays [XKBlib]	XkbAllocGeomPoints [XKBlib]
XkbAllocGeomProps [XKBlib]	XkbAllocGeomRows [XKBlib]	XkbAllocGeomSectionDoodads [XKBlib]	XkbAllocGeomSections [XKBlib]
XkbAllocGeomShapes [XKBlib]	XkbAllocGeometry [XKBlib]	XkbAllocIndicatorMaps [XKBlib]	XkbAllocKeyboard [XKBlib]
XkbAllocNames [XKBlib]	XkbAllocServerMap [XKBlib]	XkbApplyCompatMapToKey [XKBlib]	XkbBell [XKBlib]
XkbBellEvent [XKBlib]	XkbChangeEnabledControls [XKBlib]	XkbChangeMap [XKBlib]	XkbChangeNames [XKBlib]
XkbChangeTypesOfKey [XKBlib]	XkbComputeEffectiveMap [XKBlib]	XkbComputeRowBounds [XKBlib]	XkbComputeSectionBounds [XKBlib]
XkbComputeShapeBounds [XKBlib]	XkbComputeShapeTop [XKBlib]	XkbCopyKeyType [XKBlib]	XkbCopyKeyTypes [XKBlib]
XkbFindOverlayForKey [XKBlib]	XkbForceBell [XKBlib]	XkbFreeClientMap [XKBlib]	XkbFreeCompatMap [XKBlib]

XkbFreeComponentList [XKBlib]	XkbFreeControls [XKBlib]	XkbFreeGeomColors [XKBlib]	XkbFreeGeomDodads [XKBlib]
XkbFreeGeomKeyAliases [XKBlib]	XkbFreeGeomKeys [XKBlib]	XkbFreeGeomOutlines [XKBlib]	XkbFreeGeomOverlayKeys [XKBlib]
XkbFreeGeomOverlayRows [XKBlib]	XkbFreeGeomOverlays [XKBlib]	XkbFreeGeomPoints [XKBlib]	XkbFreeGeomProperties [XKBlib]
XkbFreeGeomRows [XKBlib]	XkbFreeGeomSections [XKBlib]	XkbFreeGeomShapes [XKBlib]	XkbFreeGeometry [XKBlib]
XkbFreeIndicatorMaps [XKBlib]	XkbFreeKeyboard [XKBlib]	XkbFreeNames [XKBlib]	XkbFreeServerMap [XKBlib]
XkbGetAutoRepeatRate [XKBlib]	XkbGetCompatMap [XKBlib]	XkbGetControls [XKBlib]	XkbGetGeometry [XKBlib]
XkbGetIndicatorMap [XKBlib]	XkbGetIndicatorState [XKBlib]	XkbGetKeyActions [XKBlib]	XkbGetKeyBehaviors [XKBlib]
XkbGetKeyExplicitComponents [XKBlib]	XkbGetKeyModifierMap [XKBlib]	XkbGetKeySyms [XKBlib]	XkbGetKeyTypes [XKBlib]
XkbGetKeyboard [XKBlib]	XkbGetKeyboardByName [XKBlib]	XkbGetMap [XKBlib]	XkbGetMapChanges [XKBlib]
XkbGetNamedGeometry [XKBlib]	XkbGetNamedIndicator [XKBlib]	XkbGetNames [XKBlib]	XkbGetPerClientControls [XKBlib]
XkbGetState [XKBlib]	XkbGetUpdatedMap [XKBlib]	XkbGetVirtualMods [XKBlib]	XkbGetXlibControls [XKBlib]
XkbIgnoreExtension [XKBlib]	XkbInitCanonicalKeyTypes [XKBlib]	XkbKeyTypesForCoreSymbols [XKBlib]	XkbKeycodeToKeysym [XKBlib]
XkbKeysymToModifiers [XKBlib]	XkbLatchGroup [XKBlib]	XkbLatchModifiers [XKBlib]	XkbLibraryVersion [XKBlib]
XkbListComponents [XKBlib]	XkbLockGroup [XKBlib]	XkbLockModifiers [XKBlib]	XkbLookupKeyBinding [XKBlib]
XkbLookupKeySym [XKBlib]	XkbNoteControlsChanges [XKBlib]	XkbNoteMapChanges [XKBlib]	XkbNoteNameChanges [XKBlib]
XkbOpenDisplay [XKBlib]	XkbQueryExtension [XKBlib]	XkbRefreshKeyboardMapping [XKBlib]	XkbResizeKeyActions [XKBlib]
XkbResizeKeySyms [XKBlib]	XkbResizeKeyType [XKBlib]	XkbSelectEventDetails [XKBlib]	XkbSelectEvents [XKBlib]

XkbSetAtomFuncs [XKBLib]	XkbSetAutoRepeatRate [XKBLib]	XkbSetAutoResetControls [XKBLib]	XkbSetCompatMap [XKBLib]
XkbSetControls [XKBLib]	XkbSetDebuggingFlags [XKBLib]	XkbSetDetectableAutoRepeat [XKBLib]	XkbSetGeometry [XKBLib]
XkbSetIgnoreLockMods [XKBLib]	XkbSetIndicatorMap [XKBLib]	XkbSetMap [XKBLib]	XkbSetNamedIndicator [XKBLib]
XkbSetNames [XKBLib]	XkbSetPerClientControls [XKBLib]	XkbSetServerInternalMods [XKBLib]	XkbSetXlibControls [XKBLib]
XkbToControl [XKBLib]	XkbTranslateKeyCode [XKBLib]	XkbTranslateKeySym [XKBLib]	XkbUpdateMapFromCore [XKBLib]
XkbUseExtension [XKBLib]	XkbVirtualModsToReal [XKBLib]	XmbDrawImageString [Xlib]	XmbDrawString [Xlib]
XmbDrawText [Xlib]	XmbLookupString [Xlib]	XmbResetIC [Xlib]	XmbSetWMPProperties [Xlib]
XmbTextEscapement [Xlib]	XmbTextExtents [Xlib]	XmbTextListToTextProperty [Xlib]	XmbTextPerCharExtents [Xlib]
XmbTextPropertyToTextList [Xlib]	XrmCombineDatabase [Xlib]	XrmCombineFileDatabase [Xlib]	XrmDestroyDatabase [Xlib]
XrmEnumerateDatabase [Xlib]	XrmGetDatabase [Xlib]	XrmGetFileDatabase [Xlib]	XrmGetResource [Xlib]
XrmGetStringDatabase [Xlib]	XrmInitialize [Xlib]	XrmLocaleOfDatabase [Xlib]	XrmMergeDatabases [Xlib]
XrmParseCommand [Xlib]	XrmPermStringToQuark [Xlib]	XrmPutFileDatabase [Xlib]	XrmPutLineResource [Xlib]
XrmPutResource [Xlib]	XrmPutStringResource [Xlib]	XrmQGetResource [Xlib]	XrmQGetSearchList [Xlib]
XrmQGetSearchResource [Xlib]	XrmQPutResource [Xlib]	XrmQPutStringResource [Xlib]	XrmQuarkToString [Xlib]
XrmSetDatabase [Xlib]	XrmStringToBindingQuarkList [Xlib]	XrmStringToQuark [Xlib]	XrmStringToQuarkList [Xlib]
XrmUniqueQuark [Xlib]	Xutf8TextListToTextProperty [LSB]	Xutf8TextPropertyToTextList [LSB]	XwcDrawImageString [Xlib]
XwcDrawString [Xlib]	XwcDrawText [Xlib]	XwcFreeStringList [Xlib]	XwcLookupString [Xlib]

XwcResetIC [Xlib]	XwcTextEscape ment [Xlib]	XwcTextExtents [Xlib]	XwcTextListToT extProperty [Xlib]
XwcTextPerChar Extents [Xlib]	XwcTextPropert yToTextList [Xlib]		

6.2 Data Definitions for libX11

This section defines global identifiers and their values that are associated with interfaces contained in libX11. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.2.1 X11/X.h

```
#define X_PROTOCOL_REVISION 0
#define X_PROTOCOL 11

typedef unsigned long int XID;
typedef unsigned long int Mask;
typedef unsigned long int Atom;
typedef unsigned long int VisualID;
typedef unsigned long int Time;
typedef XID Window;
typedef XID Font;
typedef XID Pixmap;
typedef unsigned char KeyCode;
typedef XID Drawable;
typedef XID Cursor;
typedef XID Colormap;
typedef XID GCContext;
typedef XID KeySym;

#define AllTemporary 0L
#define AnyButton 0L
#define AnyKey 0L
#define AnyPropertyType 0L
#define CopyFromParent 0L
#define CurrentTime 0L
#define NoSymbol 0L
#define None 0L
#define PointerWindow 0L
#define InputFocus 1L
#define ParentRelative 1L
#define PointerRoot 1L

#define KeyPressMask (1L<<0)
```

```

#define KeyReleaseMask (1L<<1)
#define Button3MotionMask (1L<<10)
#define Button4MotionMask (1L<<11)
#define Button5MotionMask (1L<<12)
#define ButtonMotionMask (1L<<13)
#define KeymapStateMask (1L<<14)
#define ExposureMask (1L<<15)
#define VisibilityChangeMask (1L<<16)
#define StructureNotifyMask (1L<<17)
#define ResizeRedirectMask (1L<<18)
#define SubstructureNotifyMask (1L<<19)
#define ButtonPressMask (1L<<2)
#define SubstructureRedirectMask (1L<<20)
#define FocusChangeMask (1L<<21)
#define PropertyChangeMask (1L<<22)
#define ColormapChangeMask (1L<<23)
#define OwnerGrabButtonMask (1L<<24)
#define ButtonReleaseMask (1L<<3)
#define EnterWindowMask (1L<<4)
#define LeaveWindowMask (1L<<5)
#define PointerMotionMask (1L<<6)
#define PointerMotionHintMask (1L<<7)
#define Button1MotionMask (1L<<8)
#define Button2MotionMask (1L<<9)

#define NoEventMask 0L
#define FocusOut 10
#define KeymapNotify 11
#define Expose 12
#define GraphicsExpose 13
#define NoExpose 14
#define VisibilityNotify 15
#define CreateNotify 16
#define DestroyNotify 17
#define UnmapNotify 18
#define MapNotify 19
#define KeyPress 2
#define MapRequest 20
#define ReparentNotify 21
#define ConfigureNotify 22
#define ConfigureRequest 23
#define GravityNotify 24
#define ResizeRequest 25
#define CirculateNotify 26
#define CirculateRequest 27
#define PropertyNotify 28
#define SelectionClear 29
#define KeyRelease 3
#define SelectionRequest 30
#define SelectionNotify 31
#define ColormapNotify 32
#define ClientMessage 33
#define MappingNotify 34
#define LASTEvent 35
#define ButtonPress 4
#define ButtonRelease 5
#define MotionNotify 6
#define EnterNotify 7
#define LeaveNotify 8
#define FocusIn 9

#define ShiftMask (1<<0)
#define LockMask (1<<1)
#define ControlMask (1<<2)
#define Mod1Mask (1<<3)
#define Mod2Mask (1<<4)

```

```

#define Mod3Mask      (1<<5)
#define Mod4Mask      (1<<6)
#define Mod5Mask      (1<<7)

#define ShiftMapIndex  0
#define LockMapIndex   1
#define ControlMapIndex 2
#define Mod1MapIndex   3
#define Mod2MapIndex   4
#define Mod3MapIndex   5
#define Mod4MapIndex   6
#define Mod5MapIndex   7

#define Button3Mask    (1<<10)
#define Button4Mask    (1<<11)
#define Button5Mask    (1<<12)
#define AnyModifier    (1<<15)
#define Button1Mask    (1<<8)
#define Button2Mask    (1<<9)

#define Button1 1
#define Button2 2
#define Button3 3
#define Button4 4
#define Button5 5

#define NotifyNormal    0
#define NotifyGrab      1
#define NotifyHint      1
#define NotifyUngrab    2
#define NotifyWhileGrabbed 3

#define NotifyAncestor  0
#define NotifyVirtual    1
#define NotifyInferior   2
#define NotifyNonlinear  3
#define NotifyNonlinearVirtual 4
#define NotifyPointer     5
#define NotifyPointerRoot 6
#define NotifyDetailNone  7

#define VisibilityUnobscured 0
#define VisibilityPartiallyObscured 1
#define VisibilityFullyObscured 2

#define PlaceOnTop      0
#define PlaceOnBottom   1

#define FamilyInternet  0
#define FamilyDECnet     1
#define FamilyChaos      2
#define FamilyServerInterpreted 5
#define FamilyInternet6 6 /* IPv6 */

#define PropModeReplace 0
#define PropertyNewValue 0
#define PropertyDelete 1

#define ColormapUninstalled 0
#define ColormapInstalled 1

#define GrabModeSync 0
#define GrabModeAsync 1

#define GrabSuccess 0
#define AlreadyGrabbed 1

```

```

#define GrabInvalidTime 2
#define GrabNotViewable 3
#define GrabFrozen 4

#define AsyncPointer 0
#define SyncPointer 1
#define ReplayPointer 2
#define AsyncKeyboard 3
#define SyncKeyboard 4
#define ReplayKeyboard 5
#define AsyncBoth 6
#define SyncBoth 7

#define RevertToNone (int)None
#define RevertToPointerRoot (int)PointerRoot
#define RevertToParent 2

#define Success 0
#define BadRequest 1
#define BadAccess 10
#define BadAlloc 11
#define BadColor 12
#define FirstExtensionError 128
#define BadGC 13
#define BadIDChoice 14
#define BadName 15
#define BadLength 16
#define BadImplementation 17
#define BadValue 2
#define LastExtensionError 255
#define BadWindow 3
#define BadPixmap 4
#define BadAtom 5
#define BadCursor 6
#define BadFont 7
#define BadMatch 8
#define BadDrawable 9

#define CWX (1<<0)
#define InputOutput 1
#define InputOnly 2

#define CWBackPixmap (1L<<0)
#define CWBackPixel (1L<<1)
#define CWSaveUnder (1L<<10)
#define CWEventMask (1L<<11)
#define CWDontPropagate (1L<<12)
#define CWColormap (1L<<13)
#define CWCursor (1L<<14)
#define CWBorderPixmap (1L<<2)
#define CWBorderPixel (1L<<3)
#define CWBitGravity (1L<<4)
#define CWWinGravity (1L<<5)
#define CWBackingStore (1L<<6)
#define CWBackingPlanes (1L<<7)
#define CWBackingPixel (1L<<8)
#define CWOverrideRedirect (1L<<9)

#define CWY (1<<1)
#define CWWidth (1<<2)
#define CWHeight (1<<3)
#define CWBorderWidth (1<<4)
#define CWSibling (1<<5)
#define CWStackMode (1<<6)

#define ForgetGravity 0

```

```

#define UnmapGravity      0
#define NorthWestGravity  1
#define StaticGravity     10
#define NorthGravity      2
#define NorthEastGravity  3
#define WestGravity       4
#define CenterGravity     5
#define EastGravity       6
#define SouthWestGravity  7
#define SouthGravity      8
#define SouthEastGravity  9

#define NotUseful         0
#define WhenMapped       1
#define Always           2

#define IsUnmapped        0
#define IsUnviewable     1
#define IsViewable       2

#define SetModeInsert     0
#define SetModeDelete    1

#define DestroyAll        0
#define RetainPermanent  1
#define RetainTemporary  2

#define Above            0
#define Below           1
#define TopIf           2
#define BottomIf        3
#define Opposite        4

#define RaiseLowest      0
#define LowerHighest    1

#define PropModePrepend  1
#define PropModeAppend   2

#define GXclear          0x0
#define GXand            0x1
#define GXandReverse     0x2
#define GXcopy           0x3
#define GXandInverted    0x4
#define GXnoop           0x5
#define GXxor            0x6
#define GXor             0x7
#define GXnor            0x8
#define GXequiv          0x9
#define GXinvert         0xa
#define GXorReverse      0xb
#define GXcopyInverted   0xc
#define GXorInverted     0xd
#define GXnand           0xe
#define GXset            0xf

#define LineSolid        0
#define LineOnOffDash    1
#define LineDoubleDash   2

#define CapNotLast       0
#define CapButt          1
#define CapRound         2
#define CapProjecting    3

#define JoinMiter        0

```

```

#define JoinRound      1
#define JoinBevel      2

#define FillSolid      0
#define FillTiled      1
#define FillStippled   2
#define FillOpaqueStippled 3

#define EvenOddRule    0
#define WindingRule    1

#define ClipByChildren 0
#define IncludeInferiors 1

#define Unsorted       0
#define XYBitmap        0
#define YSorted        1
#define YXSorted        2
#define YXBanded        3

#define CoordModeOrigin 0
#define CoordModePrevious 1

#define Complex 0
#define Nonconvex 1
#define Convex 2

#define ArcChord 0
#define ArcPieSlice 1

#define GCFunction (1L<<0)
#define GCPlaneMask (1L<<1)
#define GCTile (1L<<10)
#define GCStipple (1L<<11)
#define GCTileStipXOrigin (1L<<12)
#define GCTileStipYOrigin (1L<<13)
#define GCFont (1L<<14)
#define GCSubwindowMode (1L<<15)
#define GCGraphicsExposures (1L<<16)
#define GCClipXOrigin (1L<<17)
#define GCClipYOrigin (1L<<18)
#define GCClipMask (1L<<19)
#define GCForeground (1L<<2)
#define GCDashOffset (1L<<20)
#define GCDashList (1L<<21)
#define GCArcMode (1L<<22)
#define GCBgground (1L<<3)
#define GCLineWidth (1L<<4)
#define GCLineStyle (1L<<5)
#define GCCapStyle (1L<<6)
#define GCJoinStyle (1L<<7)
#define GCFillStyle (1L<<8)
#define GCFillRule (1L<<9)
#define GCLastBit 22
#define FontChange 255

#define FontLeftToRight 0
#define FontRightToLeft 1

#define XYPixmap 1
#define ZPixmap 2

#define AllocNone 0
#define AllocAll 1

#define DoRed (1<<0)

```



```

#define DoGreen (1<<1)
#define DoBlue (1<<2)

#define CursorShape 0
#define TileShape 1
#define StippleShape 2

#define AutoRepeatModeOff 0
#define LedModeOff 0
#define AutoRepeatModeOn 1
#define LedModeOn 1
#define AutoRepeatModeDefault 2

#define KBKeyClickPercent (1L<<0)
#define KBBellPercent (1L<<1)
#define KBBellPitch (1L<<2)
#define KBBellDuration (1L<<3)
#define KBLed (1L<<4)
#define KBLedMode (1L<<5)
#define KBKey (1L<<6)
#define KBAutoRepeatMode (1L<<7)
#define MappingModifier 0
#define MappingSuccess 0
#define MappingBusy 1
#define MappingKeyboard 1
#define MappingFailed 2
#define MappingPointer 2

#define DisableScreenInterval 0
#define DisableScreenSaver 0
#define DontAllowExposures 0
#define DontPreferBlanking 0
#define AllowExposures 1
#define PreferBlanking 1
#define DefaultBlanking 2
#define DefaultExposures 2

#define ScreenSaverReset 0
#define ScreenSaverActive 1

#define HostInsert 0
#define HostDelete 1

#define DisableAccess 0
#define EnableAccess 1

#define StaticGray 0
#define GrayScale 1
#define StaticColor 2
#define PseudoColor 3
#define TrueColor 4
#define DirectColor 5

#define LSBFirst 0
#define MSBFirst 1

```

6.2.2 X11/XKBlib.h

```

#define _XKBLIB_H_
#define XkbNoteIndicatorMapChanges(o,n,w) ((o) -
>map_changes|=((n)->map_changes&(w)))
#define XkbNoteIndicatorStateChanges(o,n,w) ((o) -
>state_changes|=((n)->state_changes&(w)))
#define XkbLC_AllComposeControls (0xc0000000)
#define XkbLC_AllControls (0xc000001f)

```

```

#define XkbLC_ForceLatin1Lookup (1<<0)
#define XkbLC_ConsumeLookupMods (1<<1)
#define XkbLC_AlwaysConsumeShiftAndLock (1<<2)
#define XkbLC_ConsumeKeysOnComposeFail (1<<29)
#define XkbLC_IgnoreNewKeyboards (1<<3)
#define XkbLC_ComposeLED (1<<30)
#define XkbLC_BeepOnComposeFail (1<<31)
#define XkbLC_ControlFallback (1<<4)
#define XkbGetIndicatorMapChanges(d,x,c)
(XkbGetIndicatorMap((d),(c)->map_changes,x))
#define XkbChangeIndicatorMaps(d,x,c)
(XkbSetIndicatorMap((d),(c)->map_changes,x))
#define XkbOD_Success 0
#define XkbOD_BadLibraryVersion 1
#define XkbOD_ConnectionRefused 2
#define XkbOD_NonXkbServer 3
#define XkbOD_BadServerVersion 4
#define XkbGetControlsChanges(d,x,c) XkbGetControls(d,(c)-
>changed_ctrls,x)
#define XkbChangeControls(d,x,c) XkbSetControls(d,(c)-
>changed_ctrls,x)

typedef struct _XkbAnyEvent {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    unsigned int device;
} XkbAnyEvent;
typedef struct _XkbNewKeyboardNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    int old_device;
    int min_key_code;
    int max_key_code;
    int old_min_key_code;
    int old_max_key_code;
    unsigned int changed;
    char req_major;
    char req_minor;
} XkbNewKeyboardNotifyEvent;
typedef struct _XkbMapNotifyEvent {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int changed;
    unsigned int flags;
    int first_type;
    int num_types;
    KeyCode min_key_code;
    KeyCode max_key_code;
    KeyCode first_key_sym;
    KeyCode first_key_act;
    KeyCode first_key_behavior;
    KeyCode first_key_explicit;

```

```

        KeyCode first_modmap_key;
        KeyCode first_vmodmap_key;
        int num_key_syms;
        int num_key_acts;
        int num_key_behaviors;
        int num_key_explicit;
        int num_modmap_keys;
        int num_vmodmap_keys;
        unsigned int vmods;
    } XkbMapNotifyEvent;
typedef struct _XkbStateNotifyEvent {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int changed;
    int group;
    int base_group;
    int latched_group;
    int locked_group;
    unsigned int mods;
    unsigned int base_mods;
    unsigned int latched_mods;
    unsigned int locked_mods;
    int compat_state;
    unsigned char grab_mods;
    unsigned char compat_grab_mods;
    unsigned char lookup_mods;
    unsigned char compat_lookup_mods;
    int ptr_buttons;
    KeyCode keycode;
    char event_type;
    char req_major;
    char req_minor;
} XkbStateNotifyEvent;
typedef struct _XkbControlsNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int changed_ctrls;
    unsigned int enabled_ctrls;
    unsigned int enabled_ctrl_changes;
    int num_groups;
    KeyCode keycode;
    char event_type;
    char req_major;
    char req_minor;
} XkbControlsNotifyEvent;
typedef struct _XkbIndicatorNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int changed;
    unsigned int state;
} XkbIndicatorNotifyEvent;

```

```

typedef struct _XkbNamesNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int changed;
    int first_type;
    int num_types;
    int first_lvl;
    int num_lvls;
    int num_aliases;
    int num_radio_groups;
    unsigned int changed_vmods;
    unsigned int changed_groups;
    unsigned int changed_indicators;
    int first_key;
    int num_keys;
} XkbNamesNotifyEvent;
typedef struct _XkbCompatMapNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int changed_groups;
    int first_si;
    int num_si;
    int num_total_si;
} XkbCompatMapNotifyEvent;
typedef struct _XkbBellNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    int percent;
    int pitch;
    int duration;
    int bell_class;
    int bell_id;
    Atom name;
    Window window;
    int event_only;
} XkbBellNotifyEvent;
typedef struct _XkbActionMessage {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    KeyCode keycode;
    int press;
    int key_event_follows;
    int group;
    unsigned int mods;
    char message;
} XkbActionMessageEvent;

```

```

typedef struct _XkbAccessXNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    int detail;
    int keycode;
    int sk_delay;
    int debounce_delay;
} XkbAccessXNotifyEvent;
typedef struct _XkbExtensionDeviceNotify {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Time time;
    int xkb_type;
    int device;
    unsigned int reason;
    unsigned int supported;
    unsigned int unsupported;
    int first_btn;
    int num_btns;
    unsigned int leds_defined;
    unsigned int led_state;
    int led_class;
    int led_id;
} XkbExtensionDeviceNotifyEvent;
union _XkbEvent {
    int type;
    XkbAnyEvent any;
    XkbNewKeyboardNotifyEvent new_kbd;
    XkbMapNotifyEvent map;
    XkbStateNotifyEvent state;
    XkbControlsNotifyEvent ctrls;
    XkbIndicatorNotifyEvent indicators;
    XkbNamesNotifyEvent names;
    XkbCompatMapNotifyEvent compat;
    XkbBellNotifyEvent bell;
    XkbActionMessageEvent message;
    XkbAccessXNotifyEvent accessx;
    XkbExtensionDeviceNotifyEvent device;
    XEvent core;
};
typedef union _XkbEvent {
    int type;
    XkbAnyEvent any;
    XkbNewKeyboardNotifyEvent new_kbd;
    XkbMapNotifyEvent map;
    XkbStateNotifyEvent state;
    XkbControlsNotifyEvent ctrls;
    XkbIndicatorNotifyEvent indicators;
    XkbNamesNotifyEvent names;
    XkbCompatMapNotifyEvent compat;
    XkbBellNotifyEvent bell;
    XkbActionMessageEvent message;
    XkbAccessXNotifyEvent accessx;
    XkbExtensionDeviceNotifyEvent device;
    XEvent core;
} XkbEvent;
typedef struct _XkbKbdDpyState XkbKbdDpyStateRec;
typedef struct _XkbKbdDpyState *XkbKbdDpyStatePtr;
typedef Atom(*XkbInternAtomFunc) (Display *, const char *, int);

```

```

typedef char *(*XkbGetAtomNameFunc) (Display *, Atom);
extern int XkbAllocClientMap(XkbDescPtr, unsigned int, unsigned
int);
extern int XkbAllocCompatMap(XkbDescPtr, unsigned int, unsigned
int);
extern int XkbAllocControls(XkbDescPtr, unsigned int);
extern int XkbAllocIndicatorMaps(XkbDescPtr);
extern XkbDescPtr XkbAllocKeyboard(void);
extern int XkbAllocNames(XkbDescPtr, unsigned int, int, int);
extern int XkbAllocServerMap(XkbDescPtr, unsigned int, unsigned
int);
extern int XkbApplyCompatMapToKey(XkbDescPtr, KeyCode,
XkbChangesPtr);
extern int XkbBell(Display *, Window, int, Atom);
extern int XkbBellEvent(Display *, Window, int, Atom);
extern int XkbChangeEnabledControls(Display *, unsigned int,
unsigned int,
unsigned int);
extern int XkbChangeMap(Display *, XkbDescPtr, XkbMapChangesPtr);
extern int XkbChangeNames(Display *, XkbDescPtr, XkbNameChangesPtr);
extern int XkbChangeTypesOfKey(XkbDescPtr, int, int, unsigned int,
int *,
XkbMapChangesPtr);
extern int XkbComputeEffectiveMap(XkbDescPtr, XkbKeyTypePtr,
unsigned char *);
extern int XkbCopyKeyType(XkbKeyTypePtr, XkbKeyTypePtr);
extern int XkbCopyKeyTypes(XkbKeyTypePtr, XkbKeyTypePtr, int);
extern int XkbForceBell(Display *, int);
extern void XkbFreeClientMap(XkbDescPtr, unsigned int, int);
extern void XkbFreeCompatMap(XkbDescPtr, unsigned int, int);
extern void XkbFreeComponentList(XkbComponentListPtr);
extern void XkbFreeControls(XkbDescPtr, unsigned int, int);
extern void XkbFreeIndicatorMaps(XkbDescPtr);
extern void XkbFreeKeyboard(XkbDescPtr, unsigned int, int);
extern void XkbFreeNames(XkbDescPtr, unsigned int, int);
extern void XkbFreeServerMap(XkbDescPtr, unsigned int, int);
extern int XkbGetAutoRepeatRate(Display *, unsigned int, unsigned
int *,
unsigned int *);
extern int XkbGetCompatMap(Display *, unsigned int, XkbDescPtr);
extern int XkbGetControls(Display *, unsigned long int, XkbDescPtr);
extern int XkbGetIndicatorMap(Display *, unsigned long int,
XkbDescPtr);
extern int XkbGetIndicatorState(Display *, unsigned int, unsigned
int *);
extern int XkbGetKeyActions(Display *, unsigned int, unsigned int,
XkbDescPtr);
extern int XkbGetKeyBehaviors(Display *, unsigned int, unsigned int,
XkbDescPtr);
extern int XkbGetKeyExplicitComponents(Display *, unsigned int,
unsigned int, XkbDescPtr);
extern int XkbGetKeyModifierMap(Display *, unsigned int, unsigned
int,
XkbDescPtr);
extern int XkbGetKeySyms(Display *, unsigned int, unsigned int,
XkbDescPtr);
extern int XkbGetKeyTypes(Display *, unsigned int, unsigned int,
XkbDescPtr);
extern XkbDescPtr XkbGetKeyboard(Display *, unsigned int, unsigned
int);
extern XkbDescPtr XkbGetKeyboardByName(Display *, unsigned int,
XkbComponentNamesPtr, unsigned
int,
unsigned int, int);
extern XkbDescPtr XkbGetMap(Display *, unsigned int, unsigned int);

```

```

extern int XkbGetMapChanges(Display *, XkbDescPtr,
XkbMapChangesPtr);
extern int XkbGetNamedIndicator(Display *, Atom, int *, int *,
XkbIndicatorMapPtr, int *);
extern int XkbGetNames(Display *, unsigned int, XkbDescPtr);
extern int XkbGetPerClientControls(Display *, unsigned int *);
extern int XkbGetState(Display *, unsigned int, XkbStatePtr);
extern int XkbGetUpdatedMap(Display *, unsigned int, XkbDescPtr);
extern int XkbGetVirtualMods(Display *, unsigned int, XkbDescPtr);
extern unsigned int XkbGetXlibControls(Display *);
extern int XkbIgnoreExtension(int);
extern int XkbInitCanonicalKeyTypes(XkbDescPtr, unsigned int, int);
extern int XkbKeyTypesForCoreSymbols(XkbDescPtr, int, KeySym *,
unsigned int, int *, KeySym *);
extern KeySym XkbKeycodeToKeysym(Display *, unsigned int, int, int);
extern unsigned int XkbKeysymToModifiers(Display *, KeySym);
extern int XkbLatchGroup(Display *, unsigned int, unsigned int);
extern int XkbLatchModifiers(Display *, unsigned int, unsigned int,
unsigned int);
extern int XkbLibraryVersion(int *, int *);
extern XkbComponentListPtr XkbListComponents(Display *, unsigned
int,
XkbComponentNamesPtr, int *);
extern int XkbLockGroup(Display *, unsigned int, unsigned int);
extern int XkbLockModifiers(Display *, unsigned int, unsigned int,
unsigned int);
extern int XkbLookupKeyBinding(Display *, KeySym, unsigned int,
char *,
int, int *);
extern int XkbLookupKeySym(Display *, KeyCode, unsigned int,
unsigned int *, KeySym *);
extern void XkbNoteControlsChanges(XkbControlsChangesPtr,
XkbControlsNotifyEvent *, unsigned
int);
extern void XkbNoteMapChanges(XkbMapChangesPtr, XkbMapNotifyEvent
*,
unsigned int);
extern void XkbNoteNameChanges(XkbNameChangesPtr,
XkbNamesNotifyEvent *,
unsigned int);
extern Display *XkbOpenDisplay(char *, int *, int *, int *, int *,
int *);
extern int XkbQueryExtension(Display *, int *, int *, int *, int *,
int *);
extern int XkbRefreshKeyboardMapping(XkbMapNotifyEvent *);
extern XkbAction *XkbResizeKeyActions(XkbDescPtr, int, int);
extern KeySym *XkbResizeKeySyms(XkbDescPtr, int, int);
extern int XkbResizeKeyType(XkbDescPtr, int, int, int);
extern int XkbSelectEventDetails(Display *, unsigned int, unsigned
int,
unsigned long int, unsigned long int);
extern int XkbSelectEvents(Display *, unsigned int, unsigned int,
unsigned int);
extern void XkbSetAtomFuncs(XkbInternAtomFunc, XkbGetAtomNameFunc);
extern int XkbSetAutoRepeatRate(Display *, unsigned int, unsigned
int,
unsigned int);
extern int XkbSetAutoResetControls(Display *, unsigned int,
unsigned int *,
unsigned int *);
extern int XkbSetCompatMap(Display *, unsigned int, XkbDescPtr,
int);
extern int XkbSetControls(Display *, unsigned long int, XkbDescPtr);
extern int XkbSetDebuggingFlags(Display *, unsigned int, unsigned
int,
char *, unsigned int, unsigned int,

```

```

                                unsigned int *, unsigned int *);
extern int XkbSetDetectableAutoRepeat(Display *, int, int *);
extern int XkbSetIgnoreLockMods(Display *, unsigned int, unsigned
int,
                                unsigned int, unsigned int, unsigned
int);
extern int XkbSetIndicatorMap(Display *, unsigned long int,
XkbDescPtr);
extern int XkbSetMap(Display *, unsigned int, XkbDescPtr);
extern int XkbSetNamedIndicator(Display *, Atom, int, int, int,
                                XkbIndicatorMapPtr);
extern int XkbSetNames(Display *, unsigned int, unsigned int,
unsigned int,
                                XkbDescPtr);
extern int XkbSetPerClientControls(Display *, unsigned int,
                                unsigned int *);
extern int XkbSetServerInternalMods(Display *, unsigned int,
unsigned int,
                                unsigned int, unsigned int,
                                unsigned int);
extern unsigned int XkbSetXlibControls(Display *, unsigned int,
                                unsigned int);
extern char XkbToControl(char);
extern int XkbTranslateKeyCode(XkbDescPtr, KeyCode, unsigned int,
                                unsigned int *, KeySym *);
extern int XkbTranslateKeySym(Display *, KeySym *, unsigned int,
char *,
                                int, int *);
extern int XkbUpdateMapFromCore(XkbDescPtr, KeyCode, int, int,
KeySym *,
                                XkbChangesPtr);
extern int XkbUseExtension(Display *, int *, int *);
extern int XkbVirtualModsToReal(XkbDescPtr, unsigned int, unsigned
int *);

```

6.2.3 X11/Xarch.h

```

#define _XARCH_H
#define X_BIG_ENDIAN    BIG_ENDIAN
#define X_BYTE_ORDER    BYTE_ORDER
#define X_LITTLE_ENDIAN LITTLE_ENDIAN

```

6.2.4 X11/Xatom.h

```

#define XA_PRIMARY      ((Atom) 1)
#define XA_CUT_BUFFER1  ((Atom) 10)
#define XA_CUT_BUFFER2  ((Atom) 11)
#define XA_CUT_BUFFER3  ((Atom) 12)
#define XA_CUT_BUFFER4  ((Atom) 13)
#define XA_CUT_BUFFER5  ((Atom) 14)
#define XA_CUT_BUFFER6  ((Atom) 15)
#define XA_CUT_BUFFER7  ((Atom) 16)
#define XA_DRAWABLE     ((Atom) 17)
#define XA_FONT         ((Atom) 18)
#define XA_INTEGER      ((Atom) 19)
#define XA_SECONDARY    ((Atom) 2)
#define XA_PIXMAP       ((Atom) 20)
#define XA_POINT        ((Atom) 21)
#define XA_RECTANGLE    ((Atom) 22)
#define XA_RESOURCE_MANAGER ((Atom) 23)
#define XA_RGB_COLOR_MAP ((Atom) 24)
#define XA_RGB_BEST_MAP ((Atom) 25)
#define XA_RGB_BLUE_MAP ((Atom) 26)

```



```

#define XA_RGB_DEFAULT_MAP ((Atom) 27)
#define XA_RGB_GRAY_MAP ((Atom) 28)
#define XA_RGB_GREEN_MAP ((Atom) 29)
#define XA_ARC ((Atom) 3)
#define XA_RGB_RED_MAP ((Atom) 30)
#define XA_STRING ((Atom) 31)
#define XA_VISUALID ((Atom) 32)
#define XA_WINDOW ((Atom) 33)
#define XA_WM_COMMAND ((Atom) 34)
#define XA_WM_HINTS ((Atom) 35)
#define XA_WM_CLIENT_MACHINE ((Atom) 36)
#define XA_WM_ICON_NAME ((Atom) 37)
#define XA_WM_ICON_SIZE ((Atom) 38)
#define XA_WM_NAME ((Atom) 39)
#define XA_ATOM ((Atom) 4)
#define XA_WM_NORMAL_HINTS ((Atom) 40)
#define XA_WM_SIZE_HINTS ((Atom) 41)
#define XA_WM_ZOOM_HINTS ((Atom) 42)
#define XA_MIN_SPACE ((Atom) 43)
#define XA_NORM_SPACE ((Atom) 44)
#define XA_MAX_SPACE ((Atom) 45)
#define XA_END_SPACE ((Atom) 46)
#define XA_SUPERSCRIPT_X ((Atom) 47)
#define XA_SUPERSCRIPT_Y ((Atom) 48)
#define XA_SUBSCRIPT_X ((Atom) 49)
#define XA_BITMAP ((Atom) 5)
#define XA_SUBSCRIPT_Y ((Atom) 50)
#define XA_UNDERLINE_POSITION ((Atom) 51)
#define XA_UNDERLINE_THICKNESS ((Atom) 52)
#define XA_STRIKEOUT_ASCENT ((Atom) 53)
#define XA_STRIKEOUT_DESCENT ((Atom) 54)
#define XA_ITALIC_ANGLE ((Atom) 55)
#define XA_X_HEIGHT ((Atom) 56)
#define XA_QUAD_WIDTH ((Atom) 57)
#define XA_WEIGHT ((Atom) 58)
#define XA_POINT_SIZE ((Atom) 59)
#define XA_CARDINAL ((Atom) 6)
#define XA_RESOLUTION ((Atom) 60)
#define XA_COPYRIGHT ((Atom) 61)
#define XA_NOTICE ((Atom) 62)
#define XA_FONT_NAME ((Atom) 63)
#define XA_FAMILY_NAME ((Atom) 64)
#define XA_FULL_NAME ((Atom) 65)
#define XA_CAP_HEIGHT ((Atom) 66)
#define XA_WM_CLASS ((Atom) 67)
#define XA_LAST_PREDEFINED ((Atom) 68)
#define XA_WM_TRANSIENT_FOR ((Atom) 68)
#define XA_COLORMAP ((Atom) 7)
#define XA_CURSOR ((Atom) 8)
#define XA_CUT_BUFFER0 ((Atom) 9)
#define XATOM_H 1

```

6.2.5 X11/Xauth.h

```

#define FamilyLocalHost (252)
#define FamilyKrb5Principal (253)
#define FamilyNetname (254)
#define FamilyLocal (256)
#define FamilyWild (65535)
#define LOCK_SUCCESS 0
#define LOCK_ERROR 1
#define _Xauth_h 1
#define LOCK_TIMEOUT 2

typedef struct xauth {

```

```

    unsigned short family;
    unsigned short address_length;
    char *address;
    unsigned short number_length;
    char *number;
    unsigned short name_length;
    char *name;
    unsigned short data_length;
    char *data;
} Xauth;
extern void XauDisposeAuth(Xauth *);
extern char *XauFileName(void);
extern Xauth *XauGetBestAuthByAddr(unsigned int, unsigned int,
    const char *, unsigned int,
    const char *, int, char **,
    const int *);
extern Xauth *XauReadAuth(FILE *);

```

6.2.6 X11/Xcms.h

```

#define ClientWhitePointOfCCC(ccc)      (&(ccc)->clientWhitePt)
#define ScreenWhitePointOfCCC(ccc)      (&(ccc)->pPerScrnInfo->
>screenWhitePt)
#define DisplayOfCCC(ccc)               ((ccc)->dpy)
#define FunctionSetOfCCC(ccc)           ((ccc)->pPerScrnInfo->functionSet)
#define ScreenNumberOfCCC(ccc)          ((ccc)->screenNumber)
#define VisualOfCCC(ccc)                 ((ccc)->visual)
#define XcmsUndefinedFormat              (XcmsColorFormat) 0x00000000
#define XcmsCIEXYZFormat                  (XcmsColorFormat) 0x00000001
#define XcmsCIEuvYFormat                  (XcmsColorFormat) 0x00000002
#define XcmsCIExyYFormat                  (XcmsColorFormat) 0x00000003
#define XcmsCIELabFormat                  (XcmsColorFormat) 0x00000004
#define XcmsCIEluvFormat                  (XcmsColorFormat) 0x00000005
#define XcmsTekHVCFormat                  (XcmsColorFormat) 0x00000006
#define XcmsRGBFormat                     (XcmsColorFormat) 0x80000000
#define XcmsRGBiFormat                    (XcmsColorFormat) 0x80000001
#define XcmsFailure                       0
#define XcmsInitNone                      0x00
#define XcmsInitSuccess                   0x01
#define XcmsInitFailure                   0xff
#define XcmsSuccess                       1
#define XcmsSuccessWithCompression        2

typedef unsigned long int XcmsColorFormat;
typedef double XcmsFloat;
typedef struct {
    unsigned short red;
    unsigned short green;
    unsigned short blue;
} XcmsRGB;
typedef struct {
    XcmsFloat red;
    XcmsFloat green;
    XcmsFloat blue;
} XcmsRGBi;
typedef struct {
    XcmsFloat X;
    XcmsFloat Y;
    XcmsFloat Z;
} XcmsCIEXYZ;
typedef struct {
    XcmsFloat u_prime;
    XcmsFloat v_prime;
    XcmsFloat Y;
} XcmsCIEuvY;

```

```

typedef struct {
    XcmsFloat x;
    XcmsFloat y;
    XcmsFloat Y;
} XcmsCIExyY;
typedef struct {
    XcmsFloat L_star;
    XcmsFloat a_star;
    XcmsFloat b_star;
} XcmsCIELab;
typedef struct {
    XcmsFloat L_star;
    XcmsFloat u_star;
    XcmsFloat v_star;
} XcmsCIELuv;
typedef struct {
    XcmsFloat H;
    XcmsFloat V;
    XcmsFloat C;
} XcmsTekHVC;
typedef struct {
    XcmsFloat pad0;
    XcmsFloat pad1;
    XcmsFloat pad2;
    XcmsFloat pad3;
} XcmsPad;
typedef struct {
    union {
        XcmsRGB RGB;
        XcmsRGBi RGBi;
        XcmsCIEXYZ CIEXYZ;
        XcmsCIEuvY CIEuvY;
        XcmsCIExyY CIExyY;
        XcmsCIELab CIELab;
        XcmsCIELuv CIELuv;
        XcmsTekHVC TekHVC;
        XcmsPad Pad;
    } spec;
    unsigned long int pixel;
    XcmsColorFormat format;
} XcmsColor;
typedef struct _XcmsPerScrnInfo {
    XcmsColor screenWhitePt;
    XPointer functionSet;
    XPointer screenData;
    unsigned char state;
    char pad[3];
} XcmsPerScrnInfo;
typedef struct _XcmsCCC {
    Display *dpy;
    int screenNumber;
    Visual *visual;
    XcmsColor clientWhitePt;
    XcmsCompressionProc gamutCompProc;
    XPointer gamutCompClientData;
    XcmsWhiteAdjustProc whitePtAdjProc;
    XPointer whitePtAdjClientData;
    XcmsPerScrnInfo *pPerScrnInfo;
} XcmsCCCRec;
typedef struct _XcmsColorSpace {
    char *prefix;
    XcmsColorFormat id;
    XcmsParseStringProc parseString;
    XcmsFuncListPtr to_CIEXYZ;
    XcmsFuncListPtr from_CIEXYZ;
    int inverse_flag;

```

```

} XcmsColorSpace;
typedef struct _XcmsFunctionSet {
    XcmsColorSpace **DDColorSpaces;
    XcmsScreenInitProc screenInitProc;
    XcmsScreenFreeProc screenFreeProc;
} XcmsFunctionSet;
typedef struct _XcmsCCC *XcmsCCC;
typedef int (*XcmsCompressionProc) (XcmsCCC, XcmsColor *, unsigned
int,
                                unsigned int, int *);
typedef int (*XcmsWhiteAdjustProc) (XcmsCCC, XcmsColor *, XcmsColor
*,
                                XcmsColorFormat, XcmsColor *,
                                unsigned int, int *);
typedef int (*XcmsScreenInitProc) (Display *, int, XcmsPerScrnInfo
*);
typedef void (*XcmsScreenFreeProc) (XPointer);
typedef int (*XcmsDIConversionProc) (XcmsCCC, XcmsColor *,
XcmsColor *,
                                unsigned int);
typedef XcmsDIConversionProc XcmsConversionProc;
typedef int (*XcmsParseStringProc) (char *, XcmsColor*);
typedef XcmsConversionProc *XcmsFuncListPtr;
typedef int (*XcmsDDConversionProc) (XcmsCCC, XcmsColor *, unsigned
int,
                                int *);

extern int XcmsAddColorSpace(XcmsColorSpace *);
extern int XcmsAddFunctionSet(XcmsFunctionSet *);
extern int XcmsAllocColor(Display *, Colormap, XcmsColor *,
XcmsColorFormat);
extern int XcmsAllocNamedColor(Display *, Colormap, const char *,
XcmsColor *, XcmsColor *,
XcmsColorFormat);
extern XcmsCCC XcmsCCCOFColormap(Display *, Colormap);
extern int XcmsCIELabClipL(XcmsCCC, XcmsColor *, unsigned int,
unsigned int, int *);
extern int XcmsCIELabClipLab(XcmsCCC, XcmsColor *, unsigned int,
unsigned int, int *);
extern int XcmsCIELabClipab(XcmsCCC, XcmsColor *, unsigned int,
unsigned int, int *);
extern int XcmsCIELabQueryMaxC(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsCIELabQueryMaxL(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsCIELabQueryMaxLC(XcmsCCC, XcmsFloat, XcmsColor *);
extern int XcmsCIELabQueryMinL(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsCIELabToCIEXYZ(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIELabWhiteShiftColors(XcmsCCC, XcmsColor *,
XcmsColor *,
XcmsColorFormat, XcmsColor *,
unsigned int, int *);
extern int XcmsCIELuvClipL(XcmsCCC, XcmsColor *, unsigned int,
unsigned int, int *);
extern int XcmsCIELuvClipLuv(XcmsCCC, XcmsColor *, unsigned int,
unsigned int, int *);
extern int XcmsCIELuvClipuv(XcmsCCC, XcmsColor *, unsigned int,
unsigned int, int *);
extern int XcmsCIELuvQueryMaxC(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsCIELuvQueryMaxL(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsCIELuvQueryMaxLC(XcmsCCC, XcmsFloat, XcmsColor *);

```

```

extern int XcmsCIELuvQueryMinL(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsCIELuvToCIEuvY(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIELuvWhiteShiftColors(XcmsCCC, XcmsColor *,
XcmsColor *,
XcmsColorFormat, XcmsColor *,
unsigned int, int *);
extern int XcmsCIEXYZToCIELab(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIEXYZToCIEuvY(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIEXYZToCIExyY(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIEXYZToRGBi(XcmsCCC, XcmsColor *, unsigned int, int
*);
extern int XcmsCIEuvYToCIELuv(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIEuvYToCIEXYZ(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIEuvYToTekHVC(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern int XcmsCIExyYToCIEXYZ(XcmsCCC, XcmsColor *, XcmsColor *,
unsigned int);
extern XcmsColor *XcmsClientWhitePointOfCCC(XcmsCCC);
extern int XcmsConvertColors(XcmsCCC, XcmsColor *, unsigned int,
XcmsColorFormat, int *);
extern XcmsCCC XcmsCreateCCC(Display *, int, Visual *, XcmsColor *,
XcmsCompressionProc, XPointer,
XcmsWhiteAdjustProc, XPointer);
extern XcmsCCC XcmsDefaultCCC(Display *, int);
extern Display *XcmsDisplayOfCCC(XcmsCCC);
extern XcmsColorFormat XcmsFormatOfPrefix(char *);
extern void XcmsFreeCCC(XcmsCCC);
extern int XcmsLookupColor(Display *, Colormap, const char *,
XcmsColor *,
XcmsColor *, XcmsColorFormat);
extern char *XcmsPrefixOfFormat(XcmsColorFormat);
extern int XcmsQueryBlack(XcmsCCC, XcmsColorFormat, XcmsColor *);
extern int XcmsQueryBlue(XcmsCCC, XcmsColorFormat, XcmsColor *);
extern int XcmsQueryColor(Display *, Colormap, XcmsColor *,
XcmsColorFormat);
extern int XcmsQueryColors(Display *, Colormap, XcmsColor *,
unsigned int,
XcmsColorFormat);
extern int XcmsQueryGreen(XcmsCCC, XcmsColorFormat, XcmsColor *);
extern int XcmsQueryRed(XcmsCCC, XcmsColorFormat, XcmsColor *);
extern int XcmsQueryWhite(XcmsCCC, XcmsColorFormat, XcmsColor *);
extern int XcmsRGBToRGBi(XcmsCCC, XcmsColor *, unsigned int, int
*);
extern int XcmsRGBiToCIEXYZ(XcmsCCC, XcmsColor *, unsigned int, int
*);
extern int XcmsRGBiToRGB(XcmsCCC, XcmsColor *, unsigned int, int
*);
extern int XcmsScreenNumberOfCCC(XcmsCCC);
extern XcmsColor *XcmsScreenWhitePointOfCCC(XcmsCCC);
extern XcmsCCC XcmsSetCCCOfColormap(Display *, Colormap, XcmsCCC);
extern XcmsCompressionProc XcmsSetCompressionProc(XcmsCCC,
XcmsCompressionProc,
XPointer);
extern XcmsWhiteAdjustProc XcmsSetWhiteAdjustProc(XcmsCCC,
XcmsWhiteAdjustProc,
XPointer);
extern int XcmsSetWhitePoint(XcmsCCC, XcmsColor *);
extern int XcmsStoreColor(Display *, Colormap, XcmsColor *);

```

```

extern int XcmsStoreColors(Display *, Colormap, XcmsColor *,
                           unsigned int,
                           int *);
extern int XcmsTekHVCClipC(XcmsCCC, XcmsColor *, unsigned int,
                           unsigned int, int *);
extern int XcmsTekHVCClipV(XcmsCCC, XcmsColor *, unsigned int,
                           unsigned int, int *);
extern int XcmsTekHVCClipVC(XcmsCCC, XcmsColor *, unsigned int,
                            unsigned int, int *);
extern int XcmsTekHVCQueryMaxC(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsTekHVCQueryMaxV(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsTekHVCQueryMaxVC(XcmsCCC, XcmsFloat, XcmsColor *);
extern int XcmsTekHVCQueryMaxVSamples(XcmsCCC, XcmsFloat,
XcmsColor *,
                                   unsigned int);
extern int XcmsTekHVCQueryMinV(XcmsCCC, XcmsFloat, XcmsFloat,
XcmsColor *);
extern int XcmsTekHVCToCIEuvY(XcmsCCC, XcmsColor *, XcmsColor *,
                              unsigned int);
extern int XcmsTekHVCWhiteShiftColors(XcmsCCC, XcmsColor *,
XcmsColor *,
                                   XcmsColorFormat, XcmsColor *,
                                   unsigned int, int *);
extern Visual *XcmsVisualOfCCC(XcmsCCC);

```

6.2.7 X11/Xfuncproto.h

```

#define NARROWPROTO
#ifdef __cplusplus
#define _XFUNCPROTOBEGIN extern "C" {
#else
#define _XFUNCPROTOBEGIN
#endif
#ifdef __cplusplus
#define _XFUNCPROTOEND }
#else
#define _XFUNCPROTOEND
#endif
#define NeedFunctionPrototypes 1
#define NeedNestedPrototypes 1
#define NeedVarargsPrototypes 1
#define NeedWidePrototypes 1
#define FUNCPROTO 15
#define _Xconst const
#define _X_INLINE inline
#define _X_SENTINEL(x) __attribute__((__sentinel__(x)))
#define _X_DEPRECATED __attribute__((deprecated))
#define _X_EXPORT __attribute__((visibility("default")))
#define _X_HIDDEN __attribute__((visibility("hidden")))
#define _X_INTERNAL __attribute__((visibility("internal")))
#define _X_ATTRIBUTE_PRINTF(x,y)
__attribute__((__format__(__printf__,x,y)))
#define _X_UNLIKELY(x) __builtin_expect(!!(x), 0)
#define _X_LIKELY(x) __builtin_expect(!!(x), 1)

```

6.2.8 X11/Xfuncs.h

```

#define _XFUNCS_H_

```

6.2.9 X11/Xlib.h

```

struct _XDisplay;

typedef char *XPointer;
typedef struct {
    unsigned long int pixel;
    unsigned short red;
    unsigned short green;
    unsigned short blue;
    char flags;
    char pad;
} XColor;
typedef struct _XDisplay Display;
typedef struct {
    int type;
    Display *display;
    XID resourceid;
    unsigned long int serial;
    unsigned char error_code;
    unsigned char request_code;
    unsigned char minor_code;
} XErrorEvent;
typedef struct _XIM *XIM;
typedef struct _XIC *XIC;
typedef void *XVaNestedList;
typedef unsigned long int XIMFeedback;
typedef struct _XIMText {
    unsigned short length;
    XIMFeedback *feedback;
    int encoding_is_wchar;
    union {
        char *multi_byte;
        wchar_t *wide_char;
    } string;
} XIMText;
typedef unsigned long int XIMPreeditState;
typedef struct _XIMPreeditStateNotifyCallbackStruct {
    XIMPreeditState state;
} XIMPreeditStateNotifyCallbackStruct;
typedef unsigned long int XIMResetState;
typedef unsigned long int XIMStringConversionFeedback;
typedef struct _XIMStringConversionText {
    unsigned short length;
    XIMStringConversionFeedback *feedback;
    int encoding_is_wchar;
    union {
        char *mbs;
        wchar_t *wcs;
    } string;
} XIMStringConversionText;
typedef unsigned short XIMStringConversionPosition;
typedef unsigned short XIMStringConversionType;
typedef unsigned short XIMStringConversionOperation;
typedef struct _XIMStringConversionCallbackStruct {
    XIMStringConversionPosition position;
    XIMCaretDirection direction;
    XIMStringConversionOperation operation;
    unsigned short factor;
    XIMStringConversionText *text;
} XIMStringConversionCallbackStruct;
typedef struct _XIMPreeditDrawCallbackStruct {
    int caret;
    int chg_first;
    int chg_length;

```

```

        XIMText *text;
    } XIMPreeditDrawCallbackStruct;
typedef struct _XIMPreeditCaretCallbackStruct {
    int position;
    XIMCaretDirection direction;
    XIMCaretStyle style;
} XIMPreeditCaretCallbackStruct;
typedef struct _XIMStatusDrawCallbackStruct {
    XIMStatusDataType type;
    union {
        XIMText *text;
        Pixmap bitmap;
    } data;
} XIMStatusDrawCallbackStruct;
typedef struct _XIMHotKeyTrigger {
    KeySym keysym;
    int modifier;
    int modifier_mask;
} XIMHotKeyTrigger;
typedef struct _XIMHotKeyTriggers {
    int num_hot_key;
    XIMHotKeyTrigger *key;
} XIMHotKeyTriggers;
typedef unsigned long int XIMHotKeyState;
typedef struct {
    unsigned short count_values;
    char **supported_values;
} XIMValuesList;
typedef void (*XIMProc) (XIM, XPointer, XPointer);
typedef int (*XICProc) (XIC, XPointer, XPointer);
typedef enum {
    XIMForwardChar = 0,
    XIMBackwardChar = 1,
    XIMForwardWord = 2,
    XIMBackwardWord = 3,
    XIMCaretUp = 4,
    XIMCaretDown = 5,
    XIMNextLine = 6,
    XIMPreviousLine = 7,
    XIMLineStart = 8,
    XIMLineEnd = 9,
    XIMAbsolutePosition = 10,
    XIMDontChange = 11
} XIMCaretDirection;
typedef enum {
    XIMIsInvisible = 0,
    XIMIsPrimary = 1,
    XIMIsSecondary = 2
} XIMCaretStyle;
typedef enum {
    XIMTextType = 0,
    XIMBitmapType = 1
} XIMStatusDataType;
typedef int (*XErrorHandler) (Display *, XErrorEvent *);
typedef int (*XIOErrorHandler) (Display *);
typedef void (*XConnectionWatchProc) (Display *, XPointer, int, int,
                                      XPointer *);

#define XNArea "area"
#define XNAreaNeeded "areaNeeded"
#define XNBackground "background"
#define XNBackgroundPixmap "backgroundPixmap"
#define XNBaseFontName "baseFontName"
#define XNClientWindow "clientWindow"
#define XNColormap "colorMap"
#define XNContextualDrawing "contextualDrawing"

```



```

#define XNCursor          "cursor"
#define XNDefaultString  "defaultString"
#define XNDestroyCallback "destroyCallback"
#define                    XNDirectionalDependentDrawing
"directionalDependentDrawing"
#define XNFilterEvents   "filterEvents"
#define XNFocusWindow    "focusWindow"
#define XNFontInfo       "fontInfo"
#define XNFontSet        "fontSet"
#define XNForeground     "foreground"
#define XNGeometryCallback "geometryCallback"
#define XNHotKey         "hotKey"
#define XNHotKeyState    "hotKeyState"
#define XNInputStyle     "inputStyle"
#define XNLineSpace      "lineSpace"
#define XNMissingCharSet "missingCharSet"
#define XNOMAutomatic    "omAutomatic"
#define XNOrientation    "orientation"
#define XNPreeditAttributes "preeditAttributes"
#define XNPreeditCaretCallback "preeditCaretCallback"
#define XNPreeditDoneCallback "preeditDoneCallback"
#define XNPreeditDrawCallback "preeditDrawCallback"
#define XNPreeditStartCallback "preeditStartCallback"
#define XNPreeditState    "preeditState"
#define                    XNPreeditStateNotifyCallback
"preeditStateNotifyCallback"
#define XNQueryICValuesList "queryICValuesList"
#define XNQueryIMValuesList "queryIMValuesList"
#define XNQueryInputStyle   "queryInputStyle"
#define XNQueryOrientation  "queryOrientation"
#define XNR6PreeditCallback "r6PreeditCallback"
#define XNRequiredCharSet   "requiredCharSet"
#define XNResetState        "resetState"
#define XNResourceClass     "resourceClass"
#define XNResourceName      "resourceName"
#define XNSeparatorofNestedList "separatorofNestedList"
#define XNSpotLocation      "spotLocation"
#define XNStatusAttributes  "statusAttributes"
#define XNStatusDoneCallback "statusDoneCallback"
#define XNStatusDrawCallback "statusDrawCallback"
#define XNStatusStartCallback "statusStartCallback"
#define XNStdColorMap       "stdColorMap"
#define XNStringConversion  "stringConversion"
#define XNStringConversionCallback "stringConversionCallback"
#define XNVisiblePosition   "visiblePosition"
#define XNVaNestedList      "XNVaNestedList"
#define ScreenOfDisplay(dpy, scr) (( (_XPrivDisplay) dpy) -
>screens[scr])
#define BitmapBitOrder(dpy) ((( _XPrivDisplay) dpy) -
>bitmap_bit_order)
#define BitmapPad(dpy) ((( _XPrivDisplay) dpy) ->bitmap_pad)
#define BitmapUnit(dpy) ((( _XPrivDisplay) dpy) ->bitmap_unit)
#define ImageByteOrder(dpy) ((( _XPrivDisplay) dpy) ->byte_order)
#define DefaultScreen(dpy) ((( _XPrivDisplay) dpy) -
>default_screen)
#define DisplayString(dpy) ((( _XPrivDisplay) dpy) ->display_name)
#define ConnectionNumber(dpy) ((( _XPrivDisplay) dpy) ->fd)
#define LastKnownRequestProcessed(dpy) ((( _XPrivDisplay) dpy) -
>last_request_read)
#define ScreenCount(dpy) ((( _XPrivDisplay) dpy) ->ncreens)
#define ProtocolVersion(dpy) ((( _XPrivDisplay) dpy) -
>proto_major_version)
#define ProtocolRevision(dpy) ((( _XPrivDisplay) dpy) -
>proto_minor_version)
#define QLength(dpy) ((( _XPrivDisplay) dpy) ->qlen)
#define VendorRelease(dpy) ((( _XPrivDisplay) dpy) ->release)

```

```

#define NextRequest(dpy)      (((_XPrivDisplay)dpy)->request+1)
#define ServerVendor(dpy)    (((_XPrivDisplay)dpy)->vendor)
#define XAllocID(dpy)        ((*((_XPrivDisplay)dpy)-
>resource_alloc)((dpy)))
#define DoesBackingStore(s)  ((s)->backing_store)
#define BlackPixelOfScreen(s) ((s)->black_pixel)
#define DefaultColormapOfScreen(s) ((s)->cmap)
#define DefaultGCOfScreen(s) ((s)->default_gc)
#define DisplayOfScreen(s)   ((s)->display)
#define HeightOfScreen(s)    ((s)->height)
#define MaxCmapsOfScreen(s)  ((s)->max_maps)
#define HeightMMOfScreen(s)  ((s)->mheight)
#define MinCmapsOfScreen(s)  ((s)->min_maps)
#define WidthMMOfScreen(s)   ((s)->mwidth)
#define RootWindowOfScreen(s) ((s)->root)
#define DefaultDepthOfScreen(s) ((s)->root_depth)
#define PlanesOfScreen(s)    ((s)->root_depth)
#define EventMaskOfScreen(s) ((s)->root_input_mask)
#define DefaultVisualOfScreen(s) ((s)->root_visual)
#define DoesSaveUnders(s)    ((s)->save_unders)
#define WhitePixelOfScreen(s) ((s)->white_pixel)
#define WidthOfScreen(s)     ((s)->width)
#define AllPlanes            ((unsigned long)~0L)
#define XIMStringConversionLeftEdge (0x00000001)
#define XIMStringConversionRightEdge (0x00000002)
#define XIMStringConversionTopEdge (0x00000004)
#define XIMStringConversionBottomEdge (0x00000008)
#define XIMStringConversionConcealed (0x00000010)
#define XIMStringConversionWrapped (0x00000020)
#define XIMStringConversionBuffer (0x0001)
#define XIMStringConversionSubstitution (0x0001)
#define XIMHotKeyStateON (0x0001L)
#define XIMStringConversionLine (0x0002)
#define XIMStringConversionRetrieval (0x0002)
#define XIMHotKeyStateOFF (0x0002L)
#define XIMStringConversionWord (0x0003)
#define XIMStringConversionChar (0x0004)
#define XIMPreeditDisable (1L<<1)
#define XIMPreserveState (1L<<1)
#define XIMUnderline (1L<<1)
#define XIMVisibleToCenter (1L<<10)
#define XIMHighlight (1L<<2)
#define XIMPrimary (1L<<5)
#define XIMSecondary (1L<<6)
#define XIMTertiary (1L<<7)
#define XIMVisibleToForward (1L<<8)
#define XIMVisibleToBackward (1L<<9)
#define DisplayCells(dpy,scr) (DefaultVisual(dpy,scr)-
>map_entries)
#define CellsOfScreen(s) (DefaultVisualOfScreen((s))-
>map_entries)
#define DefaultRootWindow(dpy) (ScreenOfDisplay(dpy,DefaultScreen(dpy))->root)
#define BlackPixel(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>black_pixel)
#define DefaultColormap(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>cmap)
#define DefaultGC(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>default_gc)
#define DisplayHeight(dpy,scr) (ScreenOfDisplay(dpy,scr)->height)
#define DisplayHeightMM(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>mheight)
#define DisplayWidthMM(dpy,scr) (ScreenOfDisplay(dpy,scr)->mwidth)
#define RootWindow(dpy,scr) (ScreenOfDisplay(dpy,scr)->root)
#define DefaultDepth(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>root_depth)

```

```

#define DisplayPlanes(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>root_depth)
#define DefaultVisual(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>root_visual)
#define WhitePixel(dpy,scr) (ScreenOfDisplay(dpy,scr)-
>white_pixel)
#define DisplayWidth(dpy,scr) (ScreenOfDisplay(dpy,scr)->width)
#define XBufferOverflow -1
#define False 0
#define QueuedAlready 0
#define XIMPreeditUnKnown 0L
#define XIMPreeditArea 0x0001L
#define XIMPreeditCallbacks 0x0002L
#define XIMPreeditPosition 0x0004L
#define XIMPreeditNothing 0x0008L
#define XIMPreeditNone 0x0010L
#define XIMStatusArea 0x0100L
#define XIMStatusCallbacks 0x0200L
#define XIMStatusNothing 0x0400L
#define XIMStatusNone 0x0800L
#define QueuedAfterReading 1
#define True 1
#define XLookupNone 1
#define _XLIB_H_ 1
#define XIMInitialState 1L
#define XIMPreeditEnable 1L
#define XIMReverse 1L
#define QueuedAfterFlush 2
#define XLookupChars 2
#define XLookupKeySym 3
#define XLookupBoth 4
#define XlibSpecificationRelease 6
#define Bool int
#define Status int
#define DefaultScreenOfDisplay(dpy)
ScreenOfDisplay(dpy,DefaultScreen(dpy))

typedef struct _XExtData {
    int number;
    struct _XExtData *next;
    int (*free_private) (struct _XExtData * extension);
    XPointer private_data;
} XExtData;
typedef struct {
    int extension;
    int major_opcode;
    int first_event;
    int first_error;
} XExtCodes;

typedef struct {
    int depth;
    int bits_per_pixel;
    int scanline_pad;
} XPixmapFormatValues;

typedef struct {
    int function;
    unsigned long int plane_mask;
    unsigned long int foreground;
    unsigned long int background;
    int line_width;
    int line_style;
    int cap_style;
    int join_style;
    int fill_style;

```

```

    int fill_rule;
    int arc_mode;
    Pixmap tile;
    Pixmap stipple;
    int ts_x_origin;
    int ts_y_origin;
    Font font;
    int subwindow_mode;
    int graphics_exposures;
    int clip_x_origin;
    int clip_y_origin;
    Pixmap clip_mask;
    int dash_offset;
    char dashes;
} XGCValues;

struct _XGC;
typedef struct _XGC *GC;

typedef struct {
    XExtData *ext_data;
    VisualID visualid;
#if defined(__cplusplus) || defined(c_plusplus)
    int c_class;
#else
    int class;
#endif
    unsigned long int red_mask;
    unsigned long int green_mask;
    unsigned long int blue_mask;
    int bits_per_rgb;
    int map_entries;
} Visual;

typedef struct {
    int depth;
    int nvisuals;
    Visual *visuals;
} Depth;

typedef struct {
    XExtData *ext_data;
    struct _XDisplay *display;
    Window root;
    int width;
    int height;
    int mwidth;
    int mheight;
    int ndepths;
    Depth *depths;
    int root_depth;
    Visual *root_visual;
    GC default_gc;
    Colormap cmap;
    unsigned long int white_pixel;
    unsigned long int black_pixel;
    int max_maps;
    int min_maps;
    int backing_store;
    int save_unders;
    long int root_input_mask;
} Screen;

typedef struct {
    XExtData *ext_data;
    int depth;

```

```

    int bits_per_pixel;
    int scanline_pad;
} ScreenFormat;

typedef struct {
    Pixmap background_pixmap;
    unsigned long int background_pixel;
    Pixmap border_pixmap;
    unsigned long int border_pixel;
    int bit_gravity;
    int win_gravity;
    int backing_store;
    unsigned long int backing_planes;
    unsigned long int backing_pixel;
    int save_under;
    long int event_mask;
    long int do_not_propagate_mask;
    int override_redirect;
    Colormap colormap;
    Cursor cursor;
} XSetWindowAttributes;
typedef struct {
    int x;
    int y;
    int width;
    int height;
    int border_width;
    int depth;
    Visual *visual;
    Window root;
#if defined(__cplusplus) || defined(c_plusplus)
    int c_class;
#else
    int class;
#endif
    int bit_gravity;
    int win_gravity;
    int backing_store;
    unsigned long int backing_planes;
    unsigned long int backing_pixel;
    int save_under;
    Colormap colormap;
    int map_installed;
    int map_state;
    long int all_event_masks;
    long int your_event_mask;
    long int do_not_propagate_mask;
    int override_redirect;
    Screen *screen;
} XWindowAttributes;

typedef struct {
    int family;
    int length;
    char *address;
} XHostAddress;

struct funcs {
    struct _XImage *(*create_image) (void);
    int (*destroy_image) (void);
    unsigned long int (*get_pixel) (void);
    int (*put_pixel) (void);
    struct _XImage *(*sub_image) (void);
    int (*add_pixel) (void);
};

typedef struct _XImage {

```

```

    int width;
    int height;
    int xoffset;
    int format;
    char *data;
    int byte_order;
    int bitmap_unit;
    int bitmap_bit_order;
    int bitmap_pad;
    int depth;
    int bytes_per_line;
    int bits_per_pixel;
    unsigned long int red_mask;
    unsigned long int green_mask;
    unsigned long int blue_mask;
    XPointer obdata;
    struct funcs f;
} XImage;

typedef struct {
    int x;
    int y;
    int width;
    int height;
    int border_width;
    Window sibling;
    int stack_mode;
} XWindowChanges;

typedef struct {
    short x1;
    short y1;
    short x2;
    short y2;
} XSegment;
typedef struct {
    short x;
    short y;
} XPoint;
typedef struct {
    short x;
    short y;
    unsigned short width;
    unsigned short height;
} XRectangle;
typedef struct {
    short x;
    short y;
    unsigned short width;
    unsigned short height;
    short angle1;
    short angle2;
} XArc;

typedef struct {
    int key_click_percent;
    int bell_percent;
    int bell_pitch;
    int bell_duration;
    int led;
    int led_mode;
    int key;
    int auto_repeat_mode;
} XKeyboardControl;

typedef struct {

```

```

    int key_click_percent;
    int bell_percent;
    unsigned int bell_pitch;
    unsigned int bell_duration;
    unsigned long int led_mask;
    int global_auto_repeat;
    char auto_repeats[32];
} XKeyboardState;

typedef struct {
    Time time;
    short x;
    short y;
} XTimeCoord;

typedef struct {
    int max_keypermod;
    KeyCode *modifiermap;
} XModifierKeymap;

typedef struct {
    XExtData *ext_data;
    struct _XPrivate *private1;
    int fd;
    int private2;
    int proto_major_version;
    int proto_minor_version;
    char *vendor;
    XID private3;
    XID private4;
    XID private5;
    int private6;
    XID(*resource_alloc) (struct _XDisplay *);
    int byte_order;
    int bitmap_unit;
    int bitmap_pad;
    int bitmap_bit_order;
    int nformats;
    ScreenFormat *pixmap_format;
    int private8;
    int release;
    struct _XPrivate *private9;
    struct _XPrivate *private10;
    int qlen;
    unsigned long int last_request_read;
    unsigned long int request;
    XPointer private11;
    XPointer private12;
    XPointer private13;
    XPointer private14;
    unsigned int max_request_size;
    struct _XrmHashBucketRec *db;
    int (*private15) (struct _XDisplay *);
    char *display_name;
    int default_screen;
    int nscreens;
    Screen *screens;
    unsigned long int motion_buffer;
    unsigned long int private16;
    int min_keycode;
    int max_keycode;
    XPointer private17;
    XPointer private18;
    int private19;
    char *xdefaults;
} *_XPrivDisplay;

```

```

typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    unsigned int keycode;
    int same_screen;
} XKeyEvent;
typedef XKeyEvent XKeyPressedEvent;
typedef XKeyEvent XKeyReleasedEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    unsigned int button;
    int same_screen;
} XButtonEvent;
typedef XButtonEvent XButtonPressedEvent;
typedef XButtonEvent XButtonReleasedEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    char is_hint;
    int same_screen;
} XMotionEvent;
typedef XMotionEvent XPointerMovedEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Window root;
    Window subwindow;
    Time time;

```



```

    int x;
    int y;
    int x_root;
    int y_root;
    int mode;
    int detail;
    int same_screen;
    int focus;
    unsigned int state;
} XCrossingEvent;
typedef XCrossingEvent XEnterWindowEvent;
typedef XCrossingEvent XLeaveWindowEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    int mode;
    int detail;
} XFocusChangeEvent;
typedef XFocusChangeEvent XFocusInEvent;
typedef XFocusChangeEvent XFocusOutEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    char key_vector[32];
} XKeyEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    int x;
    int y;
    int width;
    int height;
    int count;
} XExposeEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Drawable drawable;
    int x;
    int y;
    int width;
    int height;
    int count;
    int major_code;
    int minor_code;
} XGraphicsExposeEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Drawable drawable;
    int major_code;
    int minor_code;
} XNoExposeEvent;

```

```

typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    int state;
} XVisibilityEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window parent;
    Window window;
    int x;
    int y;
    int width;
    int height;
    int border_width;
    int override_redirect;
} XCreateWindowEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
} XDestroyWindowEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
    int from_configure;
} XUnmapEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
    int override_redirect;
} XMapEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window parent;
    Window window;
} XMapRequestEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
    Window parent;
    int x;
    int y;

```

```

        int override_redirect;
    } XReparentEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
    int x;
    int y;
    int width;
    int height;
    int border_width;
    Window above;
    int override_redirect;
} XConfigureEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
    int x;
    int y;
} XGravityEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    int width;
    int height;
} XResizeRequestEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window parent;
    Window window;
    int x;
    int y;
    int width;
    int height;
    int border_width;
    Window above;
    int detail;
    unsigned long int value_mask;
} XConfigureRequestEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window event;
    Window window;
    int place;
} XCirculateEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;

```

```

        Window parent;
        Window window;
        int place;
    } XCirculateRequestEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Atom atom;
    Time time;
    int state;
} XPropertyEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Atom selection;
    Time time;
} XSelectionClearEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window owner;
    Window requestor;
    Atom selection;
    Atom target;
    Atom property;
    Time time;
} XSelectionRequestEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window requestor;
    Atom selection;
    Atom target;
    Atom property;
    Time time;
} XSelectionEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    Colormap colormap;
#ifdef __cplusplus || defined(cplusplus)
    int c_new;
#else
    int new;
#endif
    int state;
} XColormapEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;

```

```

    Atom message_type;
    int format;
    union {
        char b[20];
        short s[10];
        long int l[5];
    } data;
} XClientMessageEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
    int request;
    int first_keycode;
    int count;
} XMappingEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Window window;
} XAnyEvent;

union _XEvent {
    int type;
    XAnyEvent xany;
    XKeyEvent xkey;
    XButtonEvent xbutton;
    XMotionEvent xmotion;
    XCrossingEvent xcrossing;
    XFocusChangeEvent xfocus;
    XExposeEvent xexpose;
    XGraphicsExposeEvent xgraphicsexpose;
    XNoExposeEvent xnoexpose;
    XVisibilityEvent xvisibility;
    XCreateWindowEvent xcreatewindow;
    XDestroyWindowEvent xdestroywindow;
    XUnmapEvent xunmap;
    XMapEvent xmap;
    XMapRequestEvent xmaprequest;
    XReparentEvent xreparent;
    XConfigureEvent xconfigure;
    XGravityEvent xgravity;
    XResizeRequestEvent xresizerequest;
    XConfigureRequestEvent xconfigurerequest;
    XCirculateEvent xcirculate;
    XCirculateRequestEvent xcirculaterequest;
    XPropertyEvent xproperty;
    XSelectionClearEvent xselectionclear;
    XSelectionRequestEvent xselectionrequest;
    XSelectionEvent xselection;
    XColormapEvent xcolormap;
    XClientMessageEvent xclient;
    XMappingEvent xmapping;
    XErrorEvent xerror;
    XKeymapEvent xkeymap;
    long int pad[24];
};
typedef union _XEvent {
    int type;
    XAnyEvent xany;
    XKeyEvent xkey;
    XButtonEvent xbutton;

```

```

    XMotionEvent xmotion;
    XCrossingEvent xcrossing;
    XFocusChangeEvent xfocus;
    XExposeEvent xexpose;
    XGraphicsExposeEvent xgraphicsexpose;
    XNoExposeEvent xnoexpose;
    XVisibilityEvent xvisibility;
    XCreateWindowEvent xcreatewindow;
    XDestroyWindowEvent xdestroywindow;
    XUnmapEvent xunmap;
    XMapEvent xmap;
    XMapRequestEvent xmaprequest;
    XReparentEvent xreparent;
    XConfigureEvent xconfigure;
    XGravityEvent xgravity;
    XResizeRequestEvent xresizerequest;
    XConfigureRequestEvent xconfigurerequest;
    XCirculateEvent xcirculate;
    XCirculateRequestEvent xcirculaterequest;
    XPropertyEvent xproperty;
    XSelectionClearEvent xselectionclear;
    XSelectionRequestEvent xselectionrequest;
    XSelectionEvent xselection;
    XColormapEvent xcolormap;
    XClientMessageEvent xclient;
    XMappingEvent xmapping;
    XErrorEvent xerror;
    XKeymapEvent xkeymap;
    long int pad[24];
} XEvent;

typedef struct {
    short lbearing;
    short rbearing;
    short width;
    short ascent;
    short descent;
    unsigned short attributes;
} XCharStruct;

typedef struct {
    Atom name;
    unsigned long int card32;
} XFontProp;

typedef struct {
    XExtData *ext_data;
    Font fid;
    unsigned int direction;
    unsigned int min_char_or_byte2;
    unsigned int max_char_or_byte2;
    unsigned int min_bytel;
    unsigned int max_bytel;
    int all_chars_exist;
    unsigned int default_char;
    int n_properties;
    XFontProp *properties;
    XCharStruct min_bounds;
    XCharStruct max_bounds;
    XCharStruct *per_char;
    int ascent;
    int descent;
} XFontStruct;

typedef struct {
    XRectangle max_ink_extent;
    XRectangle max_logical_extent;
} XFontSetExtents;

```

```

typedef struct {
    char *chars;
    int nchars;
    int delta;
    Font font;
} XTextItem;
typedef struct {
    unsigned char byte1;
    unsigned char byte2;
} XChar2b;
typedef struct {
    XChar2b *chars;
    int nchars;
    int delta;
    Font font;
} XTextItem16;

typedef union {
    Display *display;
    GC gc;
    Visual *visual;
    Screen *screen;
    ScreenFormat *pixmap_format;
    XFontStruct *font;
} XEObject;

typedef struct _XOM *XOM;
typedef struct _XOC *XOC;
typedef struct _XOC *XFontSet;

typedef struct {
    char *chars;
    int nchars;
    int delta;
    XFontSet font_set;
} XmbTextItem;
typedef struct {
    wchar_t *chars;
    int nchars;
    int delta;
    XFontSet font_set;
} XwcTextItem;

typedef struct {
    int charset_count;
    char **charset_list;
} XOMCharSetList;
typedef enum {
    XOMOrientation_LTR_TTB = 0,
    XOMOrientation_RTL_TTB = 1,
    XOMOrientation_TTB_LTR = 2,
    XOMOrientation_TTB_RTL = 3,
    XOMOrientation_Context = 4
} XOrientation;
typedef struct {
    int num_font;
    XFontStruct **font_struct_list;
    char **font_name_list;
} XOMFontInfo;
typedef struct {
    int num_orientation;
    XOrientation *orientation;
} XOMOrientation;

typedef unsigned long int XIMStyle;

```

```

typedef struct {
    unsigned short count_styles;
    XIMStyle *supported_styles;
} XIMStyles;
typedef struct {
    XPointer client_data;
    XIMProc callback;
} XIMCallback;
typedef struct {
    XPointer client_data;
    XICProc callback;
} XICCallback;
typedef void (*XIDProc) (Display *, XPointer, XPointer);
extern int XActivateScreenSaver(Display *);
extern int XAddConnectionWatch(Display *, XConnectionWatchProc,
XPointer);
extern XExtCodes *XAddExtension(Display *);
extern int XAddHost(Display *, XHostAddress *);
extern int XAddHosts(Display *, XHostAddress *, int);
extern int XAddToExtensionList(struct _XExtData **, XExtData *);
extern int XAddToSaveSet(Display *, Window);
extern unsigned long int XAllPlanes(void);
extern int XAllocColor(Display *, Colormap, XColor *);
extern int XAllocColorCells(Display *, Colormap, int, unsigned long
int *,
                                unsigned int, unsigned long int *,
                                unsigned int);
extern int XAllocColorPlanes(Display *, Colormap, int, unsigned
long int *,
                                int, int, int, int, unsigned long int *,
                                unsigned long int *, unsigned long int *);
extern int XAllocNamedColor(Display *, Colormap, const char *,
XColor *,
                                XColor *);
extern int XAllowEvents(Display *, int, Time);
extern int XAutoRepeatOff(Display *);
extern int XAutoRepeatOn(Display *);
extern char *XBaseFontNameListOfFontSet(XFontSet);
extern int XBell(Display *, int);
extern int XBitmapBitOrder(Display *);
extern int XBitmapPad(Display *);
extern int XBitmapUnit(Display *);
extern unsigned long int XBlackPixel(Display *, int);
extern unsigned long int XBlackPixelOfScreen(Screen *);
extern int XCellsOfScreen(Screen *);
extern int XChangeActivePointerGrab(Display *, unsigned int, Cursor,
Time);
extern int XChangeGC(Display *, GC, unsigned long int, XGCValues
*);
extern int XChangeKeyboardControl(Display *, unsigned long int,
XKeyboardControl *);
extern int XChangeKeyboardMapping(Display *, int, int, KeySym *,
int);
extern int XChangePointerControl(Display *, int, int, int, int,
int);
extern int XChangeProperty(Display *, Window, Atom, Atom, int, int,
const unsigned char *, int);
extern int XChangeSaveSet(Display *, Window, int);
extern int XChangeWindowAttributes(Display *, Window, unsigned long
int,
                                XSetWindowAttributes *);
extern int XCheckIfEvent(Display *, XEvent *,
int (*)(Display *, XEvent *, XPointer),
XPointer);
extern int XCheckMaskEvent(Display *, long int, XEvent *);
extern int XCheckTypedEvent(Display *, int, XEvent *);

```



```

extern int XCloseGC(XGC *);
extern int XConfigureWindow(Display *, Window, unsigned int,
                           XWindowChanges *);
extern int XConnectionNumber(Display *);
extern int XContextDependentDrawing(XFontSet);
extern int XContextualDrawing(XFontSet);
extern int XConvertSelection(Display *, Atom, Atom, Atom, Window,
                             Time);
extern int XCopyArea(Display *, Drawable, Drawable, GC, int, int,
                    unsigned int, unsigned int, int, int);
extern Colormap XCopyColormapAndFree(Display *, Colormap);
extern int XCopyGC(Display *, GC, unsigned long int, GC);
extern int XCopyPlane(Display *, Drawable, Drawable, GC, int, int,
                    unsigned int, unsigned int, int, int,
                    unsigned long int);
extern Pixmap XCreateBitmapFromData(Display *, Drawable, const char
*,
                                unsigned int, unsigned int);
extern Colormap XCreateColormap(Display *, Window, Visual *, int);
extern Cursor XCreateFontCursor(Display *, unsigned int);
extern XFontSet XCreateFontSet(Display *, const char *, char ***,
int *,
                                char **);
extern GC XCreateGC(Display *, Drawable, unsigned long int,
XGCValues *);
extern Cursor XCreateGlyphCursor(Display *, Font, Font, unsigned
int,
                                unsigned int, const XColor *,
                                const XColor *);
extern XIC XCreateIC(XIM, ...);
extern XImage *XCreateImage(Display *, Visual *, unsigned int, int,
int,
                                char *, unsigned int, unsigned int, int,
int);
extern XOC XCreateOC(XOM, ...);
extern Pixmap XCreatePixmap(Display *, Drawable, unsigned int,
                            unsigned int, unsigned int);
extern Cursor XCreatePixmapCursor(Display *, Pixmap, Pixmap, XColor
*,
                                XColor *, unsigned int, unsigned int);
extern Pixmap XCreatePixmapFromBitmapData(Display *, Drawable, char
*,
                                unsigned int, unsigned int,
                                unsigned long int,
                                unsigned long int, unsigned

```

```

        Visual *, unsigned long int,
        XSetWindowAttributes *);
extern Colormap XDefaultColormap(Display *, int);
extern Colormap XDefaultColormapOfScreen(Screen *);
extern int XDefaultDepth(Display *, int);
extern int XDefaultDepthOfScreen(Screen *);
extern GC XDefaultGC(Display *, int);
extern GC XDefaultGCOfScreen(Screen *);
extern Window XDefaultRootWindow(Display *);
extern int XDefaultScreen(Display *);
extern Screen *XDefaultScreenOfDisplay(Display *);
extern Visual *XDefaultVisual(Display *, int);
extern Visual *XDefaultVisualOfScreen(Screen *);
extern int XDefineCursor(Display *, Window, Cursor);
extern XModifierKeymap *XDeleteModifiermapEntry(XModifierKeymap *,
        unsigned int, int);
extern int XDeleteProperty(Display *, Window, Atom);
extern void XDestroyIC(XIC);
extern void XDestroyOC(XOC);
extern int XDestroySubwindows(Display *, Window);
extern int XDestroyWindow(Display *, Window);
extern int XDirectionalDependentDrawing(XFontSet);
extern int XDisableAccessControl(Display *);
extern int XDisplayCells(Display *, int);
extern int XDisplayHeight(Display *, int);
extern int XDisplayHeightMM(Display *, int);
extern int XDisplayKeycodes(Display *, int *, int *);
extern unsigned long int XDisplayMotionBufferSize(Display *);
extern char *XDisplayName(const char *);
extern Display *XDisplayOfIM(XIM);
extern Display *XDisplayOfOM(XOM);
extern Display *XDisplayOfScreen(Screen *);
extern int XDisplayPlanes(Display *, int);
extern char *XDisplayString(Display *);
extern int XDisplayWidth(Display *, int);
extern int XDisplayWidthMM(Display *, int);
extern int XDoesBackingStore(Screen *);
extern int XDoesSaveUnders(Screen *);
extern int XDrawArc(Display *, Drawable, GC, int, int, unsigned int,
        unsigned int, int, int);
extern int XDrawArcs(Display *, Drawable, GC, XArc *, int);
extern int XDrawImageString(Display *, Drawable, GC, int, int,
        const char *, int);
extern int XDrawImageStringl6(Display *, Drawable, GC, int, int,
        const XChar2b *, int);
extern int XDrawLine(Display *, Drawable, GC, int, int, int, int);
extern int XDrawLines(Display *, Drawable, GC, XPoint *, int, int);
extern int XDrawPoint(Display *, Drawable, GC, int, int);
extern int XDrawPoints(Display *, Drawable, GC, XPoint *, int, int);
extern int XDrawRectangle(Display *, Drawable, GC, int, int,
        unsigned int,
        unsigned int);
extern int XDrawRectangles(Display *, Drawable, GC, XRectangle *,
        int);
extern int XDrawSegments(Display *, Drawable, GC, XSegment *, int);
extern int XDrawString(Display *, Drawable, GC, int, int, const
        char *,
        int);
extern int XDrawStringl6(Display *, Drawable, GC, int, int,
        const XChar2b *, int);
extern int XDrawText(Display *, Drawable, GC, int, int, XTextItem
        *, int);
extern int XDrawTextl6(Display *, Drawable, GC, int, int,
        XTextIteml6 *,
        int);
extern XExtData **XEHeadOfExtensionList(XEObject);

```

```

extern
    void (*XESetBeforeFlush
          (Display *, int,
           void (*)(Display *, XExtCodes *, const char *,
                    long int))) (Display *, XExtCodes *, const char *,
                                long int);

extern
    int (*XESetCloseDisplay
         (Display *, int, int (*)(Display *, XExtCodes *))) (Display
*,
                                                             XExtCodes *);

extern
    int (*XESetCopyGC(Display *, int, int (*)(Display *, GC,
XExtCodes *)))
        (Display *, GC, XExtCodes *);

extern
    int (*XESetCreateFont
         (Display *, int,
          int (*)(Display *, XFontStruct *, XExtCodes *))) (Display
*,
                                                             XFontStruct *,
                                                             XExtCodes *);

extern
    int (*XESetCreateGC
         (Display *, int, int (*)(Display *, GC, XExtCodes *)))
(Display *,
                                                             GC,
                                                             XExtCodes
                                                             *);

extern
    int (*XESetError
         (Display *, int,
          int (*)(Display *, xError *, XExtCodes *, int *))) (Display
*,
                                                             xError *,
                                                             XExtCodes *,
                                                             int *);

extern char
    (*XESetErrorString
     (Display *, int,
      char (*)(Display *, int, XExtCodes *, char *, int))) (Display
*,
                                                             int,
                                                             XExtCodes *,
                                                             char *, int);

extern
    int (*XESetEventToWire
         (Display *, int,
          int (*)(Display *, XEvent *, xEvent *))) (Display *, XEvent
*,
                                                             xEvent *);

extern
    int (*XESetFlushGC
         (Display *, int, int (*)(Display *, GC, XExtCodes *)))
(Display *,
                                                             GC,
                                                             XExtCodes
                                                             *);

extern
    int (*XESetFreeFont
         (Display *, int,
          int (*)(Display *, XFontStruct *, XExtCodes *))) (Display
*,
                                                             XFontStruct *,
                                                             XExtCodes *);

extern

```

```

    int (*XSetFreeGC(Display *, int, int *) (Display *, GC,
XExtCodes *)))
    (Display *, GC, XExtCodes *);
extern
    void (*XSetPrintErrorValues
        (Display *, int,
        void *) (Display *, XErrorEvent *, void *)) (Display *,
        XErrorEvent *,
        void *);

extern
    int (*XSetWireToError
        (Display *, int,
        int *) (Display *, XErrorEvent *, xError *))) (Display *,
        XErrorEvent *,
        xError *);

extern
    int (*XSetWireToEvent
        (Display *, int,
        int *) (Display *, XEvent *, xEvent *))) (Display *, XEvent
*,
        xEvent *);

extern int XEnableAccessControl(Display *);
extern long int XEventMaskOfScreen(Screen *);
extern int XEventsQueued(Display *, int);
extern long int XExtendedMaxRequestSize(Display *);
extern XFontSetExtents *XExtentsOfFontSet(XFontSet);
extern char *XFetchBuffer(Display *, int *, int);
extern char *XFetchBytes(Display *, int *);
extern int XFetchName(Display *, Window, char **);
extern int XFillArc(Display *, Drawable, GC, int, int, unsigned int,
    unsigned int, int, int);
extern int XFillArcs(Display *, Drawable, GC, XArc *, int);
extern int XFillPolygon(Display *, Drawable, GC, XPoint *, int, int,
    int);
extern int XFillRectangle(Display *, Drawable, GC, int, int,
    unsigned int,
    unsigned int);
extern int XFillRectangles(Display *, Drawable, GC, XRectangle *,
    int);
extern int XFilterEvent(XEvent *, Window);
extern XExtData *XFindOnExtensionList(XExtData * *, int);
extern int XFlush(Display *);
extern void XFlushGC(Display *, GC);
extern int XFontsOfFontSet(XFontSet, XFontStruct * **, char ***);
extern int XForceScreenSaver(Display *, int);
extern int XFree(void *);
extern int XFreeColormap(Display *, Colormap);
extern int XFreeColors(Display *, Colormap, unsigned long int *,
    int,
    unsigned long int);
extern int XFreeCursor(Display *, Cursor);
extern int XFreeExtensionList(char **);
extern int XFreeFont(Display *, XFontStruct *);
extern int XFreeFontInfo(char **, XFontStruct *, int);
extern int XFreeFontNames(char **);
extern int XFreeFontPath(char **);
extern void XFreeFontSet(Display *, XFontSet);
extern int XFreeGC(Display *, GC);
extern int XFreeModifiermap(XModifierKeymap *);
extern int XFreePixmap(Display *, Pixmap);
extern void XFreeStringList(char **);
extern GContext XGContextFromGC(GC);
extern int XGeometry(Display *, int, const char *, const char *,
    unsigned int, unsigned int, unsigned int, int,
    int,
    int *, int *, int *, int *);

```

```

extern char *XGetAtomName(Display *, Atom);
extern int XGetAtomNames(Display *, Atom *, int, char **);
extern int XGetCommand(Display *, Window, char ***, int *);
extern char *XGetDefault(Display *, const char *, const char *);
extern int XGetErrorDatabaseText(Display *, const char *, const
char *,
                                const char *, char *, int);
extern int XGetErrorText(Display *, int, char *, int);
extern char **XGetFontPath(Display *, int *);
extern int XGetFontProperty(XFontStruct *, Atom, unsigned long int
*);
extern int XGetGCValues(Display *, GC, unsigned long int, XGCValues
*);
extern int XGetGeometry(Display *, Drawable, Window *, int *, int
*,
                                unsigned int *, unsigned int *, unsigned int *,
                                unsigned int *);
extern char *XGetICValues(XIC, ...);
extern char *XGetIMValues(XIM, ...);
extern int XGetIconName(Display *, Window, char **);
extern XImage *XGetImage(Display *, Drawable, int, int, unsigned
int,
                                unsigned int, unsigned long int, int);
extern int XGetInputFocus(Display *, Window *, int *);
extern int XGetKeyboardControl(Display *, XKeyboardState *);
extern KeySym *XGetKeyboardMapping(Display *, unsigned int, int,
int *);
extern XModifierKeymap *XGetModifierMapping(Display *);
extern XTimeCoord *XGetMotionEvents(Display *, Window, Time, Time,
int *);
extern char *XGetOCValues(XOC, ...);
extern char *XGetOMValues(XOM, ...);
extern int XGetPointerControl(Display *, int *, int *, int *);
extern int XGetPointerMapping(Display *, unsigned char *, int);
extern int XGetScreenSaver(Display *, int *, int *, int *, int *);
extern Window XGetSelectionOwner(Display *, Atom);
extern XImage *XGetSubImage(Display *, Drawable, int, int, unsigned
int,
                                unsigned int, unsigned long int, int,
XImage *,
                                int, int);
extern int XGetTransientForHint(Display *, Window, Window *);
extern int XGetWMColormapWindows(Display *, Window, Window * *, int
*);
extern int XGetWMProtocols(Display *, Window, Atom * *, int *);
extern int XGetWindowAttributes(Display *, Window,
XWindowAttributes *);
extern int XGetWindowProperty(Display *, Window, Atom, long int,
long int,
                                int, Atom, Atom *, int *,
                                unsigned long int *, unsigned long int *,
                                unsigned char **);
extern int XGrabButton(Display *, unsigned int, unsigned int,
Window, int,
                                unsigned int, int, int, Window, Cursor);
extern int XGrabKey(Display *, int, unsigned int, Window, int, int,
int);
extern int XGrabKeyboard(Display *, Window, int, int, int, Time);
extern int XGrabPointer(Display *, Window, int, unsigned int, int,
int,
                                Window, Cursor, Time);
extern int XGrabServer(Display *);
extern int XHeightMMOfScreen(Screen *);
extern int XHeightOfScreen(Screen *);
extern XIM XIMOfIC(XIC);
extern int XIconifyWindow(Display *, Window, int);

```

```

extern int XIfEvent(Display *, XEvent *,
                   int (*)(Display *, XEvent *, XPointer), XPointer);
extern int XImageByteOrder(Display *);
extern XExtCodes *XInitExtension(Display *, const char *);
extern int XInitImage(XImage *);
extern int XInitThreads(void);
extern XModifierKeymap *XInsertModifiermapEntry(XModifierKeymap *,
                                                unsigned int, int);
extern int XInstallColormap(Display *, Colormap);
extern Atom XInternAtom(Display *, const char *, int);
extern int XInternAtoms(Display *, char **, int, int, Atom *);
extern int XInternalConnectionNumbers(Display *, int **, int *);
extern KeySym XKeycodeToKeysym(Display *, unsigned int, int);
extern KeyCode XKeysymToKeycode(Display *, KeySym);
extern char *XKeysymToString(KeySym);
extern int XKillClient(Display *, XID);
extern unsigned long int XLastKnownRequestProcessed(Display *);
extern int *XListDepths(Display *, int, int *);
extern char **XListExtensions(Display *, int *);
extern char **XListFonts(Display *, const char *, int, int *);
extern char **XListFontsWithInfo(Display *, const char *, int, int
*,
                                XFontStruct * *);
extern XHostAddress *XListHosts(Display *, int *, int *);
extern Colormap *XListInstalledColormaps(Display *, Window, int *);
extern XPixmapFormatValues *XListPixmapFormats(Display *, int *);
extern Atom *XListProperties(Display *, Window, int *);
extern Font XLoadFont(Display *, const char *);
extern XFontStruct *XLoadQueryFont(Display *, const char *);
extern char *XLocaleOfFontSet(XFontSet);
extern char *XLocaleOfIM(XIM);
extern char *XLocaleOfOM(XOM);
extern void XLockDisplay(Display *);
extern int XLookupColor(Display *, Colormap, const char *, XColor
*,
                        XColor *);
extern KeySym XLookupKeysym(XKeyEvent *, int);
extern int XLowerWindow(Display *, Window);
extern int XMapRaised(Display *, Window);
extern int XMapSubwindows(Display *, Window);
extern int XMapWindow(Display *, Window);
extern int XMaskEvent(Display *, long int, XEvent *);
extern int XMaxCmapsOfScreen(Screen *);
extern long int XMaxRequestSize(Display *);
extern int XMinCmapsOfScreen(Screen *);
extern int XMoveResizeWindow(Display *, Window, int, int, unsigned
int,
                        unsigned int);
extern int XMoveWindow(Display *, Window, int, int);
extern XModifierKeymap *XNewModifiermap(int);
extern int XNextEvent(Display *, XEvent *);
extern unsigned long int XNextRequest(Display *);
extern int XNoOp(Display *);
extern XOM XOMOfOC(XOC);
extern Display *XOpenDisplay(const char *);
extern XIM XOpenIM(Display *, struct _XrmHashBucketRec *, char *,
char *);
extern XOM XOpenOM(Display *, struct _XrmHashBucketRec *, const
char *,
                    const char *);
extern int XParseColor(Display *, Colormap, const char *, XColor
*);
extern int XParseGeometry(const char *, int *, int *, unsigned int
*,
                        unsigned int *);
extern int XPeekEvent(Display *, XEvent *);

```

```

extern int XPeekIfEvent(Display *, XEvent *,
                        int (*)(Display *, XEvent *, XPointer),
                        XPointer);
extern int XPending(Display *);
extern int XPlanesOfScreen(Screen *);
extern void XProcessInternalConnection(Display *, int);
extern int XProtocolRevision(Display *);
extern int XProtocolVersion(Display *);
extern int XPutBackEvent(Display *, XEvent *);
extern int XPutImage(Display *, Drawable, GC, XImage *, int, int,
int, int,
                        unsigned int, unsigned int);
extern int XQLength(Display *);
extern int XQueryBestCursor(Display *, Drawable, unsigned int,
                        unsigned int, unsigned int *, unsigned int
*);
extern int XQueryBestSize(Display *, int, Drawable, unsigned int,
                        unsigned int, unsigned int *, unsigned int *);
extern int XQueryBestStipple(Display *, Drawable, unsigned int,
                        unsigned int, unsigned int *, unsigned int
*);
extern int XQueryBestTile(Display *, Drawable, unsigned int,
                        unsigned int,
                        unsigned int *, unsigned int *);
extern int XQueryColor(Display *, Colormap, XColor *);
extern int XQueryColors(Display *, Colormap, XColor *, int);
extern int XQueryExtension(Display *, const char *, int *, int *,
int *);
extern XFontStruct *XQueryFont(Display *, XID);
extern int XQueryKeymap(Display *, char[32]);
extern int XQueryPointer(Display *, Window, Window *, Window *, int
*,
                        int *, int *, int *, unsigned int *);
extern int XQueryTextExtents(Display *, XID, const char *, int, int
*,
                        int *, int *, XCharStruct *);
extern int XQueryTextExtents16(Display *, XID, const XChar2b *, int,
int *,
                        int *, int *, XCharStruct *);
extern int XQueryTree(Display *, Window, Window *, Window *, Window
* *,
                        unsigned int *);
extern int XRaiseWindow(Display *, Window);
extern int XReadBitmapFile(Display *, Drawable, const char *,
                        unsigned int *, unsigned int *, Pixmap *,
int *,
                        int *);
extern int XReadBitmapFileData(const char *, unsigned int *,
                        unsigned int *, unsigned char **, int *,
int *);
extern int XRebindKeysym(Display *, KeySym, KeySym *, int,
                        const unsigned char *, int);
extern int XRecolorCursor(Display *, Cursor, XColor *, XColor *);
extern int XReconfigureWMWindow(Display *, Window, int, unsigned
int,
                        XWindowChanges *);
extern int XRefreshKeyboardMapping(XMappingEvent *);
extern int XRegisterIMInstantiateCallback(Display *,
                        struct _XrmHashBucketRec *,
                        char *, char *, XIDProc,
                        XPointer);
extern void XRemoveConnectionWatch(Display *, XConnectionWatchProc,
                        XPointer);
extern int XRemoveFromSaveSet(Display *, Window);
extern int XRemoveHost(Display *, XHostAddress *);
extern int XRemoveHosts(Display *, XHostAddress *, int);

```



```

extern int XReparentWindow(Display *, Window, Window, int, int);
extern int XResetScreenSaver(Display *);
extern int XResizeWindow(Display *, Window, unsigned int, unsigned
int);
extern char *XResourceManagerString(Display *);
extern int XRestackWindows(Display *, Window *, int);
extern Window XRootWindow(Display *, int);
extern Window XRootWindowOfScreen(Screen *);
extern int XRotateBuffers(Display *, int);
extern int XRotateWindowProperties(Display *, Window, Atom *, int,
int);
extern int XScreenCount(Display *);
extern int XScreenNumberOfScreen(Screen *);
extern Screen *XScreenOfDisplay(Display *, int);
extern char *XScreenResourceString(Screen *);
extern int XSelectInput(Display *, Window, long int);
extern int XSendEvent(Display *, Window, int, long int, XEvent *);
extern char *XServerVendor(Display *);
extern int XSetAccessControl(Display *, int);
extern int (*XSetAfterFunction(Display *, int (*)(Display *)))
(Display *);
extern int XSetArcMode(Display *, GC, int);
extern void XSetAuthorization(char *, int, char *, int);
extern int XSetBackground(Display *, GC, unsigned long int);
extern int XSetClipMask(Display *, GC, Pixmap);
extern int XSetClipOrigin(Display *, GC, int, int);
extern int XSetClipRectangles(Display *, GC, int, int, XRectangle
*, int,
int);
extern int XSetCloseDownMode(Display *, int);
extern int XSetCommand(Display *, Window, char **, int);
extern int XSetDashes(Display *, GC, int, const char *, int);
extern XErrorHandler XSetErrorHandler(XErrorHandler);
extern int XSetFillRule(Display *, GC, int);
extern int XSetFillStyle(Display *, GC, int);
extern int XSetFont(Display *, GC, Font);
extern int XSetFontPath(Display *, char **, int);
extern int XSetForeground(Display *, GC, unsigned long int);
extern int XSetFunction(Display *, GC, int);
extern int XSetGraphicsExposures(Display *, GC, int);
extern void XSetICFocus(XIC);
extern char *XSetICValues(XIC, ...);
extern char *XSetIMValues(XIM, ...);
extern XIOErrorHandler XSetIOErrorHandler(XIOErrorHandler);
extern int XSetIconName(Display *, Window, const char *);
extern int XSetInputFocus(Display *, Window, int, Time);
extern int XSetLineAttributes(Display *, GC, unsigned int, int, int,
int);
extern char *XSetLocaleModifiers(const char *);
extern int XSetModifierMapping(Display *, XModifierKeymap *);
extern char *XSetOCValues(XOC, ...);
extern char *XSetOMValues(XOM, ...);
extern int XSetPlaneMask(Display *, GC, unsigned long int);
extern int XSetPointerMapping(Display *, const unsigned char *,
int);
extern int XSetScreenSaver(Display *, int, int, int, int);
extern int XSetSelectionOwner(Display *, Atom, Window, Time);
extern int XSetState(Display *, GC, unsigned long int, unsigned
long int,
int, unsigned long int);
extern int XSetStipple(Display *, GC, Pixmap);
extern int XSetSubwindowMode(Display *, GC, int);
extern int XSetTSOrigin(Display *, GC, int, int);
extern int XSetTile(Display *, GC, Pixmap);
extern int XSetTransientForHint(Display *, Window, Window);
extern int XSetWMColormapWindows(Display *, Window, Window *, int);

```



```

extern int XSetWMProtocols(Display *, Window, Atom *, int);
extern int XSetWindowBackground(Display *, Window, unsigned long
int);
extern int XSetWindowBackgroundPixmap(Display *, Window, Pixmap);
extern int XSetWindowBorder(Display *, Window, unsigned long int);
extern int XSetWindowBorderPixmap(Display *, Window, Pixmap);
extern int XSetWindowBorderWidth(Display *, Window, unsigned int);
extern int XSetWindowColormap(Display *, Window, Colormap);
extern int XStoreBuffer(Display *, const char *, int, int);
extern int XStoreBytes(Display *, const char *, int);
extern int XStoreColor(Display *, Colormap, XColor *);
extern int XStoreColors(Display *, Colormap, XColor *, int);
extern int XStoreName(Display *, Window, const char *);
extern int XStoreNamedColor(Display *, Colormap, const char *,
    unsigned long int, int);
extern KeySym XStringToKeysym(const char *);
extern int XSupportsLocale(void);
extern int XSync(Display *, int);
extern int (*XSynchronize(Display *, int)) (Display *);
extern int XTextExtents(XFontStruct *, const char *, int, int *,
int *,
    int *, XCharStruct *);
extern int XTextExtents16(XFontStruct *, const XChar2b *, int, int
*,
    int *, int *, XCharStruct *);
extern int XTextWidth(XFontStruct *, const char *, int);
extern int XTextWidth16(XFontStruct *, const XChar2b *, int);
extern int XTranslateCoordinates(Display *, Window, Window, int,
int,
    int *, int *, Window *);
extern int XUndefineCursor(Display *, Window);
extern int XUngrabButton(Display *, unsigned int, unsigned int,
Window);
extern int XUngrabKey(Display *, int, unsigned int, Window);
extern int XUngrabKeyboard(Display *, Time);
extern int XUngrabPointer(Display *, Time);
extern int XUngrabServer(Display *);
extern int XUninstallColormap(Display *, Colormap);
extern int XUnloadFont(Display *, Font);
extern void XUnlockDisplay(Display *);
extern int XUnmapSubwindows(Display *, Window);
extern int XUnmapWindow(Display *, Window);
extern int XUnregisterIMInstantiateCallback(Display *,
    struct _XrmHashBucketRec *,
    char *, char *, XIDProc,
    XPointer);

extern void XUnsetICFocus(XIC);
extern XVaNestedList XVaCreateNestedList(int, ...);
extern int XVendorRelease(Display *);
extern VisualID XVisualIDFromVisual(Visual *);
extern int XWarpPointer(Display *, Window, Window, int, int,
unsigned int,
    unsigned int, int, int);
extern unsigned long int XWhitePixel(Display *, int);
extern unsigned long int XWhitePixelOfScreen(Screen *);
extern int XWidthMMOfScreen(Screen *);
extern int XWidthOfScreen(Screen *);
extern int XWindowEvent(Display *, Window, long int, XEvent *);
extern int XWithdrawWindow(Display *, Window, int);
extern int XWriteBitmapFile(Display *, const char *, Pixmap,
unsigned int,
    unsigned int, int, int);
extern void XmbDrawImageString(Display *, Drawable, XFontSet, GC,
int, int,
    const char *, int);

```

```

extern void XmbDrawString(Display *, Drawable, XFontSet, GC, int,
int,
const char *, int);
extern void XmbDrawText(Display *, Drawable, GC, int, int,
XmbTextItem *,
int);
extern int XmbLookupString(XIC, XKeyPressedEvent *, char *, int,
KeySym *,
int *);
extern char *XmbResetIC(XIC);
extern int XmbTextEscapement(XFontSet, const char *, int);
extern int XmbTextExtents(XFontSet, const char *, int, XRectangle
*,
XRectangle *);
extern int XmbTextPerCharExtents(XFontSet, const char *, int,
XRectangle *,
XRectangle *, int, int *, XRectangle *,
XRectangle *);
extern void XrmInitialize(void);
extern void XwcDrawImageString(Display *, Drawable, XFontSet, GC,
int, int,
const wchar_t *, int);
extern void XwcDrawString(Display *, Drawable, XFontSet, GC, int,
int,
const wchar_t *, int);
extern void XwcDrawText(Display *, Drawable, GC, int, int,
XwcTextItem *,
int);
extern int XwcLookupString(XIC, XKeyPressedEvent *, wchar_t *, int,
KeySym *, int *);
extern wchar_t *XwcResetIC(XIC);
extern int XwcTextEscapement(XFontSet, const wchar_t *, int);
extern int XwcTextExtents(XFontSet, const wchar_t *, int,
XRectangle *,
XRectangle *);
extern int XwcTextPerCharExtents(XFontSet, const wchar_t *, int,
XRectangle *, XRectangle *, int, int *,
XRectangle *, XRectangle *);

```

6.2.10 X11/Xlibint.h

```

#define NEED_EVENTS
#define NEED_REPLIES
#define WORD64ALIGN
#define min(a,b) ((a) < (b)) ? (a) : (b)
#define max(a,b) ((a) > (b)) ? (a) : (b)
#define CI_NONEXISTCHAR(cs) (((cs)->width == 0) && (((cs)->rbearing| (cs)->lbearing| (cs)->ascent| (cs)->descent) == 0))
#define XAllocIDs(dpy,ids,n) (* (dpy)->idlist_alloc) (dpy,ids,n)
#define XlibDisplayIOError (1L << 0)
#define XlibDisplayClosing (1L << 1)
#define XlibDisplayNoXkb (1L << 2)
#define XlibDisplayPrivSync (1L << 3)
#define XlibDisplayProcConni (1L << 4)
#define XlibDisplayReadEvents (1L << 5)
#define XlibDisplayReply (1L << 5)
#define XlibDisplayWriting (1L << 6)
#define XlibDisplayDfltRMDb (1L << 7)
#define UNLOCKED 0
#define LOCKED 1
#define WRCTSPERBATCH 10
#define PTSPERBATCH 1024
#define ZLNSPERBATCH 1024
#define BUFSIZE 2048
#define FARCSPERBATCH 256

```

```

#define FRCTSPERBATCH 256
#define XCONN_CHECK_FREQ 256
#define ZRCTSPERBATCH 256
#define WLNSPERBATCH 50
#define Xcalloc(nelem,elsize) calloc((nelem), (elsize))
#define CI_GET_DEFAULT_INFO_1D(fs,cs) CI_GET_CHAR_INFO_1D (fs,
fs->default_char, NULL, cs)
#define CURSORFONT "cursor"
#define Data16(dpy,data,len) Data((dpy), (char *) (data), (len))
#define PackData16(dpy,data,len) Data16 (dpy, data, len)
#define PackData32(dpy,data,len) Data32 (dpy, data, len)
#define _XFopenFile(path,mode) fopen(path,mode)
#define Xfree(ptr) free((ptr))
#define LockDisplay(d) if ((d)->lock_fns) (*(d)->lock_fns-
>lock_display)(d)
#define UnlockDisplay(d) if ((d)->lock_fns) (*(d)->lock_fns-
>unlock_display)(d)
#define FlushGC(dpy,gc) if ((gc)->dirty) _XFlushGCCache((dpy), (gc))
#define SetReqLen(req,n,badlen) if ((req->length + n) >
(unsigned)65535) { if (dpy->bigreq_size) { MakeBigReq(req,n) } else
{ n = badlen; req->length += n; } } else req->length += n
#define BufAlloc(type,ptr,n) if (dpy->bufptr + (n) > dpy->bufmax)
_XFlush (dpy); ptr = (type) dpy->bufptr; (void)ptr; dpy->bufptr +=
(n);
#define SyncHandle() if (dpy->synchandler) (*(dpy-
>synchandler)(dpy)
#define _XCreateMutex(lock) if (_XCreateMutex_fn)
(* _XCreateMutex_fn)(lock);
#define _XFreeMutex(lock) if (_XFreeMutex_fn)
(* _XFreeMutex_fn)(lock);
#define _XLockMutex(lock) if (_XLockMutex_fn)
(* _XLockMutex_fn)(lock)
#define _XUnlockMutex(lock) if (_XUnlockMutex_fn)
(* _XUnlockMutex_fn)(lock)
#define Xmalloc(size) malloc((size))
#define _XOpenFile(path,flags) open(path,flags)
#define _XOpenFileMode(path,flags,mode) open(path,flags,mode)
#define PackData(dpy,data,len) PackData16 (dpy, data, len)
#define Xrealloc(ptr,size) realloc((ptr), (size))
#define GetReqExtra(name,n,req) WORD64ALIGN if ((dpy->bufptr +
sizeof(x ##name ##Req) + n) > dpy->bufmax) _XFlush(dpy); req = (x
##name ##Req *) (dpy->last_req = dpy->bufptr); req->reqType = X_
##name; req->length = (sizeof(x ##name ##Req) + n)>>2; dpy->bufptr
+= sizeof(x ##name ##Req) + n; dpy->request++
#define GetReq(name,req) WORD64ALIGN if ((dpy->bufptr +
sizeof(x ##name ##Req)) > dpy->bufmax) _XFlush(dpy); req = (x ##name
##Req *) (dpy->last_req = dpy->bufptr); req->reqType = X_ ##name;
req->length = (sizeof(x ##name ##Req))>>2; dpy->bufptr += sizeof(x
##name ##Req); dpy->request++
#define GetEmptyReq(name,req) WORD64ALIGN if ((dpy->bufptr +
sizeof(xReq)) > dpy->bufmax) _XFlush(dpy); req = (xReq *) (dpy-
>last_req = dpy->bufptr); req->reqType = X_ ##name; req->length =
1; dpy->bufptr += sizeof(xReq); dpy->request++
#define GetResReq(name,rid,req) WORD64ALIGN if ((dpy->bufptr +
sizeof(xResourceReq)) > dpy->bufmax) _XFlush(dpy); req =
(xResourceReq *) (dpy->last_req = dpy->bufptr); req->reqType = X_
##name; req->length = 2; req->id = (rid); dpy->bufptr +=
sizeof(xResourceReq); dpy->request++
#define _XBCOPYFUNC _Xbcopy
#define Data32(dpy,data,len) _XData32(dpy, (long *)data, len)
#define _XRead16(dpy,data,len) _XRead((dpy), (char *) (data), (len))
#define _XRead16Pad(dpy,data,len) _XReadPad((dpy), (char
*) (data), (len))
#define OneDataCard32(dpy,dstaddr,srcvar) { *(CARD32
*) (dstaddr) = (srcvar); }

```

```

#define CI_GET_CHAR_INFO_1D(fs,col,def,cs)      { cs = def; if (col
>= fs->min_char_or_byte2 && col <= fs->max_char_or_byte2) { if (fs-
>per_char == NULL) { cs = &fs->min_bounds; } else { cs = &fs-
>per_char[(col - fs->min_char_or_byte2)]; if (CI_NONEXISTCHAR(cs))
cs = def; } } }
#define CI_GET_CHAR_INFO_2D(fs,row,col,def,cs) { cs = def; if (row
>= fs->min_bytel && row <= fs->max_bytel && col >= fs-
>min_char_or_byte2 && col <= fs->max_char_or_byte2) { if (fs-
>per_char == NULL) { cs = &fs->min_bounds; } else { cs = &fs-
>per_char[((row - fs->min_bytel) * (fs->max_char_or_byte2 - fs-
>min_char_or_byte2 + 1)) + (col - fs->min_char_or_byte2)]; if
(CI_NONEXISTCHAR(cs)) cs = def; } } }
#define DeqAsyncHandler(dpy,handler)          { if (dpy->async_handlers
== (handler)) dpy->async_handlers = (handler)->next; else
_XDeqAsyncHandler(dpy, handler); }
#define Data(dpy,data,len)                    { if (dpy->bufptr + (len) <= dpy-
>bufmax) { memcpy(dpy->bufptr, data, (int)len); dpy->bufptr +=
((len) + 3) & ~3; } else _XSend(dpy, data, len); }
#define CI_GET_DEFAULT_INFO_2D(fs,cs)         { unsigned int r = (fs-
>default_char >> 8); unsigned int c = (fs->default_char & 0xff);
CI_GET_CHAR_INFO_2D (fs, r, c, NULL, cs); }

typedef struct _XSQEvent {
    struct _XSQEvent *next;
    XEvent event;
    unsigned long int qserial_num;
} _XQEvent;
typedef struct _LockInfoRec *LockInfoPtr;
typedef struct _XInternalAsync {
    struct _XInternalAsync *next;
    int (*handler) (Display *, xReply *, char *, int, XPointer);
    XPointer data;
} _XAsyncHandler;
typedef struct _XAsyncEState {
    unsigned long int min_sequence_number;
    unsigned long int max_sequence_number;
    unsigned char error_code;
    unsigned char major_opcode;
    unsigned short minor_opcode;
    unsigned char last_error_received;
    int error_count;
} _XAsyncErrorState;
typedef void (*FreeFuncType) (Display *);
typedef int (*FreeModmapType) (XModifierKeymap *);
typedef struct _XFreeFuncs {
    FreeFuncType atoms;
    FreeModmapType modifiermap;
    FreeFuncType key_bindings;
    FreeFuncType context_db;
    FreeFuncType defaultCCCs;
    FreeFuncType clientCmaps;
    FreeFuncType intensityMaps;
    FreeFuncType im_filters;
    FreeFuncType xkb;
} _XFreeFuncRec;
typedef int (*CreateGCType) (Display *, GC, XExtCodes *);
typedef int (*CopyGCType) (Display *, GC, XExtCodes *);
typedef int (*FlushGCType) (Display *, GC, XExtCodes *);
typedef int (*FreeGCType) (Display *, GC, XExtCodes *);
typedef int (*CreateFontType) (Display *, XFontStruct *, XExtCodes
*);
typedef int (*FreeFontType) (Display *, XFontStruct *, XExtCodes
*);
typedef int (*CloseDisplayType) (Display *, XExtCodes *);
typedef int (*ErrorType) (Display *, xError *, XExtCodes *, int *);

```

```

typedef char *(*ErrorStringType) (Display *, int, XExtCodes *, char
*,
                                int);
typedef void (*PrintErrorType) (Display *, XErrorEvent *, void *);
typedef void (*BeforeFlushType) (Display *, XExtCodes *, const char
*,
                                long int);

typedef struct _XExten {
    struct _XExten *next;
    XExtCodes codes;
    CreateGCType create_GC;
    CopyGCType copy_GC;
    FlushGCType flush_GC;
    FreeGCType free_GC;
    CreateFontType create_Font;
    FreeFontType free_Font;
    CloseDisplayType close_display;
    ErrorType error;
    ErrorStringType error_string;
    char *name;
    PrintErrorType error_values;
    BeforeFlushType before_flush;
    struct _XExten *next_flush;
} _XExtension;

typedef void (*_XInternalConnectionProc) (Display *, int, XPointer);

struct _XDisplay {
    XExtData *ext_data;
    struct _XFreeFuncs *free_funcs;
    int fd;
    int conn_checker;
    int proto_major_version;
    int proto_minor_version;
    char *vendor;
    XID resource_base;
    XID resource_mask;
    XID resource_id;
    int resource_shift;
    XID(*resource_alloc) (struct _XDisplay *);
    int byte_order;
    int bitmap_unit;
    int bitmap_pad;
    int bitmap_bit_order;
    int nformats;
    ScreenFormat *pixmap_format;
    int vnumber;
    int release;
    struct _XSeqEvent *head;
    struct _XSeqEvent *tail;
    int qlen;
    unsigned long int last_request_read;
    unsigned long int request;
    char *last_req;
    char *buffer;
    char *bufptr;
    char *bufmax;
    unsigned int max_request_size;
    struct _XrmHashBucketRec *db;
    int (*synchandler) (struct _XDisplay *);
    char *display_name;
    int default_screen;
    int nscreens;
    Screen *screens;
    unsigned long int motion_buffer;
    unsigned long int flags;
    int min_keycode;
    int max_keycode;

```

```

KeySym *keysyms;
XModifierKeymap *modifiermap;
int keysyms_per_keycode;
char *xdefaults;
char *scratch_buffer;
unsigned long int scratch_length;
int ext_number;
struct _XExten *ext_procs;
int (*event_vec[128]) (Display *, XEvent *, xEvent *);
int (*wire_vec[128]) (Display *, XEvent *, xEvent *);
KeySym lock_meaning;
struct _XLockInfo *lock;
struct _XInternalAsync *async_handlers;
unsigned long int bigreq_size;
struct _XLockPtrs *lock_fns;
void (*idlist_alloc) (Display *, XID *, int);
struct _XKeytrans *key_bindings;
Font cursor_font;
struct _XDisplayAtoms *atoms;
unsigned int mode_switch;
unsigned int num_lock;
struct _XContextDB *context_db;
int (**error_vec) (Display *, XErrorEvent *, xError *);
struct {
    XPointer defaultCCCs;
    XPointer clientCmaps;
    XPointer perVisualIntensityMaps;
} cms;
struct _XIMFilter *im_filters;
struct _XSQEvent *qfree;
unsigned long int next_event_serial_num;
struct _XExten *flushes;
struct _XConnectionInfo *im_fd_info;
int im_fd_length;
struct _XConnWatchInfo *conn_watchers;
int watcher_count;
XPointer filedesc;
int (*savesynchandler) (Display *);
XID resource_max;
int xcmisc_opcode;
struct _XkbInfoRec *xkb_info;
struct _XtransConnInfo *trans_conn;
};
struct _XGC {
    XExtData *ext_data;
    GCContext gid;
    int rects;
    int dashes;
    unsigned long int dirty;
    XGCValues values;
};

```

6.2.11 X11/Xmd.h

```

#define B16
#define B32
#define NEXTPTR(p,t) (((t *) (p)) + 1)
#define cvtINT16toInt(val) (val)
#define cvtINT16toLong(val) (val)
#define cvtINT16toShort(val) (val)
#define cvtINT32toInt(val) (val)
#define cvtINT32toLong(val) (val)
#define cvtINT32toShort(val) (val)
#define cvtINT8toInt(val) (val)
#define cvtINT8toLong(val) (val)

```

```

#define cvtINT8toShort(val)      (val)
#define _SIZEOF(x)              sz_ ##x
#define SIZEOF(x)               _SIZEOF(x)

typedef unsigned short CARD16;
typedef unsigned char CARD8;
typedef CARD8 BOOL;
typedef unsigned char BYTE;
typedef CARD32 BITS32;
typedef CARD16 BITS16;
typedef int INT32;
typedef short int INT16;
typedef signed char INT8;

```

6.2.12 X11/Xos.h

```

#define _XOS_H
#define X_GETTIMEOFDAY(t)      gettimeofday(t, (struct timezone*)0)

```

6.2.13 X11/Xosdefs.h

```

#ifndef MAXPATHLEN
#define MAXPATHLEN 4096
#endif
#ifndef PATH_MAX
#define PATH_MAX 4096
#endif

```

6.2.14 X11/Xproto.h

```

#define ELFlagFocus      (1<<0)
#define ELFlagSameScreen (1<<1)
#define XEventSize      (sizeof(xEvent))
#define X_Error 0
#define xFalse 0
#define X_CreateWindow 1
#define X_Reply 1
#define xTrue 1
#define X_UnmapWindow 10
#define X_ChangeKeyboardMapping 100
#define X_GetKeyboardMapping 101
#define X_ChangeKeyboardControl 102
#define X_GetKeyboardControl 103
#define X_Bell 104
#define X_ChangePointerControl 105
#define X_GetPointerControl 106
#define X_SetScreenSaver 107
#define X_GetScreenSaver 108
#define X_ChangeHosts 109
#define X_UnmapSubwindows 11
#define X_ListHosts 110
#define X_SetAccessControl 111
#define X_SetCloseDownMode 112
#define X_KillClient 113
#define X_RotateProperties 114
#define X_ForceScreenSaver 115
#define X_SetPointerMapping 116
#define X_GetPointerMapping 117
#define X_SetModifierMapping 118
#define X_GetModifierMapping 119
#define X_ConfigureWindow 12
#define sz_xAllocColorCellsReq 12

```

```

#define sz_xAllocNamedColorReq 12
#define sz_xArc 12
#define sz_xChangeGCReq 12
#define sz_xChangePointerControlReq 12
#define sz_xChangeWindowAttributesReq 12
#define sz_xCharInfo 12
#define sz_xColorItem 12
#define sz_xConfigureWindowReq 12
#define sz_xConnClientPrefix 12
#define sz_xCopyColormapAndFreeReq 12
#define sz_xDeletePropertyReq 12
#define sz_xFreeColorsReq 12
#define sz_xLookupColorReq 12
#define sz_xOpenFontReq 12
#define sz_xPolyArcReq 12
#define sz_xPolyFillArcReq 12
#define sz_xPolyFillRectangleReq 12
#define sz_xPolyLineReq 12
#define sz_xPolyPointReq 12
#define sz_xPolyRectangleReq 12
#define sz_xPolySegmentReq 12
#define sz_xQueryBestSizeReq 12
#define sz_xRotatePropertiesReq 12
#define sz_xSetClipRectanglesReq 12
#define sz_xSetDashesReq 12
#define sz_xSetInputFocusReq 12
#define sz_xSetScreenSaverReq 12
#define sz_xUngrabButtonReq 12
#define sz_xUngrabKeyReq 12
#define X_NoOperation 127
#define X_CirculateWindow 13
#define X_GetGeometry 14
#define X_QueryTree 15
#define X_InternAtom 16
#define sz_xAllocColorPlanesReq 16
#define sz_xAllocColorReq 16
#define sz_xChangeActivePointerGrabReq 16
#define sz_xClearAreaReq 16
#define sz_xCopyGCReq 16
#define sz_xCreateColormapReq 16
#define sz_xCreateGCReq 16
#define sz_xCreatePixmapReq 16
#define sz_xFillPolyReq 16
#define sz_xGetMotionEventsReq 16
#define sz_xGrabKeyReq 16
#define sz_xGrabKeyboardReq 16
#define sz_xImageText16Req 16
#define sz_xImageText8Req 16
#define sz_xImageTextReq 16
#define sz_xPolyText16Req 16
#define sz_xPolyText8Req 16
#define sz_xPolyTextReq 16
#define sz_xReparentWindowReq 16
#define sz_xSetSelectionOwnerReq 16
#define sz_xStoreNamedColorReq 16
#define sz_xTranslateCoordsReq 16
#define X_GetAtomName 17
#define X_ChangeProperty 18
#define X_DeleteProperty 19
#define X_ChangeWindowAttributes 2
#define sz_xTextElt 20
#define X_GetProperty 20
#define sz_xGetImageReq 20
#define sz_xRecolorCursorReq 20
#define X_ListProperties 21
#define X_SetSelectionOwner 22

```



```

#define X_GetSelectionOwner      23
#define X_ConvertSelection      24
#define sz_xChangePropertyReq   24
#define sz_xConvertSelectionReq 24
#define sz_xGetPropertyReq      24
#define sz_xGrabButtonReq      24
#define sz_xGrabPointerReq      24
#define sz_xPropIconSize       24
#define sz_xPutImageReq        24
#define sz_xVisualType         24
#define sz_xWarpPointerReq      24
#define X_SendEvent            25
#define X_GrabPointer          26
#define X_UngrabPointer        27
#define X_GrabButton           28
#define sz_xCopyAreaReq        28
#define X_UngrabButton         29
#define X_GetWindowAttributes   3
#define X_ChangeActivePointerGrab 30
#define X_GrabKeyboard          31
#define X_UngrabKeyboard        32
#define sz_xAllocColorCellsReply 32
#define sz_xAllocColorPlanesReply 32
#define sz_xAllocColorReply     32
#define sz_xAllocNamedColorReply 32
#define sz_xConnSetup          32
#define sz_xCopyPlaneReq       32
#define sz_xCreateCursorReq     32
#define sz_xCreateGlyphCursorReq 32
#define sz_xCreateWindowReq     32
#define sz_xError              32
#define sz_xEvent              32
#define sz_xGenericReply        32
#define sz_xGetAtomNameReply    32
#define sz_xGetFontPathReply    32
#define sz_xGetGeometryReply    32
#define sz_xGetImageReply       32
#define sz_xGetInputFocusReply  32
#define sz_xGetKeyboardMappingReply 32
#define sz_xGetModifierMappingReply 32
#define sz_xGetMotionEventsReply 32
#define sz_xGetPointerControlReply 32
#define sz_xGetPointerMappingReply 32
#define sz_xGetPropertyReply     32
#define sz_xGetScreenSaverReply 32
#define sz_xGetSelectionOwnerReply 32
#define sz_xGrabKeyboardReply    32
#define sz_xGrabPointerReply     32
#define sz_xInternAtomReply      32
#define sz_xKeymapEvent         32
#define sz_xListExtensionsReply  32
#define sz_xListFontsReply       32
#define sz_xListHostsReply       32
#define sz_xListInstalledColormapsReply 32
#define sz_xListPropertiesReply  32
#define sz_xLookupColorReply     32
#define sz_xQueryBestSizeReply   32
#define sz_xQueryColorsReply     32
#define sz_xQueryExtensionReply  32
#define sz_xQueryPointerReply    32
#define sz_xQueryTextExtentsReply 32
#define sz_xQueryTreeReply       32
#define sz_xReply                32
#define sz_xSetMappingReply       32
#define sz_xSetModifierMappingReply 32
#define sz_xSetPointerMappingReply 32

```

```

#define sz_xTranslateCoordsReply      32
#define X_GrabKey                      33
#define X_UngrabKey                    34
#define X_AllowEvents                  35
#define X_GrabServer                   36
#define X_UngrabServer                 37
#define X_QueryPointer                 38
#define X_GetMotionEvents              39
#define X_DestroyWindow               4
#define sz_xBellReq                    4
#define sz_xChangeModeReq              4
#define sz_xForceScreenSaverReq       4
#define sz_xHostEntry                  4
#define sz_xListHostsReq               4
#define sz_xPoint                      4
#define sz_xReq                        4
#define sz_xSetAccessControlReq        4
#define sz_xSetCloseDownModeReq       4
#define sz_xSetModifierMappingReq      4
#define sz_xSetPointerMappingReq       4
#define X_TranslateCoords              40
#define sz_xQueryKeymapReply           40
#define sz_xWindowRoot                 40
#define X_WarpPointer                  41
#define X_SetInputFocus                42
#define X_GetInputFocus                43
#define X_QueryKeymap                  44
#define sz_xGetWindowAttributesReply   44
#define sz_xSendEventReq               44
#define X_OpenFont                     45
#define X_CloseFont                    46
#define X_QueryFont                    47
#define X_QueryTextExtents              48
#define X_ListFonts                     49
#define X_DestroySubwindows             5
#define X_ListFontsWithInfo             50
#define X_SetFontPath                  51
#define X_GetFontPath                  52
#define sz_xGetKeyboardControlReply     52
#define X_CreatePixmap                 53
#define X_FreePixmap                   54
#define X_CreateGC                      55
#define X_ChangeGC                     56
#define X_CopyGC                       57
#define X_SetDashes                     58
#define X_SetClipRectangles             59
#define X_ChangeSaveSet                 6
#define X_FreeGC                        60
#define sz_xListFontsWithInfoReply      60
#define sz_xQueryFontReply              60
#define X_TCP_PORT                      6000
#define X_ClearArea                     61
#define X_CopyArea                      62
#define X_CopyPlane                     63
#define X_PolyPoint                     64
#define X_PolyLine                      65
#define X_PolySegment                   66
#define X_PolyRectangle                 67
#define X_PolyArc                       68
#define X_FillPoly                      69
#define X_ReparentWindow                7
#define X_PolyFillRectangle             70
#define X_PolyFillArc                   71
#define X_PutImage                      72
#define X_GetImage                      73
#define X_PolyText8                     74

```

```

#define X_PolyText16      75
#define X_ImageText8      76
#define X_ImageText16     77
#define X_CreateColormap  78
#define X_FreeColormap    79
#define X_MapWindow       8
#define sz_xAllowEventsReq 8
#define sz_xChangeHostsReq 8
#define sz_xChangeKeyboardControlReq 8
#define sz_xChangeKeyboardMappingReq 8
#define sz_xChangeSaveSetReq 8
#define sz_xCirculateWindowReq 8
#define sz_xConnSetupPrefix 8
#define sz_xDepth         8
#define sz_xFontProp      8
#define sz_xGetKeyboardMappingReq 8
#define sz_xInternAtomReq 8
#define sz_xListFontsReq   8
#define sz_xListFontsWithInfoReq 8
#define sz_xPixmapFormat   8
#define sz_xQueryColorsReq 8
#define sz_xQueryExtensionReq 8
#define sz_xQueryTextExtentsReq 8
#define sz_xRectangle      8
#define sz_xResourceReq    8
#define sz_xSegment        8
#define sz_xSetFontPathReq 8
#define sz_xStoreColorsReq 8
#define sz_xTimecoord      8
#define sz_xrgb            8
#define X_CopyColormapAndFree 80
#define X_InstallColormap  81
#define X_UninstallColormap 82
#define X_ListInstalledColormaps 83
#define X_AllocColor        84
#define X_AllocNamedColor   85
#define X_AllocColorCells   86
#define X_AllocColorPlanes  87
#define X_FreeColors        88
#define X_StoreColors       89
#define X_MapSubwindows     9
#define X_StoreNamedColor   90
#define X_QueryColors       91
#define X_LookupColor       92
#define X_CreateCursor      93
#define X_CreateGlyphCursor 94
#define X_FreeCursor        95
#define X_RecolorCursor     96
#define X_QueryBestSize     97
#define X_QueryExtension    98
#define X_ListExtensions    99

typedef struct _xEvent {
    union {
        struct {
            BYTE type;
            BYTE detail;
            CARD16 sequenceNumber;
        } u;
        struct {
            CARD32 pad00;
            CARD32 time;
            CARD32 root;
            CARD32 event;
            CARD32 child;
            INT16 rootX;
        }
    }
} xEvent;

```

```

        INT16 rootY;
        INT16 eventX;
        INT16 eventY;
        KeyButMask state;
        BOOL sameScreen;
        BYTE pad1;
    } keyButtonPointer;
    struct {
        CARD32 pad00;
        CARD32 time;
        CARD32 root;
        CARD32 event;
        CARD32 child;
        INT16 rootX;
        INT16 rootY;
        INT16 eventX;
        INT16 eventY;
        KeyButMask state;
        BYTE mode;
        BYTE flags;
    } enterLeave;
    struct {
        CARD32 pad00;
        CARD32 window;
        BYTE mode;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } focus;
    struct {
        CARD32 pad00;
        CARD32 window;
        CARD16 x;
        CARD16 y;
        CARD16 width;
        CARD16 height;
        CARD16 count;
        CARD16 pad2;
    } expose;
    struct {
        CARD32 pad00;
        CARD32 drawable;
        CARD16 x;
        CARD16 y;
        CARD16 width;
        CARD16 height;
        CARD16 minorEvent;
        CARD16 count;
        BYTE majorEvent;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } graphicsExposure;
    struct {
        CARD32 pad00;
        CARD32 drawable;
        CARD16 minorEvent;
        BYTE majorEvent;
        BYTE bpad;
    } noExposure;
    struct {
        CARD32 pad00;
        CARD32 window;
        CARD8 state;
        BYTE pad1;
        BYTE pad2;
    }

```

```

        BYTE pad3;
    } visibility;
    struct {
        CARD32 pad00;
        CARD32 parent;
        CARD32 window;
        INT16 x;
        INT16 y;
        CARD16 width;
        CARD16 height;
        CARD16 borderWidth;
        BOOL override;
        BYTE bpad;
    } createNotify;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
    } destroyNotify;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
        BOOL fromConfigure;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } unmapNotify;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
        BOOL override;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } mapNotify;
    struct {
        CARD32 pad00;
        CARD32 parent;
        CARD32 window;
    } mapRequest;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
        CARD32 parent;
        INT16 x;
        INT16 y;
        BOOL override;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } reparent;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
        CARD32 aboveSibling;
        INT16 x;
        INT16 y;
        CARD16 width;
        CARD16 height;
        CARD16 borderWidth;
        BOOL override;
        BYTE bpad;
    }

```

```

    } configureNotify;
    struct {
        CARD32 pad00;
        CARD32 parent;
        CARD32 window;
        CARD32 sibling;
        INT16 x;
        INT16 y;
        CARD16 width;
        CARD16 height;
        CARD16 borderWidth;
        CARD16 valueMask;
        CARD32 pad1;
    } configureRequest;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
        INT16 x;
        INT16 y;
        CARD32 pad1;
        CARD32 pad2;
        CARD32 pad3;
        CARD32 pad4;
    } gravity;
    struct {
        CARD32 pad00;
        CARD32 window;
        CARD16 width;
        CARD16 height;
    } resizeRequest;
    struct {
        CARD32 pad00;
        CARD32 event;
        CARD32 window;
        CARD32 parent;
        BYTE place;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } circulate;
    struct {
        CARD32 pad00;
        CARD32 window;
        CARD32 atom;
        CARD32 time;
        BYTE state;
        BYTE pad1;
        CARD16 pad2;
    } property;
    struct {
        CARD32 pad00;
        CARD32 time;
        CARD32 window;
        CARD32 atom;
    } selectionClear;
    struct {
        CARD32 pad00;
        CARD32 time;
        CARD32 owner;
        CARD32 requestor;
        CARD32 selection;
        CARD32 target;
        CARD32 property;
    } selectionRequest;
    struct {

```

```

        CARD32 pad00;
        CARD32 time;
        CARD32 requestor;
        CARD32 selection;
        CARD32 target;
        CARD32 property;
    } selectionNotify;
    struct {
        CARD32 pad00;
        CARD32 window;
        CARD32 colormap;
#ifdef __cplusplus || defined(cplusplus)
        BOOL c_new;
#else
        BOOL new;
#endif
        BYTE state;
        BYTE pad1;
        BYTE pad2;
    } colormap;
    struct {
        CARD32 pad00;
        CARD8 request;
        CARD8 firstKeyCode;
        CARD8 count;
        BYTE pad1;
    } mappingNotify;
    struct {
        CARD32 pad00;
        CARD32 window;
        union {
            struct {
                CARD32 type;
                INT32 longs0;
                INT32 longs1;
                INT32 longs2;
                INT32 longs3;
                INT32 longs4;
            } l;
            struct {
                CARD32 type;
                INT16 shorts0;
                INT16 shorts1;
                INT16 shorts2;
                INT16 shorts3;
                INT16 shorts4;
                INT16 shorts5;
                INT16 shorts6;
                INT16 shorts7;
                INT16 shorts8;
                INT16 shorts9;
            } s;
            struct {
                CARD32 type;
                INT8 bytes[20];
            } b;
        } u;
    } clientMessage;
    } u;
} xEvent;
typedef struct {
    BYTE type;
    BYTE errorCode;
    CARD16 sequenceNumber;
    CARD32 resourceID;
    CARD16 minorCode;

```

```

        CARD8 majorCode;
        BYTE pad1;
        CARD32 pad3;
        CARD32 pad4;
        CARD32 pad5;
        CARD32 pad6;
        CARD32 pad7;
    } xError;
typedef CARD16 KeyButMask;
typedef struct {
    CARD8 byteOrder;
    BYTE pad;
    CARD16 majorVersion;
    CARD16 minorVersion;
    CARD16 nbytesAuthProto;
    CARD16 nbytesAuthString;
    CARD16 pad2;
} xConnClientPrefix;
typedef struct {
    CARD8 success;
    BYTE lengthReason;
    CARD16 majorVersion;
    CARD16 minorVersion;
    CARD16 length;
} xConnSetupPrefix;
typedef struct {
    CARD32 release;
    CARD32 ridBase;
    CARD32 ridMask;
    CARD32 motionBufferSize;
    CARD16 nbytesVendor;
    CARD16 maxRequestSize;
    CARD8 numRoots;
    CARD8 numFormats;
    CARD8 imageByteOrder;
    CARD8 bitmapBitOrder;
    CARD8 bitmapScanlineUnit;
    CARD8 bitmapScanlinePad;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    CARD32 pad2;
} xConnSetup;
typedef struct {
    CARD8 depth;
    CARD8 bitsPerPixel;
    CARD8 scanLinePad;
    CARD8 pad1;
    CARD32 pad2;
} xPixmapFormat;
typedef struct {
    CARD8 depth;
    CARD8 pad1;
    CARD16 nVisuals;
    CARD32 pad2;
} xDepth;
typedef struct {
    CARD32 visualID;
#ifdef __cplusplus || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 bitsPerRGB;
    CARD16 colormapEntries;
    CARD32 redMask;
    CARD32 greenMask;

```



```

        CARD32 blueMask;
        CARD32 pad;
    } xVisualType;
typedef struct {
    CARD32 windowId;
    CARD32 defaultColormap;
    CARD32 whitePixel;
    CARD32 blackPixel;
    CARD32 currentInputMask;
    CARD16 pixWidth;
    CARD16 pixHeight;
    CARD16 mmWidth;
    CARD16 mmHeight;
    CARD16 minInstalledMaps;
    CARD16 maxInstalledMaps;
    CARD32 rootVisualID;
    CARD8 backingStore;
    BOOL saveUnders;
    CARD8 rootDepth;
    CARD8 nDepths;
} xWindowRoot;
typedef struct {
    CARD32 time;
    INT16 x;
    INT16 y;
} xTimecoord;
typedef struct {
    CARD8 family;
    BYTE pad;
    CARD16 length;
} xHostEntry;
typedef struct {
    INT16 leftSideBearing;
    INT16 rightSideBearing;
    INT16 characterWidth;
    INT16 ascent;
    INT16 descent;
    CARD16 attributes;
} xCharInfo;
typedef struct {
    CARD32 name;
    CARD32 value;
} xFontProp;
typedef struct {
    CARD8 len;
    INT8 delta;
} xTextEl;
typedef struct {
    CARD32 pixel;
    CARD16 red;
    CARD16 green;
    CARD16 blue;
    CARD8 flags;
    CARD8 pad;
} xColorItem;
typedef struct {
    CARD16 red;
    CARD16 green;
    CARD16 blue;
    CARD16 pad;
} xrgb;
typedef unsigned char KEYCODE;
typedef struct {
    BYTE type;
    BYTE data1;
    CARD16 sequenceNumber;

```

```

        CARD32 length;
        CARD32 data00;
        CARD32 data01;
        CARD32 data02;
        CARD32 data03;
        CARD32 data04;
        CARD32 data05;
    } xGenericReply;
typedef struct {
    BYTE type;
    CARD8 backingStore;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 visualID;
#if defined(__cplusplus) || defined(c_plusplus)
    CARD16 c_class;
#else
    CARD16 class;
#endif
    CARD8 bitGravity;
    CARD8 winGravity;
    CARD32 backingBitPlanes;
    CARD32 backingPixel;
    BOOL saveUnder;
    BOOL mapInstalled;
    CARD8 mapState;
    BOOL override;
    CARD32 colormap;
    CARD32 allEventMasks;
    CARD32 yourEventMask;
    CARD16 doNotPropagateMask;
    CARD16 pad;
} xGetWindowAttributesReply;
typedef struct {
    BYTE type;
    CARD8 depth;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 root;
    INT16 x;
    INT16 y;
    CARD16 width;
    CARD16 height;
    CARD16 borderWidth;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
} xGetGeometryReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 root;
    CARD32 parent;
    CARD16 nChildren;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xQueryTreeReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;

```

```

        CARD32 atom;
        CARD32 pad2;
        CARD32 pad3;
        CARD32 pad4;
        CARD32 pad5;
        CARD32 pad6;
    } xInternAtomReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nameLength;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xGetAtomNameReply;
typedef struct {
    BYTE type;
    CARD8 format;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 propertyType;
    CARD32 bytesAfter;
    CARD32 nItems;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
} xGetPropertyReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nProperties;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xListPropertiesReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 owner;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetSelectionOwnerReply;
typedef struct {
    BYTE type;
    BYTE status;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;

```

```

        CARD32 pad5;
        CARD32 pad6;
    } xGrabPointerReply;
typedef xGrabPointerReply xGrabKeyboardReply;
typedef struct {
    BYTE type;
    BOOL sameScreen;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 root;
    CARD32 child;
    INT16 rootX;
    INT16 rootY;
    INT16 winX;
    INT16 winY;
    CARD16 mask;
    CARD16 pad1;
    CARD32 pad;
} xQueryPointerReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 nEvents;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetMotionEventsReply;
typedef struct {
    BYTE type;
    BOOL sameScreen;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 child;
    INT16 dstX;
    INT16 dstY;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xTranslateCoordsReply;
typedef struct {
    BYTE type;
    CARD8 revertTo;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 focus;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xGetInputFocusReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    BYTE map[32];
} xQueryKeymapReply;
typedef struct _xQueryFontReply {
    BYTE type;
    BYTE pad1;

```

```

CARD16 sequenceNumber;
CARD32 length;
xCharInfo minBounds;
CARD32 walign1;
xCharInfo maxBounds;
CARD32 walign2;
CARD16 minCharOrByte2;
CARD16 maxCharOrByte2;
CARD16 defaultChar;
CARD16 nFontProps;
CARD8 drawDirection;
CARD8 minBytel;
CARD8 maxBytel;
BOOL allCharsExist;
INT16 fontAscent;
INT16 fontDescent;
CARD32 nCharInfos;
} xQueryFontReply;
typedef struct {
    BYTE type;
    CARD8 drawDirection;
    CARD16 sequenceNumber;
    CARD32 length;
    INT16 fontAscent;
    INT16 fontDescent;
    INT16 overallAscent;
    INT16 overallDescent;
    INT32 overallWidth;
    INT32 overallLeft;
    INT32 overallRight;
    CARD32 pad;
} xQueryTextExtentsReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nFonts;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xListFontsReply;
typedef struct {
    BYTE type;
    CARD8 nameLength;
    CARD16 sequenceNumber;
    CARD32 length;
    xCharInfo minBounds;
    CARD32 walign1;
    xCharInfo maxBounds;
    CARD32 walign2;
    CARD16 minCharOrByte2;
    CARD16 maxCharOrByte2;
    CARD16 defaultChar;
    CARD16 nFontProps;
    CARD8 drawDirection;
    CARD8 minBytel;
    CARD8 maxBytel;
    BOOL allCharsExist;
    INT16 fontAscent;
    INT16 fontDescent;
    CARD32 nReplies;
} xListFontsWithInfoReply;

```

```

typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nPaths;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xGetFontPathReply;
typedef struct {
    BYTE type;
    CARD8 depth;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 visual;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xGetImageReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nColormaps;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xListInstalledColormapsReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 red;
    CARD16 green;
    CARD16 blue;
    CARD16 pad2;
    CARD32 pixel;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xAllocColorReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pixel;
    CARD16 exactRed;
    CARD16 exactGreen;
    CARD16 exactBlue;
    CARD16 screenRed;
    CARD16 screenGreen;
    CARD16 screenBlue;
    CARD32 pad2;
    CARD32 pad3;

```

```

} xAllocNamedColorReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nPixels;
    CARD16 nMasks;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xAllocColorCellsReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nPixels;
    CARD16 pad2;
    CARD32 redMask;
    CARD32 greenMask;
    CARD32 blueMask;
    CARD32 pad3;
    CARD32 pad4;
} xAllocColorPlanesReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nColors;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xQueryColorsReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 exactRed;
    CARD16 exactGreen;
    CARD16 exactBlue;
    CARD16 screenRed;
    CARD16 screenGreen;
    CARD16 screenBlue;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xLookupColorReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 width;
    CARD16 height;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;

```

```

        CARD32 pad7;
    } xQueryBestSizeReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    BOOL present;
    CARD8 major_opcode;
    CARD8 first_event;
    CARD8 first_error;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xQueryExtensionReply;
typedef struct {
    BYTE type;
    CARD8 nExtensions;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xListExtensionsReply;
typedef struct {
    BYTE type;
    CARD8 success;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xSetMappingReply;
typedef xSetMappingReply xSetPointerMappingReply;
typedef xSetMappingReply xSetModifierMappingReply;
typedef struct {
    BYTE type;
    CARD8 nElts;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xGetPointerMappingReply;
typedef struct {
    BYTE type;
    CARD8 keySymsPerKeyCode;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;

```



```

} xGetKeyboardMappingReply;
typedef struct {
    BYTE type;
    CARD8 numKeyPerModifier;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetModifierMappingReply;
typedef struct {
    BYTE type;
    BOOL globalAutoRepeat;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 ledMask;
    CARD8 keyClickPercent;
    CARD8 bellPercent;
    CARD16 bellPitch;
    CARD16 bellDuration;
    CARD16 pad;
    BYTE map[32];
} xGetKeyboardControlReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 accelNumerator;
    CARD16 accelDenominator;
    CARD16 threshold;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetPointerControlReply;
typedef struct {
    BYTE type;
    BYTE pad1;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 timeout;
    CARD16 interval;
    BOOL preferBlanking;
    BOOL allowExposures;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetScreenSaverReply;
typedef struct {
    BYTE type;
    BOOL enabled;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nHosts;
    CARD16 pad1;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;

```

```

        CARD32 pad7;
    } xListHostsReply;
typedef struct {
    BYTE type;
    BYTE map[31];
} xKeymapEvent;
typedef struct _xReq {
    CARD8 reqType;
    CARD8 data;
    CARD16 length;
} xReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 id;
} xResourceReq;
typedef struct {
    CARD8 reqType;
    CARD8 depth;
    CARD16 length;
    CARD32 wid;
    CARD32 parent;
    INT16 x;
    INT16 y;
    CARD16 width;
    CARD16 height;
    CARD16 borderWidth;
#if defined(__cplusplus) || defined(c_plusplus)
    CARD16 c_class;
#else
    CARD16 class;
#endif
} xCreateWindowReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 window;
    CARD32 valueMask;
} xChangeWindowAttributesReq;
typedef struct {
    CARD8 reqType;
    BYTE mode;
    CARD16 length;
    CARD32 window;
} xChangeSaveSetReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 window;
    CARD32 parent;
    INT16 x;
    INT16 y;
} xReparentWindowReq;
typedef struct {
    CARD8 reqType;
    CARD8 pad;
    CARD16 length;
    CARD32 window;
    CARD16 mask;
    CARD16 pad2;
} xConfigureWindowReq;

```

```

typedef struct {
    CARD8 reqType;
    CARD8 direction;
    CARD16 length;
    CARD32 window;
} xCirculateWindowReq;
typedef struct {
    CARD8 reqType;
    BOOL onlyIfExists;
    CARD16 length;
    CARD16 nbytes;
    CARD16 pad;
} xInternAtomReq;
typedef struct {
    CARD8 reqType;
    CARD8 mode;
    CARD16 length;
    CARD32 window;
    CARD32 property;
    CARD32 type;
    CARD8 format;
    BYTE pad[3];
    CARD32 nUnits;
} xChangePropertyReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 window;
    CARD32 property;
} xDeletePropertyReq;
typedef struct {
    CARD8 reqType;
#ifdef __cplusplus || defined(c_plusplus)
    BOOL c_delete;
#else
    BOOL delete;
#endif
    CARD16 length;
    CARD32 window;
    CARD32 property;
    CARD32 type;
    CARD32 longOffset;
    CARD32 longLength;
} xGetPropertyReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 window;
    CARD32 selection;
    CARD32 time;
} xSetSelectionOwnerReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 requestor;
    CARD32 selection;
    CARD32 target;
    CARD32 property;
    CARD32 time;
} xConvertSelectionReq;
typedef struct {
    CARD8 reqType;
    BOOL ownerEvents;

```

```

        CARD16 length;
        CARD32 grabWindow;
        CARD16 eventMask;
        BYTE pointerMode;
        BYTE keyboardMode;
        CARD32 confineTo;
        CARD32 cursor;
        CARD32 time;
    } xGrabPointerReq;
typedef struct {
    CARD8 reqType;
    BOOL ownerEvents;
    CARD16 length;
    CARD32 grabWindow;
    CARD16 eventMask;
    BYTE pointerMode;
    BYTE keyboardMode;
    CARD32 confineTo;
    CARD32 cursor;
    CARD8 button;
    BYTE pad;
    CARD16 modifiers;
} xGrabButtonReq;
typedef struct {
    CARD8 reqType;
    CARD8 button;
    CARD16 length;
    CARD32 grabWindow;
    CARD16 modifiers;
    CARD16 pad;
} xUngrabButtonReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cursor;
    CARD32 time;
    CARD16 eventMask;
    CARD16 pad2;
} xChangeActivePointerGrabReq;
typedef struct {
    CARD8 reqType;
    BOOL ownerEvents;
    CARD16 length;
    CARD32 grabWindow;
    CARD32 time;
    BYTE pointerMode;
    BYTE keyboardMode;
    CARD16 pad;
} xGrabKeyboardReq;
typedef struct {
    CARD8 reqType;
    BOOL ownerEvents;
    CARD16 length;
    CARD32 grabWindow;
    CARD16 modifiers;
    CARD8 key;
    BYTE pointerMode;
    BYTE keyboardMode;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xGrabKeyReq;
typedef struct {
    CARD8 reqType;
    CARD8 key;

```

```

        CARD16 length;
        CARD32 grabWindow;
        CARD16 modifiers;
        CARD16 pad;
    } xUngrabKeyReq;
typedef struct {
    CARD8 reqType;
    CARD8 mode;
    CARD16 length;
    CARD32 time;
} xAllowEventsReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 window;
    CARD32 start;
    CARD32 stop;
} xGetMotionEventsReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 srcWid;
    CARD32 dstWid;
    INT16 srcX;
    INT16 srcY;
} xTranslateCoordsReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 srcWid;
    CARD32 dstWid;
    INT16 srcX;
    INT16 srcY;
    CARD16 srcWidth;
    CARD16 srcHeight;
    INT16 dstX;
    INT16 dstY;
} xWarpPointerReq;
typedef struct {
    CARD8 reqType;
    CARD8 revertTo;
    CARD16 length;
    CARD32 focus;
    CARD32 time;
} xSetInputFocusReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 fid;
    CARD16 nbytes;
    BYTE pad1;
    BYTE pad2;
} xOpenFontReq;
typedef struct {
    CARD8 reqType;
    BOOL oddLength;
    CARD16 length;
    CARD32 fid;
} xQueryTextExtentsReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;

```

```

        CARD16 length;
        CARD16 maxNames;
        CARD16 nbytes;
    } xListFontsReq;
typedef xListFontsReq xListFontsWithInfoReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD16 nFonts;
    BYTE pad1;
    BYTE pad2;
} xSetFontPathReq;
typedef struct {
    CARD8 reqType;
    CARD8 depth;
    CARD16 length;
    CARD32 pid;
    CARD32 drawable;
    CARD16 width;
    CARD16 height;
} xCreatePixmapReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 gc;
    CARD32 drawable;
    CARD32 mask;
} xCreateGCReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 gc;
    CARD32 mask;
} xChangeGCReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 srcGC;
    CARD32 dstGC;
    CARD32 mask;
} xCopyGCReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 gc;
    CARD16 dashOffset;
    CARD16 nDashes;
} xSetDashesReq;
typedef struct {
    CARD8 reqType;
    BYTE ordering;
    CARD16 length;
    CARD32 gc;
    INT16 xOrigin;
    INT16 yOrigin;
} xSetClipRectanglesReq;
typedef struct {
    CARD8 reqType;
    BOOL exposures;
    CARD16 length;
    CARD32 window;

```

```

    INT16 x;
    INT16 y;
    CARD16 width;
    CARD16 height;
} xClearAreaReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 srcDrawable;
    CARD32 dstDrawable;
    CARD32 gc;
    INT16 srcX;
    INT16 srcY;
    INT16 dstX;
    INT16 dstY;
    CARD16 width;
    CARD16 height;
} xCopyAreaReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 srcDrawable;
    CARD32 dstDrawable;
    CARD32 gc;
    INT16 srcX;
    INT16 srcY;
    INT16 dstX;
    INT16 dstY;
    CARD16 width;
    CARD16 height;
    CARD32 bitPlane;
} xCopyPlaneReq;
typedef struct {
    CARD8 reqType;
    BYTE coordMode;
    CARD16 length;
    CARD32 drawable;
    CARD32 gc;
} xPolyPointReq;
typedef xPolyPointReq xPolyLineReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 drawable;
    CARD32 gc;
} xPolySegmentReq;
typedef xPolySegmentReq xPolyArcReq;
typedef xPolySegmentReq xPolyRectangleReq;
typedef xPolySegmentReq xPolyFillRectangleReq;
typedef xPolySegmentReq xPolyFillArcReq;
typedef struct _FillPolyReq {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 drawable;
    CARD32 gc;
    BYTE shape;
    BYTE coordMode;
    CARD16 pad1;
} xFillPolyReq;
typedef struct _PutImageReq {
    CARD8 reqType;
    CARD8 format;

```

```

        CARD16 length;
        CARD32 drawable;
        CARD32 gc;
        CARD16 width;
        CARD16 height;
        INT16 dstX;
        INT16 dstY;
        CARD8 leftPad;
        CARD8 depth;
        CARD16 pad;
    } xPutImageReq;
typedef struct {
    CARD8 reqType;
    CARD8 format;
    CARD16 length;
    CARD32 drawable;
    INT16 x;
    INT16 y;
    CARD16 width;
    CARD16 height;
    CARD32 planeMask;
} xGetImageReq;
typedef struct {
    CARD8 reqType;
    CARD8 pad;
    CARD16 length;
    CARD32 drawable;
    CARD32 gc;
    INT16 x;
    INT16 y;
} xPolyTextReq;
typedef xPolyTextReq xPolyText8Req;
typedef xPolyTextReq xPolyText16Req;
typedef struct {
    CARD8 reqType;
    BYTE nChars;
    CARD16 length;
    CARD32 drawable;
    CARD32 gc;
    INT16 x;
    INT16 y;
} xImageTextReq;
typedef xImageTextReq xImageText8Req;
typedef xImageTextReq xImageText16Req;
typedef struct {
    CARD8 reqType;
    BYTE alloc;
    CARD16 length;
    CARD32 mid;
    CARD32 window;
    CARD32 visual;
} xCreateColormapReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 mid;
    CARD32 srcCmap;
} xCopyColormapAndFreeReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cmap;
    CARD16 red;
    CARD16 green;

```



```

        CARD16 blue;
        CARD16 pad2;
    } xAllocColorReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cmap;
    CARD16 nbytes;
    BYTE pad1;
    BYTE pad2;
} xAllocNamedColorReq;
typedef struct {
    CARD8 reqType;
    BOOL contiguous;
    CARD16 length;
    CARD32 cmap;
    CARD16 colors;
    CARD16 planes;
} xAllocColorCellsReq;
typedef struct {
    CARD8 reqType;
    BOOL contiguous;
    CARD16 length;
    CARD32 cmap;
    CARD16 colors;
    CARD16 red;
    CARD16 green;
    CARD16 blue;
} xAllocColorPlanesReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cmap;
    CARD32 planeMask;
} xFreeColorsReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cmap;
} xStoreColorsReq;
typedef struct {
    CARD8 reqType;
    CARD8 flags;
    CARD16 length;
    CARD32 cmap;
    CARD32 pixel;
    CARD16 nbytes;
    BYTE pad1;
    BYTE pad2;
} xStoreNamedColorReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cmap;
} xQueryColorsReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cmap;
    CARD16 nbytes;
    BYTE pad1;

```

```

        BYTE pad2;
    } xLookupColorReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cid;
    CARD32 source;
    CARD32 mask;
    CARD16 foreRed;
    CARD16 foreGreen;
    CARD16 foreBlue;
    CARD16 backRed;
    CARD16 backGreen;
    CARD16 backBlue;
    CARD16 x;
    CARD16 y;
} xCreateCursorReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cid;
    CARD32 source;
    CARD32 mask;
    CARD16 sourceChar;
    CARD16 maskChar;
    CARD16 foreRed;
    CARD16 foreGreen;
    CARD16 foreBlue;
    CARD16 backRed;
    CARD16 backGreen;
    CARD16 backBlue;
} xCreateGlyphCursorReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 cursor;
    CARD16 foreRed;
    CARD16 foreGreen;
    CARD16 foreBlue;
    CARD16 backRed;
    CARD16 backGreen;
    CARD16 backBlue;
} xRecolorCursorReq;
typedef struct {
    CARD8 reqType;
#ifdef __cplusplus || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD16 length;
    CARD32 drawable;
    CARD16 width;
    CARD16 height;
} xQueryBestSizeReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD16 nbytes;
    BYTE pad1;
    BYTE pad2;
} xQueryExtensionReq;

```

```

typedef struct {
    CARD8 reqType;
    CARD8 numKeyPerModifier;
    CARD16 length;
} xSetModifierMappingReq;
typedef struct {
    CARD8 reqType;
    CARD8 nElt;
    CARD16 length;
} xSetPointerMappingReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD8 firstKeyCode;
    CARD8 count;
    CARD16 pad1;
} xGetKeyboardMappingReq;
typedef struct {
    CARD8 reqType;
    CARD8 keyCodes;
    CARD16 length;
    CARD8 firstKeyCode;
    CARD8 keySymsPerKeyCode;
    CARD16 pad1;
} xChangeKeyboardMappingReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 mask;
} xChangeKeyboardControlReq;
typedef struct {
    CARD8 reqType;
    INT8 percent;
    CARD16 length;
} xBellReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    INT16 accelNum;
    INT16 accelDenum;
    INT16 threshold;
    BOOL doAccel;
    BOOL doThresh;
} xChangePointerControlReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    INT16 timeout;
    INT16 interval;
    BYTE preferBlank;
    BYTE allowExpose;
    CARD16 pad2;
} xSetScreenSaverReq;
typedef struct {
    CARD8 reqType;
    BYTE mode;
    CARD16 length;
    CARD8 hostFamily;
    BYTE pad;
    CARD16 hostLength;
} xChangeHostsReq;
typedef struct {

```

```

        CARD8 reqType;
        BYTE pad;
        CARD16 length;
    } xListHostsReq;
typedef struct {
    CARD8 reqType;
    BYTE mode;
    CARD16 length;
} xChangeModeReq;
typedef xChangeModeReq xSetAccessControlReq;
typedef xChangeModeReq xSetCloseDownModeReq;
typedef xChangeModeReq xForceScreenSaverReq;
typedef struct {
    CARD8 reqType;
    BYTE pad;
    CARD16 length;
    CARD32 window;
    CARD16 nAtoms;
    INT16 nPositions;
} xRotatePropertiesReq;

typedef union {
    xGenericReply generic;
    xGetGeometryReply geom;
    xQueryTreeReply tree;
    xInternAtomReply atom;
    xGetAtomNameReply atomName;
    xGetPropertyReply property;
    xListPropertiesReply listProperties;
    xGetSelectionOwnerReply selection;
    xGrabPointerReply grabPointer;
    xGrabKeyboardReply grabKeyboard;
    xQueryPointerReply pointer;
    xGetMotionEventsReply motionEvents;
    xTranslateCoordsReply coords;
    xGetInputFocusReply inputFocus;
    xQueryTextExtentsReply textExtents;
    xListFontsReply fonts;
    xGetFontPathReply fontPath;
    xGetImageReply image;
    xListInstalledColormapsReply colormaps;
    xAllocColorReply allocColor;
    xAllocNamedColorReply allocNamedColor;
    xAllocColorCellsReply colorCells;
    xAllocColorPlanesReply colorPlanes;
    xQueryColorsReply colors;
    xLookupColorReply lookupColor;
    xQueryBestSizeReply bestSize;
    xQueryExtensionReply extension;
    xListExtensionsReply extensions;
    xSetModifierMappingReply setModifierMapping;
    xGetModifierMappingReply getModifierMapping;
    xSetPointerMappingReply setPointerMapping;
    xGetKeyboardMappingReply getKeyboardMapping;
    xGetPointerMappingReply getPointerMapping;
    xGetPointerControlReply pointerControl;
    xGetScreenSaverReply screenSaver;
    xListHostsReply hosts;
    xError error;
    xEvent event;
} xReply;
typedef struct {
    CARD8 reqType;
    BOOL propagate;
    CARD16 length;
    CARD32 destination;

```

```

    CARD32 eventMask;
    xEvent event;
} xSendEventReq;

```

6.2.15 X11/Xprotostr.h

```

typedef struct _xSegment {
    INT16 x1;
    INT16 y1;
    INT16 x2;
    INT16 y2;
} xSegment;
typedef struct _xPoint {
    INT16 x;
    INT16 y;
} xPoint;
typedef struct _xRectangle {
    INT16 x;
    INT16 y;
    CARD16 width;
    CARD16 height;
} xRectangle;
typedef struct _xArc {
    INT16 x;
    INT16 y;
    CARD16 width;
    CARD16 height;
    INT16 angle1;
    INT16 angle2;
} xArc;

```

6.2.16 X11/Xresource.h

```

#define NULLQUARK ((XrmQuark) 0)

typedef int XrmQuark;
typedef int *XrmQuarkList;
typedef char *XrmString;
typedef enum {
    XrmBindTightly,
    XrmBindLoosely
} XrmBinding; *XrmBindingList;
typedef XrmQuark XrmName;
typedef XrmQuarkList XrmNameList;
typedef XrmQuark XrmClass;
typedef XrmQuarkList XrmClassList;
typedef XrmQuark XrmRepresentation;
typedef struct {
    unsigned int size;
    XPointer addr;
} XrmValue; *XrmValuePtr;
typedef struct _XrmHashBucketRec *XrmHashBucket;
typedef XrmHashBucket *XrmHashTable;
typedef XrmHashTable XrmSearchList[];
typedef struct _XrmHashBucketRec *XrmDatabase;
typedef enum {
    XrmoptionNoArg,
    XrmoptionIsArg,
    XrmoptionStickyArg,
    XrmoptionSepArg,
    XrmoptionResArg,
    XrmoptionSkipArg,
    XrmoptionSkipLine,

```

```

    XrmOptionSkipNArgs
} XrmOptionKind;
typedef struct {
    char *option;
    char *specifier;
    XrmOptionKind argKind;
    XPointer value;
} XrmOptionDescRec, *XrmOptionDescList;
extern void XrmCombineDatabase(XrmDatabase, XrmDatabase *, int);
extern int XrmCombineFileDatabase(const char *, XrmDatabase *, int);
extern void XrmDestroyDatabase(XrmDatabase);
extern int XrmEnumerateDatabase(XrmDatabase, XrmNameList,
XrmClassList,
                                int, int (*)(XrmDatabase *,
XrmBindingList,
                                XrmQuarkList,
                                XrmRepresentation *,
                                XrmValue *, XPointer),
                                XPointer);
extern XrmDatabase XrmGetDatabase(Display *);
extern XrmDatabase XrmGetFileDatabase(const char *);
extern int XrmGetResource(XrmDatabase, const char *, const char *,
char **,
                                XrmValue *);
extern XrmDatabase XrmGetStringDatabase(const char *);
extern const char *XrmLocaleOfDatabase(XrmDatabase);
extern void XrmMergeDatabases(XrmDatabase, XrmDatabase *);
extern void XrmParseCommand(XrmDatabase *, XrmOptionDescList, int,
const char *, int *, char **);
extern XrmQuark XrmPermStringToQuark(const char *);
extern void XrmPutFileDatabase(XrmDatabase, const char *);
extern void XrmPutLineResource(XrmDatabase *, const char *);
extern void XrmPutResource(XrmDatabase *, const char *, const char
*,
                                XrmValue *);
extern void XrmPutStringResource(XrmDatabase *, const char *,
const char *);
extern int XrmQGetResource(XrmDatabase, XrmNameList, XrmClassList,
XrmRepresentation *, XrmValue *);
extern int XrmQGetSearchList(XrmDatabase, XrmNameList,
XrmClassList,
                                XrmSearchList, int);
extern int XrmQGetSearchResource(XrmSearchList, XrmName, XrmClass,
XrmRepresentation *, XrmValue *);
extern void XrmQPutResource(XrmDatabase *, XrmBindingList,
XrmQuarkList,
                                XrmRepresentation, XrmValue *);
extern void XrmQPutStringResource(XrmDatabase *, XrmBindingList,
XrmQuarkList, const char *);
extern XrmString XrmQuarkToString(XrmQuark);
extern void XrmSetDatabase(Display *, XrmDatabase);
extern void XrmStringToBindingQuarkList(const char *,
XrmBindingList,
                                XrmQuarkList);
extern XrmQuark XrmStringToQuark(const char *);
extern void XrmStringToQuarkList(const char *, XrmQuarkList);
extern XrmQuark XrmUniqueQuark(void);

```

6.2.17 X11/Xutil.h

```

#define IsModifierKey(keysym) (((KeySym)(keysym) >= XK_Shift_L)
&& ((KeySym)(keysym) <= XK_Hyper_R) || ((KeySym)(keysym) >=
XK_ISO_Lock) && ((KeySym)(keysym) <= XK_ISO_Last_Group_Lock)) ||
((KeySym)(keysym) == XK_Mode_switch) || ((KeySym)(keysym) ==
XK_Num_Lock))

```

```

#define IsPrivateKeypadKey(keysym)      (((KeySym) (keysym)  >=
0x11000000) && ((KeySym) (keysym) <= 0x1100FFFF))
#define IsKeypadKey(keysym)             (((KeySym) (keysym) >= XK_KP_Space)
&& ((KeySym) (keysym) <= XK_KP_Equal))
#define IsMiscFunctionKey(keysym)      (((KeySym) (keysym)  >=
XK_Select) && ((KeySym) (keysym) <= XK_Break))
#define IsFunctionKey(keysym)          (((KeySym) (keysym) >= XK_F1) && ((KeySym) (keysym) <= XK_F35))
#define IsCursorKey(keysym)            (((KeySym) (keysym) >= XK_Home) && ((KeySym) (keysym) < XK_Select))
#define IsPFKey(keysym)                (((KeySym) (keysym) >= XK_KP_F1) && ((KeySym) (keysym) <= XK_KP_F4))
#define XStringToContext(string)        XStringToContext(string)
((XContext) XrmStringToQuark(string))
#define XUniqueContext()               ((XContext) XrmUniqueQuark())
#define ReleaseByFreeingColormap       ((XID) 1L)
#define InputHint                      (1L<<0)
#define USPosition                     (1L<<0)
#define StateHint                      (1L<<1)
#define USSize                         (1L<<1)
#define IconPixmapHint                 (1L<<2)
#define PPosition                      (1L<<2)
#define IconWindowHint                 (1L<<3)
#define PSize                          (1L<<3)
#define IconPositionHint                (1L<<4)
#define PMinSize                       (1L<<4)
#define IconMaskHint                   (1L<<5)
#define PMaxSize                       (1L<<5)
#define PResizeInc                     (1L<<6)
#define WindowGroupHint                (1L<<6)
#define PAspect                        (1L<<7)
#define PBaseSize                      (1L<<8)
#define XUrgencyHint                   (1L<<8)
#define PWinGravity                     (1L<<9)
#define AllHints                       (InputHint|StateHint|IconPixmapHint|IconWindowHint|IconPositionHi
nt|IconMaskHint|WindowGroupHint)
#define PAllHints                      (PPosition|PSize|PMinSize|PMaxSize|PResizeInc|PAspect)
#define XNoMemory                      -1
#define XLocaleNotSupported             -2
#define XConverterNotFound              -3
#define BitmapSuccess                  0
#define DontCareState                  0
#define RectangleOut                   0
#define WithdrawnState                 0
#define XCSUCCESS                      0
#define VisualNoMask                   0x0
#define NoValue                        0x0000
#define XValue                         0x0001
#define YValue                         0x0002
#define WidthValue                     0x0004
#define HeightValue                    0x0008
#define AllValues                      0x000F
#define XNegative                      0x0010
#define YNegative                      0x0020
#define VisualIDMask                   0x1
#define VisualRedMaskMask              0x10
#define VisualBitsPerRGBMask           0x100
#define VisualAllMask                   0x1FF
#define VisualScreenMask               0x2
#define VisualGreenMaskMask            0x20
#define VisualDepthMask                0x4
#define VisualBlueMaskMask             0x40
#define VisualClassMask                0x8
#define VisualColormapSizeMask         0x80

```

```

#define BitmapOpenFailed      1
#define NormalState          1
#define RectangleIn          1
#define XCNOEMEM             1
#define BitmapFileInvalid    2
#define RectanglePart        2
#define XCNOENT              2
#define ZoomState            2
#define BitmapNoMemory       3
#define IconicState          3
#define InactiveState        4

typedef struct {
    long int flags;
    int x;
    int y;
    int width;
    int height;
    int min_width;
    int min_height;
    int max_width;
    int max_height;
    int width_inc;
    int height_inc;
    struct {
        int x;
        int y;
    } min_aspect;
    struct {
        int x;
        int y;
    } max_aspect;
    int base_width;
    int base_height;
    int win_gravity;
} XSizeHints;
typedef struct {
    long int flags;
    int input;
    int initial_state;
    Pixmap icon_pixmap;
    Window icon_window;
    int icon_x;
    int icon_y;
    Pixmap icon_mask;
    XID window_group;
} XWMHints;
typedef struct {
    unsigned char *value;
    Atom encoding;
    int format;
    unsigned long int nitems;
} XTextProperty;
typedef struct {
    int min_width;
    int min_height;
    int max_width;
    int max_height;
    int width_inc;
    int height_inc;
} XIconSize;
typedef struct {
    char *res_name;
    char *res_class;
} XClassHint;
typedef struct _XComposeStatus {

```



```

    XPointer compose_ptr;
    int chars_matched;
} XComposeStatus;
typedef struct _XRegion *Region;
typedef struct {
    Visual *visual;
    VisualID visualid;
    int screen;
    int depth;
#if defined(__cplusplus) || defined(c_plusplus)
    int c_class;
#else
    int class;
#endif
    unsigned long int red_mask;
    unsigned long int green_mask;
    unsigned long int blue_mask;
    int colormap_size;
    int bits_per_rgb;
} XVisualInfo;
typedef struct {
    Colormap colormap;
    unsigned long int red_max;
    unsigned long int red_mult;
    unsigned long int green_max;
    unsigned long int green_mult;
    unsigned long int blue_max;
    unsigned long int blue_mult;
    unsigned long int base_pixel;
    VisualID visualid;
    XID killid;
} XStandardColormap;
typedef int XContext;
typedef enum {
    XStringStyle = 0,
    XCompoundTextStyle = 1,
    XTextStyle = 2,
    XStdICCTextStyle = 3,
    XUTF8StringStyle = 4
} XICCEncodingStyle;
extern int XAddPixel(XImage *, long int);
extern XClassHint *XAllocClassHint(void);
extern XIconSize *XAllocIconSize(void);
extern XSizeHints *XAllocSizeHints(void);
extern XStandardColormap *XAllocStandardColormap(void);
extern XWMHints *XAllocWMHints(void);
extern int XClipBox(Region, XRectangle *);
extern void XConvertCase(KeySym, KeySym *, KeySym *);
extern Region XCreateRegion(void);
extern const char *XDefaultString(void);
extern int XDeleteContext(Display *, XID, XContext);
extern int XDestroyImage(XImage *);
extern int XDestroyRegion(Region);
extern int XEmptyRegion(Region);
extern int XEqualRegion(Region, Region);
extern int XFindContext(Display *, XID, XContext, XPointer *);
extern int XGetClassHint(Display *, Window, XClassHint *);
extern int XGetIconSizes(Display *, Window, XIconSize *, int *);
extern int XGetNormalHints(Display *, Window, XSizeHints *);
extern unsigned long int XGetPixel(XImage *, int, int);
extern int XGetRGBColormaps(Display *, Window, XStandardColormap *,
    int *, Atom);
extern int XGetSizeHints(Display *, Window, XSizeHints *, Atom);
extern int XGetStandardColormap(Display *, Window,
    XStandardColormap *,

```

```

Atom);
extern int XGetTextProperty(Display *, Window, XTextProperty *,
Atom);
extern XVisualInfo *XGetVisualInfo(Display *, long int, XVisualInfo
*,
int *);
extern int XGetWMClientMachine(Display *, Window, XTextProperty *);
extern XWMHints *XGetWMHints(Display *, Window);
extern int XGetWMIconName(Display *, Window, XTextProperty *);
extern int XGetWMName(Display *, Window, XTextProperty *);
extern int XGetWMNormalHints(Display *, Window, XSizeHints *, long
int *);
extern int XGetWMSizeHints(Display *, Window, XSizeHints *, long
int *,
Atom);
extern int XGetZoomHints(Display *, Window, XSizeHints *);
extern int XIntersectRegion(Region, Region, Region);
extern int XLookupString(XKeyEvent *, char *, int, KeySym *,
XComposeStatus *);
extern int XMatchVisualInfo(Display *, int, int, int, XVisualInfo
*);
extern int XOffsetRegion(Region, int, int);
extern int XPointInRegion(Region, int, int);
extern Region XPolygonRegion(XPoint *, int, int);
extern int XPutPixel(XImage *, int, int, unsigned long int);
extern int XRectInRegion(Region, int, int, unsigned int, unsigned
int);
extern int XSaveContext(Display *, XID, XContext, const char *);
extern int XSetClassHint(Display *, Window, XClassHint *);
extern int XSetIconSizes(Display *, Window, XIconSize *, int);
extern int XSetNormalHints(Display *, Window, XSizeHints *);
extern void XSetRGBColormaps(Display *, Window, XStandardColormap
*, int,
Atom);
extern int XSetRegion(Display *, GC, Region);
extern int XSetSizeHints(Display *, Window, XSizeHints *, Atom);
extern void XSetStandardColormap(Display *, Window,
XStandardColormap *,
Atom);
extern int XSetStandardProperties(Display *, Window, const char *,
const char *, Pixmap, char **, int,
XSizeHints *);
extern void XSetTextProperty(Display *, Window, XTextProperty *,
Atom);
extern void XSetWMClientMachine(Display *, Window, XTextProperty
*);
extern int XSetWMHints(Display *, Window, XWMHints *);
extern void XSetWMIconName(Display *, Window, XTextProperty *);
extern void XSetWMName(Display *, Window, XTextProperty *);
extern void XSetWMNormalHints(Display *, Window, XSizeHints *);
extern void XSetWMProperties(Display *, Window, XTextProperty *,
XTextProperty *, char **, int, XSizeHints
*,
XWMHints *, XClassHint *);
extern void XSetWMSizeHints(Display *, Window, XSizeHints *, Atom);
extern int XSetZoomHints(Display *, Window, XSizeHints *);
extern int XShrinkRegion(Region, int, int);
extern int XStringListToTextProperty(char **, int, XTextProperty
*);
extern XImage *XSubImage(XImage *, int, int, unsigned int, unsigned
int);
extern int XSubtractRegion(Region, Region, Region);
extern int XTextPropertyToStringList(XTextProperty *, char ***, int
*);
extern int XUnionRectWithRegion(XRectangle *, Region, Region);
extern int XUnionRegion(Region, Region, Region);

```

```

extern int XWMGeometry(Display *, int, const char *, const char *,
                      unsigned int, XSizeHints *, int *, int *, int
*,
                      int *, int *);
extern int XXorRegion(Region, Region, Region);
extern void XmbSetWMProperties(Display *, Window, const char *,
                             const char *, char **, int, XSizeHints *,
                             XWMHints *, XClassHint *);
extern int XmbTextListToTextProperty(Display * display, char **list,
                                     int count, XICCEncodingStyle style,
                                     XTextProperty * text_prop_return);
extern int XmbTextPropertyToTextList(Display * display,
                                     const XTextProperty * text_prop,
                                     char ***list_return,
                                     int *count_return);
extern int Xutf8TextListToTextProperty(Display * display, char
**list,
                                     int count, XICCEncodingStyle style,
                                     XTextProperty * text_prop_return);
extern int Xutf8TextPropertyToTextList(Display * display,
                                     const XTextProperty * text_prop,
                                     char ***list_return,
                                     int *count_return);
extern void XwcFreeStringList(wchar_t * *list);
extern int XwcTextListToTextProperty(Display * display, wchar_t *
*list,
                                     int count, XICCEncodingStyle style,
                                     XTextProperty * text_prop_return);
extern int XwcTextPropertyToTextList(Display * display,
                                     const XTextProperty * text_prop,
                                     wchar_t * **list_return,
                                     int *count_return);

```

6.2.18 X11/cursorfont.h

```

#define _cursorfont_h_
#define XC_X_cursor 0
#define XC_arrow 2
#define XC_based_arrow_down 4
#define XC_based_arrow_up 6
#define XC_boat 8
#define XC_bogosity 10
#define XC_bottom_left_corner 12
#define XC_bottom_right_corner 14
#define XC_bottom_side 16
#define XC_bottom_tee 18
#define XC_box_spiral 20
#define XC_center_ptr 22
#define XC_circle 24
#define XC_clock 26
#define XC_coffee_mug 28
#define XC_cross 30
#define XC_cross_reverse 32
#define XC_crosshair 34
#define XC_diamond_cross 36
#define XC_dot 38
#define XC_dotbox 40
#define XC_double_arrow 42
#define XC_draft_large 44
#define XC_draft_small 46
#define XC_draped_box 48
#define XC_exchange 50
#define XC_fleur 52
#define XC_gobbler 54
#define XC_gumby 56

```

```

#define XC_hand1      58
#define XC_hand2      60
#define XC_heart      62
#define XC_icon 64
#define XC_iron_cross  66
#define XC_left_ptr    68
#define XC_left_side   70
#define XC_left_tee    72
#define XC_leftbutton  74
#define XC_ll_angle    76
#define XC_lr_angle    78
#define XC_man 80
#define XC_middlebutton 82
#define XC_mouse      84
#define XC_pencil     86
#define XC_pirate     88
#define XC_plus 90
#define XC_question_arrow 92
#define XC_right_ptr   94
#define XC_right_side  96
#define XC_right_tee   98
#define XC_rightbutton 100
#define XC_rtl_logo    102
#define XC_sailboat    104
#define XC_sb_down_arrow 106
#define XC_sb_h_double_arrow 108
#define XC_sb_left_arrow 110
#define XC_sb_right_arrow 112
#define XC_sb_up_arrow 114
#define XC_sb_v_double_arrow 116
#define XC_shuttle     118
#define XC_sizing      120
#define XC_spider      122
#define XC_spraycan    124
#define XC_star 126
#define XC_target      128
#define XC_tcross      130
#define XC_top_left_arrow 132
#define XC_top_left_corner 134
#define XC_top_right_corner 136
#define XC_top_side    138
#define XC_top_tee     140
#define XC_trek 142
#define XC_ul_angle    144
#define XC_umbrella    146
#define XC_ur_angle    148
#define XC_watch       150
#define XC_xterm       152
#define XC_num_glyphs  154

```

6.2.19 X11/extensions/XKB.h

```

#define XkbSingleXIClass(c)
(((c) & (~0xff)) == 0) || ((c) == XkbDfltXIClass))
#define XkbSingleXIId(c)
(((c) & (~0xff)) == 0) || ((c) == XkbDfltXIId))
#define XkbBuildCoreState(m,g) (((g) & 0x3) << 13) | ((m) & 0xff)
#define XkbIsPtrAction(a) ((a) -> type == XkbSA_MovePtr) && ((a) -> type <= XkbSA_SetPtrDflt)
#define XkbIsGroupAction(a) (((a) -> type == XkbSA_SetGroup) && ((a) -> type <= XkbSA_LockGroup))
#define XkbIsModAction(a) (((a) -> type == Xkb_SASetMods) && ((a) -> type <= XkbSA_LockMods))
#define XkbExplicitXIClass(c) (((c) & (~0xff)) == 0)
#define XkbExplicitXIDevice(c) (((c) & (~0xff)) == 0)

```

```

#define XkbExplicitXIId(c)      ((c) & (~0xff)) == 0)
#define XkbLegalXIBellClass(c) ((c) == KbdFeedbackClass) ||
(c) == BellFeedbackClass) || ((c) == XkbDfltXIClass) ||
(c) == XkbAllXIClasses))
#define XkbLegalXILedClass(c)  ((c) == KbdFeedbackClass) ||
(c) == LedFeedbackClass) || ((c) == XkbDfltXIClass) ||
(c) == XkbAllXIClasses))
#define XkbIsLegalGroup(g)     ((g) >= 0) && ((g) < XkbNumKbdGroups))
#define XkbGroupForCoreState(s) ((s) >> 13) & 0x3)
#define XkbSA_BreakLatch
((1 << XkbSA_NoAction) | (1 << XkbSA_PtrBtn) | (1 << XkbSA_LockPtrBtn) |
(1 << XkbSA_Terminate) | (1 << XkbSA_SwitchScreen) | (1 << XkbSA_SetControl
s) | (1 << XkbSA_LockControls) | (1 << XkbSA_ActionMessage) |
(1 << XkbSA_RedirectKey) | (1 << XkbSA_DeviceBtn) | (1 << XkbSA_LockDeviceB
tn))
#define XkbSA_ValOp(a) ((a) & XkbSA_ValOpMask)
#define XkbSA_ValScale(a) ((a) & XkbSA_ValScaleMask)
#define XkbIsLegalKeycode(k) ((k) >= XkbMinLegalKeyCode)
#define XkbShiftLevel(n) ((n) - 1)
#define XkbPerKeyBitArraySize ((XkbMaxLegalKeyCode + 1) / 8)
#define XkbSI_NoneOf (0)
#define XkbSA_IgnoreVal (0x00)
#define XkbWrapIntoRange (0x00)
#define XkbAllBooleanCtrlsMask (0x00001FFF)
#define XkbXI_IndicatorsMask (0x001c)
#define XkbXI_AllDeviceFeaturesMask (0x001e)
#define XkbXI_AllFeaturesMask (0x001f)
#define XkbSA_ValScaleMask (0x07)
#define XkbAX_SKOptionsMask (0x0C0)
#define XkbExplicitKeyTypesMask (0x0f)
#define XkbSA_SetValMin (0x10)
#define XkbPCF_AllFlagsMask (0x1F)
#define XkbSA_SetValCenter (0x20)
#define XkbAllCompatMask (0x3)
#define XkbSA_SetValMax (0x30)
#define XkbComponentNamesMask (0x3f)
#define XkbAllNamesMask (0x3fff)
#define XkbAllStateComponentsMask (0x3fff)
#define XkbClampIntoRange (0x40)
#define XkbSA_SetValRelative (0x40)
#define XkbSA_SetValAbsolute (0x50)
#define XkbAllNewKeyboardEventsMask (0x7)
#define XkbSA_ValOpMask (0x70)
#define XkbSA_ISOAffectedMask (0x78)
#define XkbAllComponentsMask (0x7f)
#define XkbSI_OpMask (0x7f)
#define XkbRedirectIntoRange (0x80)
#define XkbSI_LevelOneOnly (0x80)
#define XkbAllExtensionDeviceEventsMask (0x801f)
#define XkbAllAccessXEventsMask (0xf)
#define XkbAllGroupsMask (0xf)
#define XkbAllRequiredTypes (0xf)
#define XkbAX_FBOptionsMask (0xF3F)
#define XkbAllControlsMask (0xF8001FFF)
#define XkbAllExplicitMask (0xff)
#define XkbGBN_AllComponentsMask (0xff)
#define XkbAX_AllOptionsMask (0xFFF)
#define XkbAllEventsMask (0xFFF)
#define XkbAllIndicatorsMask (0xffffffff)
#define XkbAllRadioGroupsMask (0xffffffff)
#define XkbSI_AnyOfOrNone (1)
#define XkbShiftLevelMask(n) (1 << ((n) - 1))
#define XkbDF_DisableLocks (1 << 0)
#define XkbExplicitKeyType1Mask (1 << 0)
#define XkbGroup1Mask (1 << 0)
#define XkbKeyTypesMask (1 << 0)

```

```

#define XkbKeycodesNameMask      (1<<0)
#define XkbOneLevelMask (1<<0)
#define XkbSI_AutoRepeat      (1<<0)
#define XkbSymInterpMask      (1<<0)
#define XkbExplicitKeyType2Mask (1<<1)
#define XkbGeometryNameMask    (1<<1)
#define XkbGroup2Mask (1<<1)
#define XkbGroupCompatMask      (1<<1)
#define XkbKeySymsMask (1<<1)
#define XkbSI_LockingKey      (1<<1)
#define XkbTwoLevelMask (1<<1)
#define XkbKeyAliasesMask      (1<<10)
#define XkbVirtualModNamesMask (1<<11)
#define XkbGroupNamesMask      (1<<12)
#define XkbRGNamesMask (1<<13)
#define XkbAlphabeticMask      (1<<2)
#define XkbExplicitKeyType3Mask (1<<2)
#define XkbGroup3Mask (1<<2)
#define XkbModifierMapMask      (1<<2)
#define XkbSymbolsNameMask      (1<<2)
#define XkbExplicitComponentsMask (1<<3)
#define XkbExplicitKeyType4Mask (1<<3)
#define XkbGroup4Mask (1<<3)
#define XkbKeypadMask (1<<3)
#define XkbPhysSymbolsNameMask (1<<3)
#define XkbExplicitInterpretMask (1<<4)
#define XkbKeyActionsMask      (1<<4)
#define XkbTypesNameMask      (1<<4)
#define XkbCompatNameMask      (1<<5)
#define XkbExplicitAutoRepeatMask (1<<5)
#define XkbKeyBehaviorsMask      (1<<5)
#define XkbExplicitBehaviorMask (1<<6)
#define XkbKeyTypeNamesMask      (1<<6)
#define XkbVirtualModsMask      (1<<6)
#define XkbAnyGroupMask (1<<7)
#define XkbExplicitVModMapMask (1<<7)
#define XkbKTLevelNamesMask      (1<<7)
#define XkbVirtualModMapMask      (1<<7)
#define XkbIndicatorNamesMask      (1<<8)
#define XkbKeyNamesMask (1<<9)
#define XkbAXN_SKPressMask      (1L << 0)
#define XkbAX_SKPressFBMask      (1L << 0)
#define XkbAllActionMessagesMask (1L << 0)
#define XkbAllBellEventsMask      (1L << 0)
#define XkbClientMapMask      (1L << 0)
#define XkbGBN_TypesMask      (1L << 0)
#define XkbIM_UseBase (1L << 0)
#define XkbLC_Hidden (1L << 0)
#define XkbModifierStateMask      (1L << 0)
#define XkbNKN_KeycodesMask      (1L << 0)
#define XkbNewKeyboardNotifyMask (1L << 0)
#define XkbPCF_DetectableAutoRepeatMask (1L << 0)
#define XkbRepeatKeysMask      (1L << 0)
#define XkbSA_ClearLocks      (1L << 0)
#define XkbSA_LockNoLock      (1L << 0)
#define XkbSA_MessageOnPress      (1L << 0)
#define XkbSA_NoAcceleration      (1L << 0)
#define XkbSA_SwitchApplication (1L << 0)
#define XkbXI_KeyboardsMask      (1L << 0)
#define XkbAXN_SKAcceptMask      (1L << 1)
#define XkbAX_SKAcceptFBMask      (1L << 1)
#define XkbGBN_CompatMapMask      (1L << 1)
#define XkbIM_UseLatched      (1L << 1)
#define XkbLC_Default (1L << 1)
#define XkbMapNotifyMask      (1L << 1)
#define XkbModifierBaseMask      (1L << 1)

```

```

#define XkbNKN_GeometryMask      (1L << 1)
#define XkbPCF_GrabsUseXKBStateMask (1L << 1)
#define XkbSA_LatchToLock        (1L << 1)
#define XkbSA_LockNoUnlock        (1L << 1)
#define XkbSA_MessageOnRelease    (1L << 1)
#define XkbSA_MoveAbsoluteX       (1L << 1)
#define XkbServerMapMask          (1L << 1)
#define XkbSlowKeysMask           (1L << 1)
#define XkbXI_ButtonActionsMask   (1L << 1)
#define XkbAX_BKRejectFBMask      (1L << 10)
#define XkbAccessXNotifyMask      (1L << 10)
#define XkbCompatGrabModsMask     (1L << 10)
#define XkbLC_KeypadKeys           (1L << 10)
#define XkbOverlay1Mask            (1L << 10)
#define XkbAX_DumbBellFBMask       (1L << 11)
#define XkbExtensionDeviceNotifyMask (1L << 11)
#define XkbLC_FunctionKeys         (1L << 11)
#define XkbLookupModsMask          (1L << 11)
#define XkbOverlay2Mask            (1L << 11)
#define XkbCompatLookupModsMask    (1L << 12)
#define XkbIgnoreGroupLockMask     (1L << 12)
#define XkbLC_AlternateGroup        (1L << 12)
#define XkbPointerButtonMask        (1L << 13)
#define XkbXI_UnsupportedFeatureMask (1L << 15)
#define XkbAXN_SKRejectMask        (1L << 2)
#define XkbAX_FeatureFBMask        (1L << 2)
#define XkbBounceKeysMask          (1L << 2)
#define XkbCompatMapMask           (1L << 2)
#define XkbGBN_ClientSymbolsMask    (1L << 2)
#define XkbIM_UseLocked             (1L << 2)
#define XkbLC_Partial              (1L << 2)
#define XkbModifierLatchMask        (1L << 2)
#define XkbNKN_DeviceIDMask         (1L << 2)
#define XkbPCF_AutoResetControlsMask (1L << 2)
#define XkbSA_DfltBtnAbsolute        (1L << 2)
#define XkbSA_GroupAbsolute          (1L << 2)
#define XkbSA_MessageGenKeyEvent     (1L << 2)
#define XkbSA_MoveAbsoluteY          (1L << 2)
#define XkbSA_SwitchAbsolute          (1L << 2)
#define XkbSA_UseModMapMods          (1L << 2)
#define XkbStateNotifyMask           (1L << 2)
#define XkbXI_IndicatorNamesMask     (1L << 2)
#define XkbGroupsWrapMask            (1L << 27)
#define XkbInternalModsMask          (1L << 28)
#define XkbIgnoreLockModsMask        (1L << 29)
#define XkbAXN_SKReleaseMask          (1L << 3)
#define XkbAX_SlowWarnFBMask          (1L << 3)
#define XkbControlsNotifyMask         (1L << 3)
#define XkbGBN_ServerSymbolsMask      (1L << 3)
#define XkbIM_UseEffective             (1L << 3)
#define XkbIndicatorMapMask           (1L << 3)
#define XkbModifierLockMask           (1L << 3)
#define XkbPCF_LookupStateWhenGrabbed (1L << 3)
#define XkbSA_ISONoAffectCtrls        (1L << 3)
#define XkbStickyKeysMask             (1L << 3)
#define XkbXI_IndicatorMapsMask       (1L << 3)
#define XkbPerKeyRepeatMask           (1L << 30)
#define XkbControlsEnabledMask        (1L << 31)
#define XkbAXN_BKAcceptMask           (1L << 4)
#define XkbAX_IndicatorFBMask         (1L << 4)
#define XkbGBN_IndicatorMapMask       (1L << 4)
#define XkbGroupStateMask             (1L << 4)
#define XkbIM_UseCompat              (1L << 4)
#define XkbIndicatorStateNotifyMask    (1L << 4)
#define XkbMouseKeysMask              (1L << 4)
#define XkbNamesMask                 (1L << 4)

```



```

#define XkbPCF_SendEventUsesXKBState (1L << 4)
#define XkbSA_ISONoAffectPtr (1L << 4)
#define XkbXI_IndicatorStateMask (1L << 4)
#define XkbAXN_BKRejectMask (1L << 5)
#define XkbAX_StickyKeysFBMask (1L << 5)
#define XkbGBN_KeyNamesMask (1L << 5)
#define XkbGeometryMask (1L << 5)
#define XkbGroupBaseMask (1L << 5)
#define XkbIM_LEDDrivesKB (1L << 5)
#define XkbIndicatorMapNotifyMask (1L << 5)
#define XkbMouseKeysAccelMask (1L << 5)
#define XkbSA_ISONoAffectGroup (1L << 5)
#define XkbAXN_AXKWarningMask (1L << 6)
#define XkbAX_TwoKeysMask (1L << 6)
#define XkbAccessXKeysMask (1L << 6)
#define XkbControlsMask (1L << 6)
#define XkbGBN_GeometryMask (1L << 6)
#define XkbGroupLatchMask (1L << 6)
#define XkbIM_NoAutomatic (1L << 6)
#define XkbNamesNotifyMask (1L << 6)
#define XkbSA_ISONoAffectMods (1L << 6)
#define XkbAX_LatchToLockMask (1L << 7)
#define XkbAccessXTimeoutMask (1L << 7)
#define XkbCompatMapNotifyMask (1L << 7)
#define XkbGBN_OtherNamesMask (1L << 7)
#define XkbGroupLockMask (1L << 7)
#define XkbIM_NoExplicit (1L << 7)
#define XkbSA_ISODfltIsGroup (1L << 7)
#define XkbAX_SKReleaseFBMask (1L << 8)
#define XkbAccessXFeedbackMask (1L << 8)
#define XkbBellNotifyMask (1L << 8)
#define XkbCompatStateMask (1L << 8)
#define XkbLC_AlphanumericKeys (1L << 8)
#define XkbAX_SKRejectFBMask (1L << 9)
#define XkbActionMessageMask (1L << 9)
#define XkbAudibleBellMask (1L << 9)
#define XkbGrabModsMask (1L << 9)
#define XkbLC_ModifierKeys (1L << 9)
#define XkbSI_AnyOf (2)
#define XkbSI_AllOf (3)
#define XkbSI_Exactly (4)
#define XkbAllMapComponentsMask
(XkbAllClientInfoMask|XkbAllServerInfoMask)
#define XkbNumberEvents (XkbEventCode+1)
#define XkbAllServerInfoMask
(XkbExplicitComponentsMask|XkbKeyActionsMask|XkbKeyBehaviorsMask|
XkbVirtualModsMask|XkbVirtualModMapMask)
#define XkbGBN_SymbolsMask
(XkbGBN_ClientSymbolsMask|XkbGBN_ServerSymbolsMask)
#define XkbIM_UseAnyMods (XkbIM_UseAnyGroup|XkbIM_UseCompat)
#define XkbIM_UseAnyGroup
(XkbIM_UseBase|XkbIM_UseLatched|XkbIM_UseLocked
|XkbIM_UseEffective)
#define XkbAllClientInfoMask
(XkbKeyTypesMask|XkbKeySymsMask|XkbModifierMapMask)
#define XkbNumRequiredTypes (XkbLastRequiredType+1)
#define XkbMaxKeyCount (XkbMaxLegalKeyCode-XkbMinLegalKeyCode+1)
#define XkbMaxSymsPerKey (XkbMaxShiftLevel*XkbNumKbdGroups)
#define XkbMaxKbdGroup (XkbNumKbdGroups-1)
#define XkbSA_NumActions (XkbSA_LastAction+1)
#define XkbAccessXOptionsMask
(XkbStickyKeysMask|XkbAccessXFeedbackMask)
#define X_kbUseExtension 0
#define XkbAXN_SKPress 0
#define XkbEventCode 0
#define XkbGroupIndex 0

```



```

#define XkbIM_UseNone 0
#define XkbKeyboard 0
#define XkbMinorVersion 0
#define XkbNewKeyboardNotify 0
#define XkbNoModifierMask 0
#define XkbOneLevelIndex 0
#define XkbSA_UseDfltButton 0
#define XkbKB_Default 0x00
#define XkbSA_NoAction 0x00
#define XkbKB_Lock 0x01
#define XkbSA_SetMods 0x01
#define XkbUseCoreKbd 0x0100
#define XkbKB_RadioGroup 0x02
#define XkbSA_LatchMods 0x02
#define XkbUseCorePtr 0x0200
#define XkbKB_Overlay1 0x03
#define XkbSA_LockMods 0x03
#define XkbDfltXIClass 0x0300
#define XkbKB_Overlay2 0x04
#define XkbSA_SetGroup 0x04
#define XkbDfltXIId 0x0400
#define XkbSA_LatchGroup 0x05
#define XkbAllXIClasses 0x0500
#define XkbSA_LockGroup 0x06
#define XkbAllXIIds 0x0600
#define XkbSA_MovePtr 0x07
#define XkbSA_PtrBtn 0x08
#define XkbSA_LockPtrBtn 0x09
#define XkbSA_SetPtrDflt 0x0a
#define XkbSA_ISOLock 0x0b
#define XkbSA_Terminate 0x0c
#define XkbSA_SwitchScreen 0x0d
#define XkbSA_SetControls 0x0e
#define XkbSA_LockControls 0x0f
#define XkbSA_ActionMessage 0x10
#define XkbSA_RedirectKey 0x11
#define XkbSA_DeviceBtn 0x12
#define XkbSA_LockDeviceBtn 0x13
#define XkbSA_DeviceValuator 0x14
#define XkbKB_OpMask 0x7f
#define XkbKB_Permanent 0x80
#define XkbKB_RGAllowNone 0x80
#define XkbSA_XFree86Private 0x86
#define XkbErr_BadId 0xfd
#define XkbErr_BadClass 0xfe
#define XkbAllModifiersMask 0xff
#define XkbErr_BadDevice 0xff
#define XkbNoIndicator 0xff
#define XkbNoModifier 0xff
#define XkbNoShape 0xff
#define XkbNoShiftLevel 0xff
#define XkbXINone 0xff00
#define XkbAllVirtualModsMask 0xffff
#define X_kbSelectEvents 1
#define XkbAXN_SKAccept 1
#define XkbGroup2Index 1
#define XkbMajorVersion 1
#define XkbMapNotify 1
#define XkbNumberErrors 1
#define XkbSA_AffectDfltBtn 1
#define XkbTwoLevelIndex 1
#define X_kbGetCompatMap 10
#define XkbAccessXNotify 10
#define XkbGeomPtsPerMM 10
#define X_kbSetDebuggingFlags 101
#define X_kbSetCompatMap 11

```

```

#define XkbExtensionDeviceNotify      11
#define X_kbGetIndicatorState         12
#define XkbRGMaxMembers               12
#define X_kbGetIndicatorMap           13
#define X_kbSetIndicatorMap           14
#define X_kbGetNamedIndicator         15
#define X_kbSetNamedIndicator         16
#define XkbNumVirtualMods             16
#define X_kbGetNames                  17
#define X_kbSetNames                  18
#define X_kbGetGeometry               19
#define XkbAXN_SKReject               2
#define XkbAlphabeticIndex            2
#define XkbGroup3Index               2
#define XkbStateNotify               2
#define X_kbSetGeometry               20
#define X_kbPerClientFlags            21
#define X_kbListComponents             22
#define X_kbGetKbdByName              23
#define X_kbGetDeviceInfo              24
#define X_kbSetDeviceInfo              25
#define XkbAnyGroup                   254
#define XkbAllGroups                  255
#define XkbGeomMaxPriority             255
#define XkbMaxKeyTypes                255
#define XkbMaxLegalKeyCode            255
#define X_kbBell                      3
#define XkbAXN_SKRelease              3
#define XkbControlsNotify             3
#define XkbGeomMaxLabelColors         3
#define XkbGroup4Index                3
#define XkbKeypadIndex                3
#define XkbGeomMaxColors              32
#define XkbMaxRadioGroups             32
#define XkbNumIndicators              32
#define X_kbGetState                  4
#define XkbAXN_BKAccept               4
#define XkbIndicatorStateNotify       4
#define XkbKeyNameLength              4
#define XkbMaxMouseKeysBtn            4
#define XkbNumKbdGroups               4
#define X_kbLatchLockState            5
#define XkbAXN_BKReject               5
#define XkbIndicatorMapNotify         5
#define X_kbGetControls               6
#define XkbAXN_AXKWarning              6
#define XkbActionMessageLength        6
#define XkbNamesNotify                6
#define XkbMaxShiftLevel              63
#define X_kbSetControls               7
#define XkbCompatMapNotify            7
#define X_kbGetMap                    8
#define XkbBellNotify                 8
#define XkbMaxRedirectCount           8
#define XkbMinLegalKeyCode            8
#define XkbNumModifiers               8
#define X_kbSetMap                    9
#define XkbActionMessage              9
#define XkbAllCompatMapEventsMask     XkbAllCompatMask
#define XkbAllControlEventsMask       XkbAllControlsMask
#define XkbAllIndicatorEventsMask     XkbAllIndicatorsMask
#define XkbAllMapEventsMask           XkbAllMapComponentsMask
#define XkbAllNameEventsMask          XkbAllNamesMask
#define XkbAllStateEventsMask         XkbAllStateComponentsMask
#define XkbLastRequiredType           XkbKeypadIndex
#define XkbSA_LastAction               XkbSA_DeviceValuator

```

```
#define XkbName "XKEYBOARD"
```

6.2.20 X11/extensions/XKBgeom.h

```
#define _XKBGEOM_H_
#define XkbLogoDoodadColor(g,d) (&(g)->colors[(d)->color_ndx])
#define XkbShapeDoodadColor(g,d) (&(g)->colors[(d)->color_ndx])
#define XkbTextDoodadColor(g,d) (&(g)->colors[(d)->color_ndx])
#define XkbIndicatorDoodadOffColor(g,d) (&(g)->colors[(d)->off_color_ndx])
#define XkbIndicatorDoodadOnColor(g,d) (&(g)->colors[(d)->on_color_ndx])
#define XkbKeyColor(g,k) (&(g)->colors[(k)->color_ndx])
#define XkbIndicatorDoodadShape(g,d) (&(g)->shapes[(d)->shape_ndx])
#define XkbLogoDoodadShape(g,d) (&(g)->shapes[(d)->shape_ndx])
#define XkbShapeDoodadShape(g,d) (&(g)->shapes[(d)->shape_ndx])
#define XkbKeyShape(g,k) (&(g)->shapes[(k)->shape_ndx])
#define XkbBoundsHeight(b) ((b)->y2)-((b)->y1)
#define XkbSetLogoDoodadColor(g,d,c) ((d)->color_ndx= (c)-&(g)->colors[0])
#define XkbSetShapeDoodadColor(g,d,c) ((d)->color_ndx= (c)-&(g)->colors[0])
#define XkbSetTextDoodadColor(g,d,c) ((d)->color_ndx= (c)-&(g)->colors[0])
#define XkbSetIndicatorDoodadOffColor(g,d,c) ((d)->off_color_ndx= (c)-&(g)->colors[0])
#define XkbSetIndicatorDoodadOnColor(g,d,c) ((d)->on_color_ndx= (c)-&(g)->colors[0])
#define XkbSetIndicatorDoodadShape(g,d,s) ((d)->shape_ndx= (s)-&(g)->shapes[0])
#define XkbSetLogoDoodadShape(g,d,s) ((d)->shape_ndx= (s)-&(g)->shapes[0])
#define XkbSetShapeDoodadShape(g,d,s) ((d)->shape_ndx= (s)-&(g)->shapes[0])
#define XkbGeomColorIndex(g,c) ((int)((c)-&(g)->colors[0]))
#define XkbOutlineIndex(s,o) ((int)((o)-&(s)->outlines[0]))
#define XkbSetKeyColor(g,k,c) ((k)->color_ndx= (c)-&(g)->colors[0])
#define XkbSetKeyShape(g,k,s) ((k)->shape_ndx= (s)-&(g)->shapes[0])
#define XkbGeomAllMask (0x3f)
#define XkbGeomPropertiesMask (1<<0)
#define XkbGeomColorsMask (1<<1)
#define XkbGeomShapesMask (1<<2)
#define XkbGeomSectionsMask (1<<3)
#define XkbGeomDoodadsMask (1<<4)
#define XkbGeomKeyAliasesMask (1<<5)
#define XkbUnknownDoodad 0
#define XkbOutlineDoodad 1
#define XkbSolidDoodad 2
#define XkbTextDoodad 3
#define XkbIndicatorDoodad 4
#define XkbLogoDoodad 5

typedef struct _XkbProperty {
    char *name;
    char *value;
} XkbPropertyRec;
typedef struct _XkbProperty *XkbPropertyPtr;
typedef struct _XkbColor {
    unsigned int pixel;
    char *spec;
```

```

    } XkbColorRec;
typedef struct _XkbColor *XkbColorPtr;
typedef struct _XkbPoint {
    short x;
    short y;
} XkbPointRec;
typedef struct _XkbPoint *XkbPointPtr;
typedef struct _XkbBounds {
    short x1;
    short y1;
    short x2;
    short y2;
} XkbBoundsRec;
typedef struct _XkbBounds *XkbBoundsPtr;
typedef struct _XkbOutline {
    unsigned short num_points;
    unsigned short sz_points;
    unsigned short corner_radius;
    XkbPointPtr points;
} XkbOutlineRec;
typedef struct _XkbOutline *XkbOutlinePtr;
typedef struct _XkbShape {
    Atom name;
    unsigned short num_outlines;
    unsigned short sz_outlines;
    XkbOutlinePtr outlines;
    XkbOutlinePtr approx;
    XkbOutlinePtr primary;
    XkbBoundsRec bounds;
} XkbShapeRec;
typedef struct _XkbShape *XkbShapePtr;
typedef struct _XkbShapeDoodad {
    Atom name;
    unsigned char type;
    unsigned char priority;
    short top;
    short left;
    short angle;
    unsigned short color_ndx;
    unsigned short shape_ndx;
} XkbShapeDoodadRec;
typedef struct _XkbShapeDoodad *XkbShapeDoodadPtr;
typedef struct _XkbTextDoodad {
    Atom name;
    unsigned char type;
    unsigned char priority;
    short top;
    short left;
    short angle;
    short width;
    short height;
    unsigned short color_ndx;
    char *text;
    char *font;
} XkbTextDoodadRec;
typedef struct _XkbTextDoodad *XkbTextDoodadPtr;
typedef struct _XkbIndicatorDoodad {
    Atom name;
    unsigned char type;
    unsigned char priority;
    short top;
    short left;
    short angle;
    unsigned short shape_ndx;
    unsigned short on_color_ndx;
    unsigned short off_color_ndx;

```

```

} XkbIndicatorDoodadRec;
typedef struct _XkbIndicatorDoodad *XkbIndicatorDoodadPtr;
typedef struct _XkbLogoDoodad {
    Atom name;
    unsigned char type;
    unsigned char priority;
    short top;
    short left;
    short angle;
    unsigned short color_ndx;
    unsigned short shape_ndx;
    char *logo_name;
} XkbLogoDoodadRec;
typedef struct _XkbLogoDoodad *XkbLogoDoodadPtr;
typedef struct _XkbAnyDoodad {
    Atom name;
    unsigned char type;
    unsigned char priority;
    short top;
    short left;
    short angle;
} XkbAnyDoodadRec;
typedef struct _XkbAnyDoodad *XkbAnyDoodadPtr;
union _XkbDoodad {
    XkbAnyDoodadRec any;
    XkbShapeDoodadRec shape;
    XkbTextDoodadRec text;
    XkbIndicatorDoodadRec indicator;
    XkbLogoDoodadRec logo;
};
typedef union _XkbDoodad {
    XkbAnyDoodadRec any;
    XkbShapeDoodadRec shape;
    XkbTextDoodadRec text;
    XkbIndicatorDoodadRec indicator;
    XkbLogoDoodadRec logo;
} XkbDoodadRec;
typedef union _XkbDoodad *XkbDoodadPtr;
typedef struct _XkbKey {
    XkbKeyNameRec name;
    short gap;
    unsigned char shape_ndx;
    unsigned char color_ndx;
} XkbKeyRec;
typedef struct _XkbKey *XkbKeyPtr;
typedef struct _XkbRow {
    short top;
    short left;
    unsigned short num_keys;
    unsigned short sz_keys;
    int vertical;
    XkbKeyPtr keys;
    XkbBoundsRec bounds;
} XkbRowRec;
typedef struct _XkbRow *XkbRowPtr;
typedef struct _XkbSection {
    Atom name;
    unsigned char priority;
    short top;
    short left;
    unsigned short width;
    unsigned short height;
    short angle;
    unsigned short num_rows;
    unsigned short num_doodads;
    unsigned short num_overlays;

```

```

        unsigned short sz_rows;
        unsigned short sz_doodads;
        unsigned short sz_overlays;
        XkbRowPtr rows;
        XkbDoodadPtr doodads;
        XkbBoundsRec bounds;
        struct _XkbOverlay *overlays;
    } XkbSectionRec;
typedef struct _XkbSection *XkbSectionPtr;
typedef struct _XkbOverlayKey {
    XkbKeyNameRec over;
    XkbKeyNameRec under;
} XkbOverlayKeyRec;
typedef struct _XkbOverlayKey *XkbOverlayKeyPtr;
typedef struct _XkbOverlayRow {
    unsigned short row_under;
    unsigned short num_keys;
    unsigned short sz_keys;
    XkbOverlayKeyPtr keys;
} XkbOverlayRowRec;
typedef struct _XkbOverlayRow *XkbOverlayRowPtr;
typedef struct _XkbOverlay {
    Atom name;
    XkbSectionPtr section_under;
    unsigned short num_rows;
    unsigned short sz_rows;
    XkbOverlayRowPtr rows;
    XkbBoundsPtr bounds;
} XkbOverlayRec;
typedef struct _XkbOverlay *XkbOverlayPtr;
typedef struct _XkbGeometry {
    Atom name;
    unsigned short width_mm;
    unsigned short height_mm;
    char *label_font;
    XkbColorPtr label_color;
    XkbColorPtr base_color;
    unsigned short sz_properties;
    unsigned short sz_colors;
    unsigned short sz_shapes;
    unsigned short sz_sections;
    unsigned short sz_doodads;
    unsigned short sz_key_aliases;
    unsigned short num_properties;
    unsigned short num_colors;
    unsigned short num_shapes;
    unsigned short num_sections;
    unsigned short num_doodads;
    unsigned short num_key_aliases;
    XkbPropertyPtr properties;
    XkbColorPtr colors;
    XkbShapePtr shapes;
    XkbSectionPtr sections;
    XkbDoodadPtr doodads;
    XkbKeyAliasPtr key_aliases;
} XkbGeometryRec;
typedef struct _XkbGeometrySizes {
    unsigned int which;
    unsigned short num_properties;
    unsigned short num_colors;
    unsigned short num_shapes;
    unsigned short num_sections;
    unsigned short num_doodads;
    unsigned short num_key_aliases;
} XkbGeometrySizesRec;
typedef struct _XkbGeometrySizes *XkbGeometrySizesPtr;

```

```

extern int XkbAllocGeomColors(XkbGeometryPtr, int);
extern int XkbAllocGeomDoodads(XkbGeometryPtr, int);
extern int XkbAllocGeomKeyAliases(XkbGeometryPtr, int);
extern int XkbAllocGeomKeys(XkbRowPtr, int);
extern int XkbAllocGeomOutlines(XkbShapePtr, int);
extern int XkbAllocGeomOverlayKeys(XkbOverlayRowPtr, int);
extern int XkbAllocGeomOverlayRows(XkbOverlayPtr, int);
extern int XkbAllocGeomOverlays(XkbSectionPtr, int);
extern int XkbAllocGeomPoints(XkbOutlinePtr, int);
extern int XkbAllocGeomProps(XkbGeometryPtr, int);
extern int XkbAllocGeomRows(XkbSectionPtr, int);
extern int XkbAllocGeomSectionDoodads(XkbSectionPtr, int);
extern int XkbAllocGeomSections(XkbGeometryPtr, int);
extern int XkbAllocGeomShapes(XkbGeometryPtr, int);
extern int XkbAllocGeometry(XkbDescPtr, XkbGeometrySizesPtr);
extern int XkbComputeRowBounds(XkbGeometryPtr, XkbSectionPtr,
XkbRowPtr);
extern int XkbComputeSectionBounds(XkbGeometryPtr, XkbSectionPtr);
extern int XkbComputeShapeBounds(XkbShapePtr);
extern int XkbComputeShapeTop(XkbShapePtr, XkbBoundsPtr);
extern char *XkbFindOverlayForKey(XkbGeometryPtr, XkbSectionPtr,
char *);
extern void XkbFreeGeomColors(XkbGeometryPtr, int, int, int);
extern void XkbFreeGeomDoodads(XkbDoodadPtr, int, int);
extern void XkbFreeGeomKeyAliases(XkbGeometryPtr, int, int, int);
extern void XkbFreeGeomKeys(XkbRowPtr, int, int, int);
extern void XkbFreeGeomOutlines(XkbShapePtr, int, int, int);
extern void XkbFreeGeomOverlayKeys(XkbOverlayRowPtr, int, int, int);
extern void XkbFreeGeomOverlayRows(XkbOverlayPtr, int, int, int);
extern void XkbFreeGeomOverlays(XkbSectionPtr, int, int, int);
extern void XkbFreeGeomPoints(XkbOutlinePtr, int, int, int);
extern void XkbFreeGeomProperties(XkbGeometryPtr, int, int, int);
extern void XkbFreeGeomRows(XkbSectionPtr, int, int, int);
extern void XkbFreeGeomSections(XkbGeometryPtr, int, int, int);
extern void XkbFreeGeomShapes(XkbGeometryPtr, int, int, int);
extern void XkbFreeGeometry(XkbGeometryPtr, unsigned int, int);
extern int XkbGetGeometry(Display *, XkbDescPtr);
extern int XkbGetNamedGeometry(Display *, XkbDescPtr, Atom);
extern int XkbSetGeometry(Display *, unsigned int, XkbGeometryPtr);

```

6.2.21 X11/extensions/XKBproto.h

```

#define XkbPaddedSize(n) (((unsigned int)(n)+3) >> 2) << 2)
#define XkbSetMapAllFlags (0x3)
#define XkbSetMapResizeTypes (1L<<0)
#define XkbSetMapRecomputeActions (1L<<1)
#define sz_xkbSetControlsReq 100
#define sz_xkbGetCompatMapReq 12
#define sz_xkbGetGeometryReq 12
#define sz_xkbGetIndicatorMapReq 12
#define sz_xkbGetKbdByNameReq 12
#define sz_xkbGetNamesReq 12
#define sz_xkbIndicatorMapWireDesc 12
#define sz_xkbSetDeviceInfoReq 12
#define sz_xkbSetIndicatorMapReq 12
#define sz_xkbGetDeviceInfoReq 16
#define sz_xkbGetNamedIndicatorReq 16
#define sz_xkbLatchLockStateReq 16
#define sz_xkbSelectEventsReq 16
#define sz_xkbSetCompatMapReq 16
#define sz_xkbSymInterpretWireDesc 16
#define sz_xkbAnyDoodadWireDesc 20
#define sz_xkbDeviceLedsWireDesc 20
#define sz_xkbDoodadWireDesc 20
#define sz_xkbIndicatorDoodadWireDesc 20

```

```

#define sz_xkbLogoDoodadWireDesc      20
#define sz_xkbSectionWireDesc      20
#define sz_xkbShapeDoodadWireDesc      20
#define sz_xkbTextDoodadWireDesc      20
#define sz_xkbSetDebuggingFlagsReq      24
#define sz_xkbBellReq      28
#define sz_xkbGetMapReq      28
#define sz_xkbPerClientFlagsReq      28
#define sz_xkbSetGeometryReq      28
#define sz_xkbSetNamesReq      28
#define sz_xkbAccessXNotify      32
#define sz_xkbActionMessage      32
#define sz_xkbAnyEvent      32
#define sz_xkbBellNotify      32
#define sz_xkbCompatMapNotify      32
#define sz_xkbControlsNotify      32
#define sz_xkbEvent      32
#define sz_xkbExtensionDeviceNotify      32
#define sz_xkbGetCompatMapReply      32
#define sz_xkbGetDeviceInfoReply      32
#define sz_xkbGetGeometryReply      32
#define sz_xkbGetIndicatorMapReply      32
#define sz_xkbGetIndicatorStateReply      32
#define sz_xkbGetKbdByNameReply      32
#define sz_xkbGetNamedIndicatorReply      32
#define sz_xkbGetNamesReply      32
#define sz_xkbGetStateReply      32
#define sz_xkbIndicatorNotify      32
#define sz_xkbListComponentsReply      32
#define sz_xkbMapNotify      32
#define sz_xkbNamesNotify      32
#define sz_xkbNewKeyboardNotify      32
#define sz_xkbPerClientFlagsReply      32
#define sz_xkbSetDebuggingFlagsReply      32
#define sz_xkbSetNamedIndicatorReq      32
#define sz_xkbStateNotify      32
#define sz_xkbUseExtensionReply      32
#define sz_xkbSetMapReq      36
#define sz_xkbBehaviorWireDesc      4
#define sz_xkbKTSetMapEntryWireDesc      4
#define sz_xkbModWireDesc      4
#define sz_xkbOutlineWireDesc      4
#define sz_xkbOverlayRowWireDesc      4
#define sz_xkbPointWireDesc      4
#define sz_xkbVModMapWireDesc      4
#define sz_xkbGetMapReply      40
#define sz_xkbActionWireDesc      8
#define sz_xkbGetControlsReq      8
#define sz_xkbGetIndicatorStateReq      8
#define sz_xkbGetStateReq      8
#define sz_xkbKTMapEntryWireDesc      8
#define sz_xkbKeyTypeWireDesc      8
#define sz_xkbKeyWireDesc      8
#define sz_xkbListComponentsReq      8
#define sz_xkbOverlayKeyWireDesc      8
#define sz_xkbOverlayWireDesc      8
#define sz_xkbRowWireDesc      8
#define sz_xkbShapeWireDesc      8
#define sz_xkbSymMapWireDesc      8
#define sz_xkbUseExtensionReq      8
#define sz_xkbGetControlsReply      92

```

```

typedef struct _xkbUseExtension {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
}

```



```

        CARD16 wantedMajor;
        CARD16 wantedMinor;
    } xkbUseExtensionReq;
typedef struct _xkbUseExtensionReply {
    BYTE type;
    BOOL supported;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 serverMajor;
    CARD16 serverMinor;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xkbUseExtensionReply;
typedef struct _xkbSelectEvents {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 affectWhich;
    CARD16 clear;
    CARD16 selectAll;
    CARD16 affectMap;
    CARD16 map;
} xkbSelectEventsReq;
typedef struct _xkbBell {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 bellClass;
    CARD16 bellID;
    INT8 percent;
    BOOL forceSound;
    BOOL eventOnly;
    CARD8 pad1;
    INT16 pitch;
    INT16 duration;
    CARD16 pad2;
    CARD32 name;
    CARD32 window;
} xkbBellReq;
typedef struct _xkbGetState {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad;
} xkbGetStateReq;
typedef struct _xkbGetStateReply {
    BYTE type;
    BYTE deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 mods;
    CARD8 baseMods;
    CARD8 latchedMods;
    CARD8 lockedMods;
    CARD8 group;
    CARD8 lockedGroup;
    INT16 baseGroup;
    INT16 latchedGroup;
    CARD8 compatState;
    CARD8 grabMods;

```

```

        CARD8 compatGrabMods;
        CARD8 lookupMods;
        CARD8 compatLookupMods;
        CARD8 pad1;
        CARD16 ptrBtnState;
        CARD16 pad2;
        CARD32 pad3;
    } xkbGetStateReply;
typedef struct _xkbLatchLockState {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD8 affectModLocks;
    CARD8 modLocks;
    BOOL lockGroup;
    CARD8 groupLock;
    CARD8 affectModLatches;
    CARD8 modLatches;
    CARD8 pad;
    BOOL latchGroup;
    INT16 groupLatch;
} xkbLatchLockStateReq;
typedef struct _xkbGetControls {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad;
} xkbGetControlsReq;
typedef struct _xkbGetControlsReply {
    BYTE type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 mkDfltBtn;
    CARD8 numGroups;
    CARD8 groupsWrap;
    CARD8 internalMods;
    CARD8 ignoreLockMods;
    CARD8 internalRealMods;
    CARD8 ignoreLockRealMods;
    CARD8 pad1;
    CARD16 internalVMods;
    CARD16 ignoreLockVMods;
    CARD16 repeatDelay;
    CARD16 repeatInterval;
    CARD16 slowKeysDelay;
    CARD16 debounceDelay;
    CARD16 mkDelay;
    CARD16 mkInterval;
    CARD16 mkTimeToMax;
    CARD16 mkMaxSpeed;
    INT16 mkCurve;
    CARD16 axOptions;
    CARD16 axTimeout;
    CARD16 axtOptsMask;
    CARD16 axtOptsValues;
    CARD16 pad2;
    CARD32 axtCtrlsMask;
    CARD32 axtCtrlsValues;
    CARD32 enabledCtrls;
    BYTE perKeyRepeat[32];
} xkbGetControlsReply;
typedef struct _xkbSetControls {
    CARD8 reqType;

```

```

CARD8  xkbReqType;
CARD16 length;
CARD16 deviceSpec;
CARD8  affectInternalMods;
CARD8  internalMods;
CARD8  affectIgnoreLockMods;
CARD8  ignoreLockMods;
CARD16 affectInternalVMods;
CARD16 internalVMods;
CARD16 affectIgnoreLockVMods;
CARD16 ignoreLockVMods;
CARD8  mkDfltBtn;
CARD8  groupsWrap;
CARD16 axOptions;
CARD16 pad1;
CARD32 affectEnabledCtrls;
CARD32 enabledCtrls;
CARD32 changeCtrls;
CARD16 repeatDelay;
CARD16 repeatInterval;
CARD16 slowKeysDelay;
CARD16 debounceDelay;
CARD16 mkDelay;
CARD16 mkInterval;
CARD16 mkTimeToMax;
CARD16 mkMaxSpeed;
INT16  mkCurve;
CARD16 axTimeout;
CARD32 axCtrlsMask;
CARD32 axCtrlsValues;
CARD16 axOptsMask;
CARD16 axOptsValues;
BYTE  perKeyRepeat[32];
} xkbSetControlsReq;
typedef struct _xkbKMapEntryWireDesc {
    BOOL active;
    CARD8 mask;
    CARD8 level;
    CARD8 realMods;
    CARD16 virtualMods;
    CARD16 pad;
} xkbKMapEntryWireDesc;
typedef struct _xkbKSetMapEntryWireDesc {
    CARD8 level;
    CARD8 realMods;
    CARD16 virtualMods;
} xkbKSetMapEntryWireDesc;
typedef struct _xkbModsWireDesc {
    CARD8 mask;
    CARD8 realMods;
    CARD16 virtualMods;
} xkbModsWireDesc;
typedef struct _xkbKeyTypeWireDesc {
    CARD8 mask;
    CARD8 realMods;
    CARD16 virtualMods;
    CARD8 numLevels;
    CARD8 nMapEntries;
    BOOL preserve;
    CARD8 pad;
} xkbKeyTypeWireDesc;
typedef struct _xkbSymMapWireDesc {
    CARD8 ktIndex[4];
    CARD8 groupInfo;
    CARD8 width;
    CARD16 nSyms;

```

```

} xkbSymMapWireDesc;
typedef struct _xkbVModMapWireDesc {
    CARD8 key;
    CARD8 pad;
    CARD16 vmods;
} xkbVModMapWireDesc;
typedef struct _xkbBehaviorWireDesc {
    CARD8 key;
    CARD8 type;
    CARD8 data;
    CARD8 pad;
} xkbBehaviorWireDesc;
typedef struct _xkbActionWireDesc {
    CARD8 type;
    CARD8 data[7];
} xkbActionWireDesc;
typedef struct _xkbGetMap {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 full;
    CARD16 partial;
    CARD8 firstType;
    CARD8 nTypes;
    CARD8 firstKeySym;
    CARD8 nKeySyms;
    CARD8 firstKeyAct;
    CARD8 nKeyActs;
    CARD8 firstKeyBehavior;
    CARD8 nKeyBehaviors;
    CARD16 virtualMods;
    CARD8 firstKeyExplicit;
    CARD8 nKeyExplicit;
    CARD8 firstModMapKey;
    CARD8 nModMapKeys;
    CARD8 firstVModMapKey;
    CARD8 nVModMapKeys;
    CARD16 pad1;
} xkbGetMapReq;
typedef struct _xkbGetMapReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 pad1;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    CARD16 present;
    CARD8 firstType;
    CARD8 nTypes;
    CARD8 totalTypes;
    CARD8 firstKeySym;
    CARD16 totalSyms;
    CARD8 nKeySyms;
    CARD8 firstKeyAct;
    CARD16 totalActs;
    CARD8 nKeyActs;
    CARD8 firstKeyBehavior;
    CARD8 nKeyBehaviors;
    CARD8 totalKeyBehaviors;
    CARD8 firstKeyExplicit;
    CARD8 nKeyExplicit;
    CARD8 totalKeyExplicit;
    CARD8 firstModMapKey;
    CARD8 nModMapKeys;

```

```

CARD8 totalModMapKeys;
CARD8 firstVModMapKey;
CARD8 nVModMapKeys;
CARD8 totalVModMapKeys;
CARD8 pad2;
CARD16 virtualMods;
} xkbGetMapReply;
typedef struct _xkbSetMap {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 present;
    CARD16 flags;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    CARD8 firstType;
    CARD8 nTypes;
    CARD8 firstKeySym;
    CARD8 nKeySyms;
    CARD16 totalSyms;
    CARD8 firstKeyAct;
    CARD8 nKeyActs;
    CARD16 totalActs;
    CARD8 firstKeyBehavior;
    CARD8 nKeyBehaviors;
    CARD8 totalKeyBehaviors;
    CARD8 firstKeyExplicit;
    CARD8 nKeyExplicit;
    CARD8 totalKeyExplicit;
    CARD8 firstModMapKey;
    CARD8 nModMapKeys;
    CARD8 totalModMapKeys;
    CARD8 firstVModMapKey;
    CARD8 nVModMapKeys;
    CARD8 totalVModMapKeys;
    CARD16 virtualMods;
} xkbSetMapReq;
typedef struct _xkbSymInterpretWireDesc {
    CARD32 sym;
    CARD8 mods;
    CARD8 match;
    CARD8 virtualMod;
    CARD8 flags;
    xkbActionWireDesc act;
} xkbSymInterpretWireDesc;
typedef struct _xkbGetCompatMap {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD8 groups;
    BOOL getAllSI;
    CARD16 firstSI;
    CARD16 nSI;
} xkbGetCompatMapReq;
typedef struct _xkbGetCompatMapReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 groups;
    CARD8 pad1;
    CARD16 firstSI;
    CARD16 nSI;
    CARD16 nTotalSI;

```

```

        CARD32 pad2;
        CARD32 pad3;
        CARD32 pad4;
        CARD32 pad5;
    } xkbGetCompatMapReply;
typedef struct _xkbSetCompatMap {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD8 pad1;
    BOOL recomputeActions;
    BOOL truncatesI;
    CARD8 groups;
    CARD16 firstSI;
    CARD16 nSI;
    CARD16 pad2;
} xkbSetCompatMapReq;
typedef struct _xkbGetIndicatorState {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad1;
} xkbGetIndicatorStateReq;
typedef struct _xkbGetIndicatorStateReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 state;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xkbGetIndicatorStateReply;
typedef struct _xkbGetIndicatorMap {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad;
    CARD32 which;
} xkbGetIndicatorMapReq;
typedef struct _xkbGetIndicatorMapReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 which;
    CARD32 realIndicators;
    CARD8 nIndicators;
    CARD8 pad1;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xkbGetIndicatorMapReply;
typedef struct _xkbIndicatorMapWireDesc {
    CARD8 flags;
    CARD8 whichGroups;
    CARD8 groups;
    CARD8 whichMods;
    CARD8 mods;
    CARD8 realMods;

```

```

        CARD16 virtualMods;
        CARD32 ctrls;
    } xkbIndicatorMapWireDesc;
typedef struct _xkbSetIndicatorMap {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad1;
    CARD32 which;
} xkbSetIndicatorMapReq;
typedef struct _xkbGetNamedIndicator {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 ledClass;
    CARD16 ledID;
    CARD16 pad1;
    CARD32 indicator;
} xkbGetNamedIndicatorReq;
typedef struct _xkbGetNamedIndicatorReply {
    BYTE type;
    BYTE deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 indicator;
    BOOL found;
    BOOL on;
    BOOL realIndicator;
    CARD8 ndx;
    CARD8 flags;
    CARD8 whichGroups;
    CARD8 groups;
    CARD8 whichMods;
    CARD8 mods;
    CARD8 realMods;
    CARD16 virtualMods;
    CARD32 ctrls;
    BOOL supported;
    CARD8 pad1;
    CARD16 pad2;
} xkbGetNamedIndicatorReply;
typedef struct _xkbSetNamedIndicator {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 ledClass;
    CARD16 ledID;
    CARD16 pad1;
    CARD32 indicator;
    BOOL setState;
    BOOL on;
    BOOL setMap;
    BOOL createMap;
    CARD8 pad2;
    CARD8 flags;
    CARD8 whichGroups;
    CARD8 groups;
    CARD8 whichMods;
    CARD8 realMods;
    CARD16 virtualMods;
    CARD32 ctrls;
} xkbSetNamedIndicatorReq;
typedef struct _xkbGetNames {

```

```

        CARD8 reqType;
        CARD8 xkbReqType;
        CARD16 length;
        CARD16 deviceSpec;
        CARD16 pad;
        CARD32 which;
    } xkbGetNamesReq;
typedef struct _xkbGetNamesReply {
    BYTE type;
    BYTE deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 which;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    CARD8 nTypes;
    CARD8 groupNames;
    CARD16 virtualMods;
    CARD8 firstKey;
    CARD8 nKeys;
    CARD32 indicators;
    CARD8 nRadioGroups;
    CARD8 nKeyAliases;
    CARD16 nKTLevels;
    CARD32 pad3;
} xkbGetNamesReply;
typedef struct _xkbSetNames {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 virtualMods;
    CARD32 which;
    CARD8 firstType;
    CARD8 nTypes;
    CARD8 firstKTLevel;
    CARD8 nKTLevels;
    CARD32 indicators;
    CARD8 groupNames;
    CARD8 nRadioGroups;
    CARD8 firstKey;
    CARD8 nKeys;
    CARD8 nKeyAliases;
    CARD8 pad1;
    CARD16 totalKTLevelNames;
} xkbSetNamesReq;
typedef struct _xkbPointWireDesc {
    INT16 x;
    INT16 y;
} xkbPointWireDesc;
typedef struct _xkbOutlineWireDesc {
    CARD8 nPoints;
    CARD8 cornerRadius;
    CARD16 pad;
} xkbOutlineWireDesc;
typedef struct _xkbShapeWireDesc {
    CARD32 name;
    CARD8 nOutlines;
    CARD8 primaryNdx;
    CARD8 approxNdx;
    CARD8 pad;
} xkbShapeWireDesc;
typedef struct _xkbSectionWireDesc {
    CARD32 name;
    INT16 top;
    INT16 left;

```



```

        CARD16 width;
        CARD16 height;
        INT16 angle;
        CARD8 priority;
        CARD8 nRows;
        CARD8 nDoodads;
        CARD8 nOverlays;
        CARD16 pad;
    } xkbSectionWireDesc;
typedef struct _xkbRowWireDesc {
    INT16 top;
    INT16 left;
    CARD8 nKeys;
    BOOL vertical;
    CARD16 pad;
} xkbRowWireDesc;
typedef struct _xkbKeyWireDesc {
    CARD8 name[4];
    INT16 gap;
    CARD8 shapeNdx;
    CARD8 colorNdx;
} xkbKeyWireDesc;
typedef struct _xkbOverlayWireDesc {
    CARD32 name;
    CARD8 nRows;
    CARD8 pad1;
    CARD16 pad2;
} xkbOverlayWireDesc;
typedef struct _xkbOverlayRowWireDesc {
    CARD8 rowUnder;
    CARD8 nKeys;
    CARD16 pad1;
} xkbOverlayRowWireDesc;
typedef struct _xkbOverlayKeyWireDesc {
    CARD8 over[4];
    CARD8 under[4];
} xkbOverlayKeyWireDesc;
typedef struct _xkbShapeDoodadWireDesc {
    CARD32 name;
    CARD8 type;
    CARD8 priority;
    INT16 top;
    INT16 left;
    INT16 angle;
    CARD8 colorNdx;
    CARD8 shapeNdx;
    CARD16 pad1;
    CARD32 pad2;
} xkbShapeDoodadWireDesc;
typedef struct _xkbTextDoodadWireDesc {
    CARD32 name;
    CARD8 type;
    CARD8 priority;
    INT16 top;
    INT16 left;
    INT16 angle;
    CARD16 width;
    CARD16 height;
    CARD8 colorNdx;
    CARD8 pad1;
    CARD16 pad2;
} xkbTextDoodadWireDesc;
typedef struct _xkbIndicatorDoodadWireDesc {
    CARD32 name;
    CARD8 type;
    CARD8 priority;

```

```

        INT16 top;
        INT16 left;
        INT16 angle;
        CARD8 shapeNdx;
        CARD8 onColorNdx;
        CARD8 offColorNdx;
        CARD8 pad1;
        CARD32 pad2;
    } xkbIndicatorDoodadWireDesc;
typedef struct _xkbLogoDoodadWireDesc {
    CARD32 name;
    CARD8 type;
    CARD8 priority;
    INT16 top;
    INT16 left;
    INT16 angle;
    CARD8 colorNdx;
    CARD8 shapeNdx;
    CARD16 pad1;
    CARD32 pad2;
} xkbLogoDoodadWireDesc;
typedef struct _xkbAnyDoodadWireDesc {
    CARD32 name;
    CARD8 type;
    CARD8 priority;
    INT16 top;
    INT16 left;
    INT16 angle;
    CARD32 pad2;
    CARD32 pad3;
} xkbAnyDoodadWireDesc;
union _xkbDoodadWireDesc {
    xkbAnyDoodadWireDesc any;
    xkbShapeDoodadWireDesc shape;
    xkbTextDoodadWireDesc text;
    xkbIndicatorDoodadWireDesc indicator;
    xkbLogoDoodadWireDesc logo;
};
typedef union _xkbDoodadWireDesc {
    xkbAnyDoodadWireDesc any;
    xkbShapeDoodadWireDesc shape;
    xkbTextDoodadWireDesc text;
    xkbIndicatorDoodadWireDesc indicator;
    xkbLogoDoodadWireDesc logo;
} xkbDoodadWireDesc;
typedef struct _xkbGetGeometry {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad;
    CARD32 name;
} xkbGetGeometryReq;
typedef struct _xkbGetGeometryReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 name;
    BOOL found;
    CARD8 pad;
    CARD16 widthMM;
    CARD16 heightMM;
    CARD16 nProperties;
    CARD16 nColors;
    CARD16 nShapes;

```

```

CARD16 nSections;
CARD16 nDoodads;
CARD16 nKeyAliases;
CARD8 baseColorNdx;
CARD8 labelColorNdx;
} xkbGetGeometryReply;
typedef struct _xkbSetGeometry {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD8 nShapes;
    CARD8 nSections;
    CARD32 name;
    CARD16 widthMM;
    CARD16 heightMM;
    CARD16 nProperties;
    CARD16 nColors;
    CARD16 nDoodads;
    CARD16 nKeyAliases;
    CARD8 baseColorNdx;
    CARD8 labelColorNdx;
    CARD16 pad;
} xkbSetGeometryReq;
typedef struct _xkbPerClientFlags {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 pad1;
    CARD32 change;
    CARD32 value;
    CARD32 ctrlsToChange;
    CARD32 autoCtrls;
    CARD32 autoCtrlValues;
} xkbPerClientFlagsReq;
typedef struct _xkbPerClientFlagsReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 supported;
    CARD32 value;
    CARD32 autoCtrls;
    CARD32 autoCtrlValues;
    CARD32 pad1;
    CARD32 pad2;
} xkbPerClientFlagsReply;
typedef struct _xkbListComponents {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 maxNames;
} xkbListComponentsReq;
typedef struct _xkbListComponentsReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 nKeymaps;
    CARD16 nKeycodes;
    CARD16 nTypes;
    CARD16 nCompatMaps;
    CARD16 nSymbols;
    CARD16 nGeometries;

```

```

        CARD16 extra;
        CARD16 pad1;
        CARD32 pad2;
        CARD32 pad3;
    } xkbListComponentsReply;
typedef struct _xkbGetKbdByName {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 need;
    CARD16 want;
    BOOL load;
    CARD8 pad;
} xkbGetKbdByNameReq;
typedef struct _xkbGetKbdByNameReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    BOOL loaded;
    BOOL newKeyboard;
    CARD16 found;
    CARD16 reported;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xkbGetKbdByNameReply;
typedef struct _xkbDeviceLedsWireDesc {
    CARD16 ledClass;
    CARD16 ledID;
    CARD32 namesPresent;
    CARD32 mapsPresent;
    CARD32 physIndicators;
    CARD32 state;
} xkbDeviceLedsWireDesc;
typedef struct _xkbGetDeviceInfo {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD16 wanted;
    BOOL allBtns;
    CARD8 firstBtn;
    CARD8 nBtns;
    CARD8 pad;
    CARD16 ledClass;
    CARD16 ledID;
} xkbGetDeviceInfoReq;
typedef struct _xkbGetDeviceInfoReply {
    CARD8 type;
    CARD8 deviceID;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 present;
    CARD16 supported;
    CARD16 unsupported;
    CARD16 nDeviceLedFBs;
    CARD8 firstBtnWanted;
    CARD8 nBtnsWanted;
    CARD8 firstBtnRtrn;
    CARD8 nBtnsRtrn;
    CARD8 totalBtns;
}

```

```

        BOOL hasOwnState;
        CARD16 dfltKbdFB;
        CARD16 dfltLedFB;
        CARD16 pad;
        CARD32 devType;
    } xkbGetDeviceInfoReply;
typedef struct _xkbSetDeviceInfo {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 deviceSpec;
    CARD8 firstBtn;
    CARD8 nBtns;
    CARD16 change;
    CARD16 nDeviceLedFBs;
} xkbSetDeviceInfoReq;
typedef struct _xkbSetDebuggingFlags {
    CARD8 reqType;
    CARD8 xkbReqType;
    CARD16 length;
    CARD16 msgLength;
    CARD16 pad;
    CARD32 affectFlags;
    CARD32 flags;
    CARD32 affectCtrls;
    CARD32 ctrls;
} xkbSetDebuggingFlagsReq;
typedef struct _xkbSetDebuggingFlagsReply {
    BYTE type;
    CARD8 pad0;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 currentFlags;
    CARD32 currentCtrls;
    CARD32 supportedFlags;
    CARD32 supportedCtrls;
    CARD32 pad1;
    CARD32 pad2;
} xkbSetDebuggingFlagsReply;
typedef struct _xkbAnyEvent {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 pad1;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xkbAnyEvent;
typedef struct _xkbNewKeyboardNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 oldDeviceID;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    CARD8 oldMinKeyCode;
    CARD8 oldMaxKeyCode;
    CARD8 requestMajor;
    CARD8 requestMinor;

```

```

        CARD16 changed;
        CARD8 detail;
        CARD8 pad1;
        CARD32 pad2;
        CARD32 pad3;
        CARD32 pad4;
    } xkbNewKeyboardNotify;
typedef struct _xkbMapNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 ptrBtnActions;
    CARD16 changed;
    CARD8 minKeyCode;
    CARD8 maxKeyCode;
    CARD8 firstType;
    CARD8 nTypes;
    CARD8 firstKeySym;
    CARD8 nKeySyms;
    CARD8 firstKeyAct;
    CARD8 nKeyActs;
    CARD8 firstKeyBehavior;
    CARD8 nKeyBehaviors;
    CARD8 firstKeyExplicit;
    CARD8 nKeyExplicit;
    CARD8 firstModMapKey;
    CARD8 nModMapKeys;
    CARD8 firstVModMapKey;
    CARD8 nVModMapKeys;
    CARD16 virtualMods;
    CARD16 pad1;
} xkbMapNotify;
typedef struct _xkbStateNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 mods;
    CARD8 baseMods;
    CARD8 latchedMods;
    CARD8 lockedMods;
    CARD8 group;
    INT16 baseGroup;
    INT16 latchedGroup;
    CARD8 lockedGroup;
    CARD8 compatState;
    CARD8 grabMods;
    CARD8 compatGrabMods;
    CARD8 lookupMods;
    CARD8 compatLookupMods;
    CARD16 ptrBtnState;
    CARD16 changed;
    CARD8 keycode;
    CARD8 eventType;
    CARD8 requestMajor;
    CARD8 requestMinor;
} xkbStateNotify;
typedef struct _xkbControlsNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;

```

```

CARD8 numGroups;
CARD16 pad1;
CARD32 changedControls;
CARD32 enabledControls;
CARD32 enabledControlChanges;
CARD8 keycode;
CARD8 eventType;
CARD8 requestMajor;
CARD8 requestMinor;
CARD32 pad2;
} xkbControlsNotify;
typedef struct _xkbIndicatorNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 pad1;
    CARD16 pad2;
    CARD32 state;
    CARD32 changed;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xkbIndicatorNotify;
typedef struct _xkbNamesNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 pad1;
    CARD16 changed;
    CARD8 firstType;
    CARD8 nTypes;
    CARD8 firstLevelName;
    CARD8 nLevelNames;
    CARD8 pad2;
    CARD8 nRadioGroups;
    CARD8 nAliases;
    CARD8 changedGroupNames;
    CARD16 changedVirtualMods;
    CARD8 firstKey;
    CARD8 nKeys;
    CARD32 changedIndicators;
    CARD32 pad3;
} xkbNamesNotify;
typedef struct _xkbCompatMapNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 changedGroups;
    CARD16 firstSI;
    CARD16 nSI;
    CARD16 nTotalSI;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xkbCompatMapNotify;
typedef struct _xkbBellNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;

```

```

        CARD32 time;
        CARD8 deviceID;
        CARD8 bellClass;
        CARD8 bellID;
        CARD8 percent;
        CARD16 pitch;
        CARD16 duration;
        CARD32 name;
        CARD32 window;
        BOOL eventOnly;
        CARD8 pad1;
        CARD16 pad2;
        CARD32 pad3;
    } xkbBellNotify;
typedef struct _xkbActionMessage {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 keycode;
    BOOL press;
    BOOL keyEventFollows;
    CARD8 mods;
    CARD8 group;
    CARD8 message[8];
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
} xkbActionMessage;
typedef struct _xkbAccessXNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 keycode;
    CARD16 detail;
    CARD16 slowKeysDelay;
    CARD16 debounceDelay;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xkbAccessXNotify;
typedef struct _xkbExtensionDeviceNotify {
    BYTE type;
    BYTE xkbType;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 deviceID;
    CARD8 pad1;
    CARD16 reason;
    CARD16 ledClass;
    CARD16 ledID;
    CARD32 ledsDefined;
    CARD32 ledState;
    CARD8 firstBtn;
    CARD8 nBtns;
    CARD16 supported;
    CARD16 unsupported;
    CARD16 pad3;
} xkbExtensionDeviceNotify;
typedef struct _xkbEvent {
    union {
        xkbAnyEvent any;
    }

```



```

xkbNewKeyboardNotify new_kbd;
xkbMapNotify map;
xkbStateNotify state;
xkbControlsNotify ctrls;
xkbIndicatorNotify indicators;
xkbNamesNotify names;
xkbCompatMapNotify compat;
xkbBellNotify bell;
xkbActionMessage message;
xkbAccessXNotify accessx;
xkbExtensionDeviceNotify device;
} u;
} xkbEvent;

```

6.2.22 X11/extensions/XKBstr.h

```

#define _XKBSTR_H_
#define XkbSMKeyActionsPtr(m,k) (&(m)->acts[(m)->key_acts[k]])
#define XkbCMKeySymsPtr(m,k) (&(m)->syms[XkbCMKeySymsOffset(m,k)])
#define XkbCMKeyType(m,k,g) (&(m)->types[XkbCMKeyTypeIndex(m,k,g)])
#define XkbIM_IsAuto(i) (((i)->flags&XkbIM_NoAutomatic)==0)&&
((i)->which_groups&&(i)->groups)|| ((i)->which_mods&&(i)->
>mods.mask)|| ((i)->ctrls))
#define XkbActionCtrls(a) (((unsigned int)(a)->ctrls3)<<24)|
(((unsigned int)(a)->ctrls2)<<16)| ((unsigned int)(a)->
>ctrls1)<<8)| ((unsigned int)(a)->ctrls0))
#define XkbSARedirectVMods(a) (((unsigned int)(a)->vmods1)<<8)|
((unsigned int)(a)->vmods0))
#define XkbSARedirectVModsMask(a) (((unsigned int)(a)->
>vmods_mask1)<<8)| ((unsigned int)(a)->vmods_mask0))
#define XkbActionSetCtrls(a,c) (((a)->ctrls3=(((c)>>24)&0xff)),
((a)->ctrls2=(((c)>>16)&0xff)), ((a)->ctrls1=(((c)>>8)&0xff)),
((a)->ctrls0=((c)&0xff)))
#define XkbSetModActionVMods(a,v) (((a)->
>vmods1=(((v)>>8)&0xff)), (a)->vmods2=((v)&0xff))
#define XkbSARedirectSetVMods(a,m) (((a)->
>vmods_mask1=(((m)>>8)&0xff)), (a)->vmods_mask0=((m)&0xff))
#define XkbSARedirectSetVModsMask(a,m) (((a)->
>vmods_mask1=(((m)>>8)&0xff)), (a)->vmods_mask0=((m)&0xff))
#define XkbXI_DevHasBtnActs(d) (((d)->num_btns>0)&&((d)->
>btn_acts!=NULL))
#define XkbXI_DevHasLeds(d) (((d)->num_leds>0)&&((d)->
>leds!=NULL))
#define XkbOutOfRangeGroupNumber(g) (((g)&0x30)>>4)
#define XkbSetNumGroups(g,n) (((g)&0xf0)|((n)&0x0f))
#define XkbIntTo2Chars(i,h,l) XkbIntTo2Chars(i,h,l)
((h)=((i)>>8)&0xff), ((l)=((i)&0xff))
#define XkbIM_InUse(i) (((i)->flags)||((i)->which_groups)|| ((i)->
>which_mods)||((i)->ctrls))
#define XkbKeycodeInRange(d,k) (((k)>=(d)->min_key_code)&&
((k)<=(d)->max_key_code))
#define XkbSetGroupInfo(g,w,n)
(((w)&0xc0)|((n)&3)<<4)|((g)&0x0f))
#define XkbSASetGroup(a,g) ((a)->group_XXX=(g))
#define XkbSASetScreen(a,s) ((a)->screenXXX=((s)&0xff))
#define XkbSASetPtrDfltValue(a,c) ((a)->valueXXX=((c)&0xff))
#define XkbAX_NeedOption(c,w) ((c)->ax_options&(w))
#define XkbAX_AnyFeedback(c) ((c)->
>enabled_ctrls&XkbAccessXFeedbackMask)
#define XkbNumKeys(d) ((d)->max_key_code-(d)->min_key_code+1)
#define XkbKeyHasActions(d,k) ((d)->server->key_acts[k]!=0)
#define XkbNumGroups(g) ((g)&0x0f)
#define XkbOutOfRangeGroupAction(g) ((g)&0xc0)

```

```

#define XkbOutOfRangeGroupInfo(g) ((g) & 0xf0)
#define XkbCMKeyGroupInfo(m,k) ((m) -> key_sym_map[k].group_info)
#define XkbCMKeyTypeIndex(m,k,g) ((m) -> key_sym_map[k].kt_index[g & 0x3])
#define XkbCMKeySymsOffset(m,k) ((m) -> key_sym_map[k].offset)
#define XkbCMKeyGroupsWidth(m,k) ((m) -> key_sym_map[k].width)
#define XkbStateGroup(s) ((s) -> base_group + (s) -> latched_group + XkbGroupLock(s))
#define XkbStateMods(s) ((s) -> base_mods | (s) -> latched_mods | XkbModLocks(s))
#define XkbGroupLock(s) ((s) -> locked_group)
#define XkbModLocks(s) ((s) -> locked_mods)
#define XkbModActionVMods(a) ((short) (((a) -> vmods1 < 8) | ((a) -> vmods2)))
#define Xkb2CharsToInt(h,l) ((short) (((h) < 8) | (l)))
#define XkbCharToInt(v) ((v) & 0x80 ? (int) ((v) | (~0xff)) : (int) ((v) & 0x7f))
#define XkbPtrActionX(a) (Xkb2CharsToInt((a) -> high_XXX, (a) -> low_XXX))
#define XkbPtrActionY(a) (Xkb2CharsToInt((a) -> high_YYY, (a) -> low_YYY))
#define XkbAX_NeedFeedback(c,w) (XkbAX_AnyFeedback(c) && XkbAX_NeedOption(c,w))
#define XkbSAGroup(a) (XkbCharToInt((a) -> group_XXX))
#define XkbSAScreen(a) (XkbCharToInt((a) -> screen_XXX))
#define XkbSAPtrDfltValue(a) (XkbCharToInt((a) -> value_XXX))
#define XkbKeyGroupInfo(d,k) (XkbCMKeyGroupInfo((d) -> map, (k)))
#define XkbKeyGroupsWidth(d,k) (XkbCMKeyGroupsWidth((d) -> map, k))
#define XkbCMKeyNumSyms(m,k) (XkbCMKeyGroupsWidth(m,k) * XkbCMKeyNumGroups(m,k))
#define XkbKeyGroupWidth(d,k,g) (XkbCMKeyGroupWidth((d) -> map, k, g))
#define XkbKeyNumGroups(d,k) (XkbCMKeyNumGroups((d) -> map, (k)))
#define XkbKeyNumSyms(d,k) (XkbCMKeyNumSyms((d) -> map, (k)))
#define XkbKeySymsPtr(d,k) (XkbCMKeySymsPtr((d) -> map, (k)))
#define XkbKeyKeyType(d,k,g) (XkbCMKeyType((d) -> map, k, g))
#define XkbCMKeyGroupWidth(m,k,g) (XkbCMKeyType(m,k,g) -> num_levels)
#define XkbKeyKeyTypeIndex(d,k,g) (XkbCMKeyTypeIndex((d) -> map, k, g))
#define XkbSetPtrActionX(a,x) (XkbIntTo2Chars(x, (a) -> high_XXX, (a) -> low_XXX))
#define XkbSetPtrActionY(a,y) (XkbIntTo2Chars(y, (a) -> high_YYY, (a) -> low_YYY))
#define XkbKeyActionEntry(d,k,sl,g) (XkbKeyHasActions(d,k) ? XkbKeyAction(d,k, ((XkbKeyGroupsWidth(d,k) * (g)) + (sl))) : NULL)
#define XkbKeyHasActions(d,k) ?&XkbKeyActionsPtr(d,k)[n] : NULL
#define XkbKeyNumActions(d,k) (XkbKeyHasActions(d,k) ? XkbKeyNumSyms(d,k) : 1)
#define XkbKeySymEntry(d,k,sl,g) (XkbKeySym(d,k, ((XkbKeyGroupsWidth(d,k) * (g)) + (sl))))
#define XkbKeySym(d,k,n) (XkbKeySymsPtr(d,k)[n])
#define XkbCMKeyNumGroups(m,k) (XkbNumGroups((m) -> key_sym_map[k].group_info))
#define XkbKeyActionsPtr(d,k) (XkbSMKeyActionsPtr((d) -> server, k))
#define XkbXI_LegalDevBtn(d,b) (XkbXI_DevHasBtnActs(d) && ((b) < (d) -> num_btns))
#define XkbAnyActionDataSize 7
#define XkbGrabStateFromRec(s) XkbBuildCoreState((s) -> grab_mods, (s) -> group)
#define XkbStateFieldFromRec(s) XkbBuildCoreState((s) -> lookup_mods, (s) -> group)

typedef struct _XkbStateRec {
    unsigned char group;
    unsigned char locked_group;

```

```

    unsigned short base_group;
    unsigned short latched_group;
    unsigned char mods;
    unsigned char base_mods;
    unsigned char latched_mods;
    unsigned char locked_mods;
    unsigned char compat_state;
    unsigned char grab_mods;
    unsigned char compat_grab_mods;
    unsigned char lookup_mods;
    unsigned char compat_lookup_mods;
    unsigned short ptr_buttons;
} XkbStateRec;
typedef struct _XkbStateRec *XkbStatePtr;
typedef struct _XkbMods *XkbModsPtr;
typedef struct _XkbMods {
    unsigned char mask;
    unsigned char real_mods;
    unsigned short vmods;
} XkbModsRec;
typedef struct _XkbKMapEntry {
    int active;
    unsigned char level;
    XkbModsRec mods;
} XkbKMapEntryRec;
typedef struct _XkbKMapEntry *XkbKMapEntryPtr;
typedef struct _XkbKeyType {
    XkbModsRec mods;
    unsigned char num_levels;
    unsigned char map_count;
    XkbKMapEntryPtr map;
    XkbModsPtr preserve;
    Atom name;
    Atom *level_names;
} XkbKeyTypeRec;
typedef struct _XkbKeyType *XkbKeyTypePtr;
typedef struct _XkbBehavior {
    unsigned char type;
    unsigned char data;
} XkbBehavior;
typedef struct _XkbAnyAction {
    unsigned char type;
    unsigned char data[8];
} XkbAnyAction;
typedef struct _XkbModAction {
    unsigned char type;
    unsigned char flags;
    unsigned char mask;
    unsigned char real_mods;
    unsigned char vmods1;
    unsigned char vmods2;
} XkbModAction;
typedef struct _XkbGroupAction {
    unsigned char type;
    unsigned char flags;
    char group_XXX;
} XkbGroupAction;
typedef struct _XkbISOAction {
    unsigned char type;
    unsigned char flags;
    unsigned char mask;
    unsigned char real_mods;
    char group_XXX;
    unsigned char affect;
    unsigned char vmods1;
    unsigned char vmods2;
}

```

```

} XkbISOAction;
typedef struct _XkbPtrAction {
    unsigned char type;
    unsigned char flags;
    unsigned char high_XXX;
    unsigned char low_XXX;
    unsigned char high_YYY;
    unsigned char low_YYY;
} XkbPtrAction;
typedef struct _XkbPtrBtnAction {
    unsigned char type;
    unsigned char flags;
    unsigned char count;
    unsigned char button;
} XkbPtrBtnAction;
typedef struct _XkbPtrDfltAction {
    unsigned char type;
    unsigned char flags;
    unsigned char affect;
    char valueXXX;
} XkbPtrDfltAction;
typedef struct _XkbSwitchScreenAction {
    unsigned char type;
    unsigned char flags;
    char screenXXX;
} XkbSwitchScreenAction;
typedef struct _XkbCtrlsAction {
    unsigned char type;
    unsigned char flags;
    unsigned char ctrls3;
    unsigned char ctrls2;
    unsigned char ctrls1;
    unsigned char ctrls0;
} XkbCtrlsAction;
typedef struct _XkbMessageAction {
    unsigned char type;
    unsigned char flags;
    unsigned char message[6];
} XkbMessageAction;
typedef struct _XkbRedirectKeyAction {
    unsigned char type;
    unsigned char new_key;
    unsigned char mods_mask;
    unsigned char mods;
    unsigned char vmods_mask0;
    unsigned char vmods_mask1;
    unsigned char vmods0;
    unsigned char vmods1;
} XkbRedirectKeyAction;
typedef struct _XkbDeviceBtnAction {
    unsigned char type;
    unsigned char flags;
    unsigned char count;
    unsigned char button;
    unsigned char device;
} XkbDeviceBtnAction;
typedef struct _XkbDeviceValuatorAction {
    unsigned char type;
    unsigned char device;
    unsigned char v1_what;
    unsigned char v1_ndx;
    unsigned char v1_value;
    unsigned char v2_what;
    unsigned char v2_ndx;
    unsigned char v2_value;
} XkbDeviceValuatorAction;

```

```

typedef union _XkbAction {
    XkbAnyAction any;
    XkbModAction mods;
    XkbGroupAction group;
    XkbISOAction iso;
    XkbPtrAction ptr;
    XkbPtrBtnAction btn;
    XkbPtrDfltAction dflt;
    XkbSwitchScreenAction screen;
    XkbCtrlsAction ctrls;
    XkbMessageAction msg;
    XkbRedirectKeyAction redirect;
    XkbDeviceBtnAction devbtn;
    XkbDeviceValuatorAction devval;
    unsigned char type;
} XkbAction;

typedef struct _XkbControls {
    unsigned char mk_dflt_btn;
    unsigned char num_groups;
    unsigned char groups_wrap;
    XkbModsRec internal;
    XkbModsRec ignore_lock;
    unsigned int enabled_ctrls;
    unsigned short repeat_delay;
    unsigned short repeat_interval;
    unsigned short slow_keys_delay;
    unsigned short debounce_delay;
    unsigned short mk_delay;
    unsigned short mk_interval;
    unsigned short mk_time_to_max;
    unsigned short mk_max_speed;
    short mk_curve;
    unsigned short ax_options;
    unsigned short ax_timeout;
    unsigned short axt_opts_mask;
    unsigned short axt_opts_values;
    unsigned int axt_ctrls_mask;
    unsigned int axt_ctrls_values;
    unsigned char per_key_repeat[32];
} XkbControlsRec;

typedef struct _XkbControls *XkbControlsPtr;

typedef struct _XkbServerMapRec {
    unsigned short num_acts;
    unsigned short size_acts;
    XkbAction *acts;
    XkbBehavior *behaviors;
    unsigned short *key_acts;
#ifdef __cplusplus || defined(c_plusplus)
    unsigned char *c_explicit;
#else
    unsigned char *explicit;
#endif
    unsigned char vmods[16];
    unsigned short *vmodmap;
} XkbServerMapRec;

typedef struct _XkbServerMapRec *XkbServerMapPtr;

typedef struct _XkbSymMapRec {
    unsigned char kt_index[4];
    unsigned char group_info;
    unsigned char width;
    unsigned short offset;
} XkbSymMapRec;

typedef struct _XkbSymMapRec *XkbSymMapPtr;

typedef struct _XkbClientMapRec {
    unsigned char size_types;
    unsigned char num_types;

```

```

        XkbKeyTypePtr types;
        unsigned short size_syms;
        unsigned short num_syms;
        KeySym *syms;
        XkbSymMapPtr key_sym_map;
        unsigned char *modmap;
    } XkbClientMapRec;
typedef struct _XkbClientMapRec *XkbClientMapPtr;
typedef struct _XkbSymInterpretRec {
    KeySym sym;
    unsigned char flags;
    unsigned char match;
    unsigned char mods;
    unsigned char virtual_mod;
    XkbAnyAction act;
} XkbSymInterpretRec;
typedef struct _XkbSymInterpretRec *XkbSymInterpretPtr;
typedef struct _XkbCompatMapRec {
    XkbSymInterpretPtr sym_interpret;
    XkbModsRec groups[4];
    unsigned short num_si;
    unsigned short size_si;
} XkbCompatMapRec;
typedef struct _XkbCompatMapRec *XkbCompatMapPtr;
typedef struct _XkbIndicatorMapRec *XkbIndicatorMapPtr;
typedef struct _XkbIndicatorMapRec {
    unsigned char flags;
    unsigned char which_groups;
    unsigned char groups;
    unsigned char which_mods;
    XkbModsRec mods;
    unsigned int ctrls;
} XkbIndicatorMapRec;
typedef struct _XkbIndicatorRec {
    unsigned long int phys_indicators;
    XkbIndicatorMapRec maps[32];
} XkbIndicatorRec;
typedef struct _XkbIndicatorRec *XkbIndicatorPtr;
typedef struct _XkbKeyNameRec {
    char name[4];
} XkbKeyNameRec;
typedef struct _XkbKeyNameRec *XkbKeyNamePtr;
typedef struct _XkbKeyAliasRec {
    char real[4];
    char alias[4];
} XkbKeyAliasRec;
typedef struct _XkbKeyAliasRec *XkbKeyAliasPtr;
typedef struct _XkbNamesRec {
    Atom keycodes;
    Atom geometry;
    Atom symbols;
    Atom types;
    Atom compat;
    Atom vmods[16];
    Atom indicators[32];
    Atom groups[4];
    XkbKeyNamePtr keys;
    XkbKeyAliasPtr key_aliases;
    Atom *radio_groups;
    Atom phys_symbols;
    unsigned char num_keys;
    unsigned char num_key_aliases;
    unsigned short num_rg;
} XkbNamesRec;
typedef struct _XkbNamesRec *XkbNamesPtr;
typedef struct _XkbGeometry *XkbGeometryPtr;

```

```

typedef struct _XkbDesc {
    struct _XDisplay *dpy;
    unsigned short flags;
    unsigned short device_spec;
    KeyCode min_key_code;
    KeyCode max_key_code;
    XkbControlsPtr ctrls;
    XkbServerMapPtr server;
    XkbClientMapPtr map;
    XkbIndicatorPtr indicators;
    XkbNamesPtr names;
    XkbCompatMapPtr compat;
    XkbGeometryPtr geom;
} XkbDescRec;
typedef struct _XkbDesc *XkbDescPtr;
typedef struct _XkbMapChanges *XkbMapChangesPtr;
typedef struct _XkbMapChanges {
    unsigned short changed;
    KeyCode min_key_code;
    KeyCode max_key_code;
    unsigned char first_type;
    unsigned char num_types;
    KeyCode first_key_sym;
    unsigned char num_key_syms;
    KeyCode first_key_act;
    unsigned char num_key_acts;
    KeyCode first_key_behavior;
    unsigned char num_key_behaviors;
    KeyCode first_key_explicit;
    unsigned char num_key_explicit;
    KeyCode first_modmap_key;
    unsigned char num_modmap_keys;
    KeyCode first_vmodmap_key;
    unsigned char num_vmodmap_keys;
    unsigned char pad;
    unsigned short vmods;
} XkbMapChangesRec;
typedef struct _XkbControlsChanges *XkbControlsChangesPtr;
typedef struct _XkbControlsChanges {
    unsigned int changed_ctrls;
    unsigned int enabled_ctrls_changes;
    int num_groups_changed;
} XkbControlsChangesRec;
typedef struct _XkbIndicatorChanges *XkbIndicatorChangesPtr;
typedef struct _XkbIndicatorChanges {
    unsigned int state_changes;
    unsigned int map_changes;
} XkbIndicatorChangesRec;
typedef struct _XkbNameChanges {
    unsigned int changed;
    unsigned char first_type;
    unsigned char num_types;
    unsigned char first_lvl;
    unsigned char num_lvls;
    unsigned char num_aliases;
    unsigned char num_rg;
    unsigned char first_key;
    unsigned char num_keys;
    unsigned short changed_vmods;
    unsigned long int changed_indicators;
    unsigned char changed_groups;
} XkbNameChangesRec;
typedef struct _XkbNameChanges *XkbNameChangesPtr;
typedef struct _XkbCompatChanges {
    unsigned char changed_groups;
    unsigned short first_si;

```

```

        unsigned short num_si;
    } XkbCompatChangesRec;
typedef struct _XkbCompatChanges *XkbCompatChangesPtr;
typedef struct _XkbChanges {
    unsigned short device_spec;
    unsigned short state_changes;
    XkbMapChangesRec map;
    XkbControlsChangesRec ctrls;
    XkbIndicatorChangesRec indicators;
    XkbNameChangesRec names;
    XkbCompatChangesRec compat;
} XkbChangesRec;
typedef struct _XkbChanges *XkbChangesPtr;
typedef struct _XkbComponentNames {
    char *keymap;
    char *keycodes;
    char *types;
    char *compat;
    char *symbols;
    char *geometry;
} XkbComponentNamesRec;
typedef struct _XkbComponentNames *XkbComponentNamesPtr;
typedef struct _XkbComponentName {
    unsigned short flags;
    char *name;
} XkbComponentNameRec;
typedef struct _XkbComponentName *XkbComponentNamePtr;
typedef struct _XkbComponentList {
    int num_keymaps;
    int num_keycodes;
    int num_types;
    int num_compat;
    int num_symbols;
    int num_geometry;
    XkbComponentNamePtr keymaps;
    XkbComponentNamePtr keycodes;
    XkbComponentNamePtr types;
    XkbComponentNamePtr compat;
    XkbComponentNamePtr symbols;
    XkbComponentNamePtr geometry;
} XkbComponentListRec;
typedef struct _XkbComponentList *XkbComponentListPtr;
typedef struct _XkbDeviceLedInfo {
    unsigned short led_class;
    unsigned short led_id;
    unsigned int phys_indicators;
    unsigned int maps_present;
    unsigned int names_present;
    unsigned int state;
    Atom names[32];
    XkbIndicatorMapRec maps[32];
} XkbDeviceLedInfoRec;
typedef struct _XkbDeviceLedInfo *XkbDeviceLedInfoPtr;
typedef struct _XkbDeviceInfo {
    char *name;
    Atom type;
    unsigned short device_spec;
    int has_own_state;
    unsigned short supported;
    unsigned short unsupported;
    unsigned short num_btns;
    XkbAction *btn_acts;
    unsigned short sz_leds;
    unsigned short num_leds;
    unsigned short dflt_kbd_fb;
    unsigned short dflt_led_fb;

```



```

    XkbDeviceLedInfoPtr leds;
} XkbDeviceInfoRec;
typedef struct _XkbDeviceInfo *XkbDeviceInfoPtr;
typedef struct _XkbDeviceLedChanges {
    unsigned short led_class;
    unsigned short led_id;
    unsigned int defined;
    struct _XkbDeviceLedChanges *next;
} XkbDeviceLedChangesRec;
typedef struct _XkbDeviceLedChanges *XkbDeviceLedChangesPtr;
typedef struct _XkbDeviceChanges {
    unsigned int changed;
    unsigned short first_btn;
    unsigned short num_btns;
    XkbDeviceLedChangesRec leds;
} XkbDeviceChangesRec;
typedef struct _XkbDeviceChanges *XkbDeviceChangesPtr;

```

6.2.23 X11/extensions/securstr.h

```

#define SECURITY_MINOR_VERSION 0
#define X_SecurityQueryVersion 0
#define SECURITY_MAJOR_VERSION 1
#define X_SecurityGenerateAuthorization 1
#define _SECURSTR_H 1
#define sz_xSecurityGenerateAuthorizationReq 12
#define X_SecurityRevokeAuthorization 2
#define sz_xSecurityAuthorizationRevokedEvent 32
#define sz_xSecurityGenerateAuthorizationReply 32
#define sz_xSecurityQueryVersionReply 32
#define sz_xSecurityQueryVersionReq 8
#define sz_xSecurityRevokeAuthorizationReq 8
#define SECURITY_EXTENSION_NAME "SECURITY"

typedef struct {
    CARD8 reqType;
    CARD8 securityReqType;
    CARD16 length;
    CARD16 majorVersion;
    CARD16 minorVersion;
} xSecurityQueryVersionReq;
typedef struct {
    CARD8 type;
    CARD8 pad0;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 majorVersion;
    CARD16 minorVersion;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xSecurityQueryVersionReply;
typedef struct {
    CARD8 reqType;
    CARD8 securityReqType;
    CARD16 length;
    CARD16 nbytesAuthProto;
    CARD16 nbytesAuthData;
    CARD32 valueMask;
} xSecurityGenerateAuthorizationReq;
typedef struct {
    CARD8 type;
    CARD8 pad0;

```

```

        CARD16  sequenceNumber;
        CARD32  length;
        CARD32  authId;
        CARD16  dataLength;
        CARD16  pad1;
        CARD32  pad2;
        CARD32  pad3;
        CARD32  pad4;
        CARD32  pad5;
    } xSecurityGenerateAuthorizationReply;
typedef struct {
    CARD8  reqType;
    CARD8  securityReqType;
    CARD16  length;
    CARD32  authId;
} xSecurityRevokeAuthorizationReq;
typedef struct _xSecurityAuthorizationRevokedEvent {
    BYTE  type;
    BYTE  detail;
    CARD16  sequenceNumber;
    CARD32  authId;
    CARD32  pad0;
    CARD32  pad1;
    CARD32  pad2;
    CARD32  pad3;
    CARD32  pad4;
    CARD32  pad5;
} xSecurityAuthorizationRevokedEvent;

```

6.2.24 X11/keysym.h

```

#define XK_APL
#define XK_ARABIC
#define XK_ARMENIAN
#define XK_BRAILLE
#define XK_CAUCASUS
#define XK_CURRENCY
#define XK_CYRILLIC
#define XK_GEORGIAN
#define XK_GREEK
#define XK_HEBREW
#define XK_KATAKANA
#define XK_KOREAN
#define XK_LATIN1
#define XK_LATIN2
#define XK_LATIN3
#define XK_LATIN4
#define XK_LATIN8
#define XK_LATIN9
#define XK_MATHEMATICAL
#define XK_MISCELLANY
#define XK_PUBLISHING
#define XK_SPECIAL
#define XK_TECHNICAL
#define XK_THAI
#define XK_VIETNAMESE
#define XK_XKB_KEYS

```

6.2.25 X11/keysymdef.h

```

#define XK_space      0x0020
#define XK_exclam     0x0021
#define XK_quotedbl   0x0022

```

```

#define XK_numbersign 0x0023
#define XK_dollar      0x0024
#define XK_percent     0x0025
#define XK_ampersand   0x0026
#define XK_apostrophe  0x0027
#define XK_quoteright  0x0027
#define XK_parenleft   0x0028
#define XK_parenright  0x0029
#define XK_asterisk     0x002a
#define XK_plus        0x002b
#define XK_comma       0x002c
#define XK_minus       0x002d
#define XK_period      0x002e
#define XK_slash       0x002f
#define XK_0           0x0030
#define XK_1           0x0031
#define XK_2           0x0032
#define XK_3           0x0033
#define XK_4           0x0034
#define XK_5           0x0035
#define XK_6           0x0036
#define XK_7           0x0037
#define XK_8           0x0038
#define XK_9           0x0039
#define XK_colon       0x003a
#define XK_semicolon   0x003b
#define XK_less        0x003c
#define XK_equal       0x003d
#define XK_greater     0x003e
#define XK_question    0x003f
#define XK_at          0x0040
#define XK_A           0x0041
#define XK_B           0x0042
#define XK_C           0x0043
#define XK_D           0x0044
#define XK_E           0x0045
#define XK_F           0x0046
#define XK_G           0x0047
#define XK_H           0x0048
#define XK_I           0x0049
#define XK_J           0x004a
#define XK_K           0x004b
#define XK_L           0x004c
#define XK_M           0x004d
#define XK_N           0x004e
#define XK_O           0x004f
#define XK_P           0x0050
#define XK_Q           0x0051
#define XK_R           0x0052
#define XK_S           0x0053
#define XK_T           0x0054
#define XK_U           0x0055
#define XK_V           0x0056
#define XK_W           0x0057
#define XK_X           0x0058
#define XK_Y           0x0059
#define XK_Z           0x005a
#define XK_bracketleft 0x005b
#define XK_backslash   0x005c
#define XK_bracketright 0x005d
#define XK_asciicircum 0x005e
#define XK_underscore  0x005f
#define XK_grave       0x0060
#define XK_quoteleft   0x0060
#define XK_a           0x0061
#define XK_b           0x0062

```

```

#define XK_c      0x0063
#define XK_d      0x0064
#define XK_e      0x0065
#define XK_f      0x0066
#define XK_g      0x0067
#define XK_h      0x0068
#define XK_i      0x0069
#define XK_j      0x006a
#define XK_k      0x006b
#define XK_l      0x006c
#define XK_m      0x006d
#define XK_n      0x006e
#define XK_o      0x006f
#define XK_p      0x0070
#define XK_q      0x0071
#define XK_r      0x0072
#define XK_s      0x0073
#define XK_t      0x0074
#define XK_u      0x0075
#define XK_v      0x0076
#define XK_w      0x0077
#define XK_x      0x0078
#define XK_y      0x0079
#define XK_z      0x007a
#define XK_braceleft 0x007b
#define XK_bar 0x007c
#define XK_braceright 0x007d
#define XK_asciitilde 0x007e
#define XK_nobreakspace 0x00a0
#define XK_exclamdown 0x00a1
#define XK_cent 0x00a2
#define XK_sterling 0x00a3
#define XK_currency 0x00a4
#define XK_yen 0x00a5
#define XK_brokenbar 0x00a6
#define XK_section 0x00a7
#define XK_diaeresis 0x00a8
#define XK_copyright 0x00a9
#define XK_ordfeminine 0x00aa
#define XK_guillemotleft 0x00ab
#define XK_notsign 0x00ac
#define XK_hyphen 0x00ad
#define XK_registered 0x00ae
#define XK_macron 0x00af
#define XK_degree 0x00b0
#define XK_plusminus 0x00b1
#define XK_twosuperior 0x00b2
#define XK_threesuperior 0x00b3
#define XK_acute 0x00b4
#define XK_mu 0x00b5
#define XK_paragraph 0x00b6
#define XK_periodcentered 0x00b7
#define XK_cedilla 0x00b8
#define XK_onesuperior 0x00b9
#define XK_masculine 0x00ba
#define XK_guillemotright 0x00bb
#define XK_onequarter 0x00bc
#define XK_onehalf 0x00bd
#define XK_threequarters 0x00be
#define XK_questiondown 0x00bf
#define XK_Agrave 0x00c0
#define XK_Aacute 0x00c1
#define XK_Acircumflex 0x00c2
#define XK_Atilde 0x00c3
#define XK_Adiaeresis 0x00c4
#define XK_Aring 0x00c5

```

```

#define XK_AE      0x00c6
#define XK_Ccedilla 0x00c7
#define XK_Egrave   0x00c8
#define XK_Eacute   0x00c9
#define XK_Ecircumflex 0x00ca
#define XK_Ediaeresis 0x00cb
#define XK_Igrave   0x00cc
#define XK_Iacute   0x00cd
#define XK_Icircumflex 0x00ce
#define XK_Idiaeresis 0x00cf
#define XK_ETH      0x00d0
#define XK_Eth      0x00d0
#define XK_Ntilde   0x00d1
#define XK_Ograve   0x00d2
#define XK_Oacute   0x00d3
#define XK_Ocircumflex 0x00d4
#define XK_Otilde   0x00d5
#define XK_Odiaeresis 0x00d6
#define XK_multiply 0x00d7
#define XK_Ooblique 0x00d8
#define XK_Oslash   0x00d8
#define XK_Ugrave   0x00d9
#define XK_Uacute   0x00da
#define XK_Ucircumflex 0x00db
#define XK_Udiaeresis 0x00dc
#define XK_Yacute   0x00dd
#define XK_THORN    0x00de
#define XK_Thorn    0x00de
#define XK_ssharp   0x00df
#define XK_agrave   0x00e0
#define XK_aacute   0x00e1
#define XK_acircumflex 0x00e2
#define XK_atilde   0x00e3
#define XK_adiaeresis 0x00e4
#define XK_aring     0x00e5
#define XK_ae        0x00e6
#define XK_ccedilla 0x00e7
#define XK_egrave   0x00e8
#define XK_eacute   0x00e9
#define XK_ecircumflex 0x00ea
#define XK_ediaeresis 0x00eb
#define XK_igrave   0x00ec
#define XK_iacute   0x00ed
#define XK_icircumflex 0x00ee
#define XK_idiaeresis 0x00ef
#define XK_eth       0x00f0
#define XK_ntilde   0x00f1
#define XK_ograde   0x00f2
#define XK_oacute   0x00f3
#define XK_ocircumflex 0x00f4
#define XK_otilde   0x00f5
#define XK_odiaeresis 0x00f6
#define XK_division 0x00f7
#define XK_ooblique 0x00f8
#define XK_oslash   0x00f8
#define XK_ugrave   0x00f9
#define XK_uacute   0x00fa
#define XK_ucircumflex 0x00fb
#define XK_udiaeresis 0x00fc
#define XK_yacute   0x00fd
#define XK_thorn     0x00fe
#define XK_ydiaeresis 0x00ff
#define XK_Aogonek   0x01a1
#define XK_breve     0x01a2
#define XK_Lstroke   0x01a3
#define XK_Lcaron    0x01a5

```

```

#define XK_Sacute      0x01a6
#define XK_Scaron      0x01a9
#define XK_Scedilla    0x01aa
#define XK_Tcaron      0x01ab
#define XK_Zacute      0x01ac
#define XK_Zcaron      0x01ae
#define XK_Zabovedot   0x01af
#define XK_aogonek     0x01b1
#define XK_ogonek      0x01b2
#define XK_lstroke     0x01b3
#define XK_lcaron      0x01b5
#define XK_sacute      0x01b6
#define XK_caron       0x01b7
#define XK_scaron      0x01b9
#define XK_scedilla    0x01ba
#define XK_tcaron      0x01bb
#define XK_zacute      0x01bc
#define XK_doubleacute  0x01bd
#define XK_zcaron      0x01be
#define XK_zabovedot   0x01bf
#define XK_Racute      0x01c0
#define XK_Abreve      0x01c3
#define XK_Lacute      0x01c5
#define XK_Cacute      0x01c6
#define XK_Ccaron      0x01c8
#define XK_Eogonek     0x01ca
#define XK_Ecaron      0x01cc
#define XK_Dcaron      0x01cf
#define XK_Dstroke     0x01d0
#define XK_Nacute      0x01d1
#define XK_Ncaron      0x01d2
#define XK_Odoubleacute 0x01d5
#define XK_Rcaron      0x01d8
#define XK_Uring       0x01d9
#define XK_Udoubleacute 0x01db
#define XK_Tcedilla    0x01de
#define XK_racute      0x01e0
#define XK_abreve      0x01e3
#define XK_lacute      0x01e5
#define XK_cacute      0x01e6
#define XK_ccaron      0x01e8
#define XK_eogonek     0x01ea
#define XK_ecaron      0x01ec
#define XK_dcaron      0x01ef
#define XK_dstroke     0x01f0
#define XK_nacute      0x01f1
#define XK_ncaron      0x01f2
#define XK_odoubleacute 0x01f5
#define XK_rcaron      0x01f8
#define XK_uring       0x01f9
#define XK_udoubleacute 0x01fb
#define XK_tcedilla    0x01fe
#define XK_abovedot    0x01ff
#define XK_Hstroke     0x02a1
#define XK_Hcircumflex 0x02a6
#define XK_Iabovedot    0x02a9
#define XK_Gbreve      0x02ab
#define XK_Jcircumflex 0x02ac
#define XK_hstroke     0x02b1
#define XK_hcircumflex 0x02b6
#define XK_idotless    0x02b9
#define XK_gbreve      0x02bb
#define XK_jcircumflex 0x02bc
#define XK_Cabovedot    0x02c5
#define XK_Ccircumflex 0x02c6
#define XK_Gabovedot    0x02d5

```

```

#define XK_Gcircumflex 0x02d8
#define XK_Ubreve      0x02dd
#define XK_Scircumflex 0x02de
#define XK_cabovedot   0x02e5
#define XK_ccircumflex 0x02e6
#define XK_gabovedot   0x02f5
#define XK_gcircumflex 0x02f8
#define XK_ubreve      0x02fd
#define XK_scircumflex 0x02fe
#define XK_kappa       0x03a2
#define XK_kra 0x03a2
#define XK_Rcedilla    0x03a3
#define XK_Itilde      0x03a5
#define XK_Lcedilla    0x03a6
#define XK_Emacron     0x03aa
#define XK_Gcedilla    0x03ab
#define XK_Tslash      0x03ac
#define XK_rcedilla    0x03b3
#define XK_ityilde     0x03b5
#define XK_lcedilla    0x03b6
#define XK_emacron     0x03ba
#define XK_gcedilla    0x03bb
#define XK_tslash      0x03bc
#define XK_ENG 0x03bd
#define XK_eng 0x03bf
#define XK_Amacron     0x03c0
#define XK_Iogonek     0x03c7
#define XK_Eabovedot   0x03cc
#define XK_Imacron     0x03cf
#define XK_Ncedilla    0x03d1
#define XK_Omacron     0x03d2
#define XK_Kcedilla    0x03d3
#define XK_Uogonek     0x03d9
#define XK_Utilde      0x03dd
#define XK_Umacron     0x03de
#define XK_amacron     0x03e0
#define XK_iogonek     0x03e7
#define XK_eabovedot   0x03ec
#define XK_imacron     0x03ef
#define XK_ncedilla    0x03f1
#define XK_omacron     0x03f2
#define XK_kcedilla    0x03f3
#define XK_uogonek     0x03f9
#define XK_utilde      0x03fd
#define XK_umacron     0x03fe
#define XK_overline     0x047e
#define XK_kana_fullstop 0x04a1
#define XK_kana_openingbracket 0x04a2
#define XK_kana_closingbracket 0x04a3
#define XK_kana_comma 0x04a4
#define XK_kana_conjunctive 0x04a5
#define XK_kana_middledot 0x04a5
#define XK_kana_WO 0x04a6
#define XK_kana_a 0x04a7
#define XK_kana_i 0x04a8
#define XK_kana_u 0x04a9
#define XK_kana_e 0x04aa
#define XK_kana_o 0x04ab
#define XK_kana_ya 0x04ac
#define XK_kana_yu 0x04ad
#define XK_kana_yo 0x04ae
#define XK_kana_tsu 0x04af
#define XK_kana_tu 0x04af
#define XK_prolongedsound 0x04b0
#define XK_kana_A 0x04b1
#define XK_kana_I 0x04b2

```

```

#define XK_kana_U      0x04b3
#define XK_kana_E      0x04b4
#define XK_kana_O      0x04b5
#define XK_kana_KA     0x04b6
#define XK_kana_KI     0x04b7
#define XK_kana_KU     0x04b8
#define XK_kana_KE     0x04b9
#define XK_kana_KO     0x04ba
#define XK_kana_SA     0x04bb
#define XK_kana_SHI    0x04bc
#define XK_kana_SU     0x04bd
#define XK_kana_SE     0x04be
#define XK_kana_SO     0x04bf
#define XK_kana_TA     0x04c0
#define XK_kana_CHI    0x04c1
#define XK_kana_TI     0x04c1
#define XK_kana_TSU    0x04c2
#define XK_kana_TU     0x04c2
#define XK_kana_TE     0x04c3
#define XK_kana_TO     0x04c4
#define XK_kana_NA     0x04c5
#define XK_kana_NI     0x04c6
#define XK_kana_NU     0x04c7
#define XK_kana_NE     0x04c8
#define XK_kana_NO     0x04c9
#define XK_kana_HA     0x04ca
#define XK_kana_HI     0x04cb
#define XK_kana_FU     0x04cc
#define XK_kana_HU     0x04cc
#define XK_kana_HE     0x04cd
#define XK_kana_HO     0x04ce
#define XK_kana_MA     0x04cf
#define XK_kana_MI     0x04d0
#define XK_kana_MU     0x04d1
#define XK_kana_ME     0x04d2
#define XK_kana_MO     0x04d3
#define XK_kana_YA     0x04d4
#define XK_kana_YU     0x04d5
#define XK_kana_YO     0x04d6
#define XK_kana_RA     0x04d7
#define XK_kana_RI     0x04d8
#define XK_kana_RU     0x04d9
#define XK_kana_RE     0x04da
#define XK_kana_RO     0x04db
#define XK_kana_WA     0x04dc
#define XK_kana_N      0x04dd
#define XK_voicedsound 0x04de
#define XK_semivoicedsound 0x04df
#define XK_Arabic_comma 0x05ac
#define XK_Arabic_semicolon 0x05bb
#define XK_Arabic_question_mark 0x05bf
#define XK_Arabic_hamza 0x05c1
#define XK_Arabic_maddaonalef 0x05c2
#define XK_Arabic_hamzaonalef 0x05c3
#define XK_Arabic_hamzaonwaw 0x05c4
#define XK_Arabic_hamzaunderalef 0x05c5
#define XK_Arabic_hamzaonyeh 0x05c6
#define XK_Arabic_alef 0x05c7
#define XK_Arabic_beh 0x05c8
#define XK_Arabic_tehmarbuta 0x05c9
#define XK_Arabic_teh 0x05ca
#define XK_Arabic_theh 0x05cb
#define XK_Arabic_jeem 0x05cc
#define XK_Arabic_hah 0x05cd
#define XK_Arabic_khah 0x05ce
#define XK_Arabic_dal 0x05cf

```



```

#define XK_Arabic_thal 0x05d0
#define XK_Arabic_ra 0x05d1
#define XK_Arabic_zain 0x05d2
#define XK_Arabic_seen 0x05d3
#define XK_Arabic_sheen 0x05d4
#define XK_Arabic_sad 0x05d5
#define XK_Arabic_dad 0x05d6
#define XK_Arabic_tah 0x05d7
#define XK_Arabic_zah 0x05d8
#define XK_Arabic_ain 0x05d9
#define XK_Arabic_ghain 0x05da
#define XK_Arabic_tatweel 0x05e0
#define XK_Arabic_feh 0x05e1
#define XK_Arabic_qaf 0x05e2
#define XK_Arabic_kaf 0x05e3
#define XK_Arabic_lam 0x05e4
#define XK_Arabic_meem 0x05e5
#define XK_Arabic_noon 0x05e6
#define XK_Arabic_ha 0x05e7
#define XK_Arabic_heh 0x05e7
#define XK_Arabic_waw 0x05e8
#define XK_Arabic_alefmaksura 0x05e9
#define XK_Arabic_yeh 0x05ea
#define XK_Arabic_fathatan 0x05eb
#define XK_Arabic_dammatan 0x05ec
#define XK_Arabic_kasratan 0x05ed
#define XK_Arabic_fatha 0x05ee
#define XK_Arabic_damma 0x05ef
#define XK_Arabic_kasra 0x05f0
#define XK_Arabic_shadda 0x05f1
#define XK_Arabic_sukun 0x05f2
#define XK_Serbian_dje 0x06a1
#define XK_Macedonia_gje 0x06a2
#define XK_Cyrillic_io 0x06a3
#define XK_Ukrainian_ie 0x06a4
#define XK_Ukrainian_je 0x06a4
#define XK_Macedonia_dse 0x06a5
#define XK_Ukrainian_i 0x06a6
#define XK_Ukrainian_i 0x06a6
#define XK_Ukrainian_yi 0x06a7
#define XK_Ukrainian_yi 0x06a7
#define XK_Cyrillic_je 0x06a8
#define XK_Serbian_je 0x06a8
#define XK_Cyrillic_lje 0x06a9
#define XK_Serbian_lje 0x06a9
#define XK_Cyrillic_nje 0x06aa
#define XK_Serbian_nje 0x06aa
#define XK_Serbian_tshe 0x06ab
#define XK_Macedonia_kje 0x06ac
#define XK_Ukrainian_ghe_with_upturn 0x06ad
#define XK_Byelorussian_shortu 0x06ae
#define XK_Cyrillic_dzhe 0x06af
#define XK_Serbian_dze 0x06af
#define XK_numerosign 0x06b0
#define XK_Serbian_DJE 0x06b1
#define XK_Macedonia_GJE 0x06b2
#define XK_Cyrillic_IO 0x06b3
#define XK_Ukrainian_IE 0x06b4
#define XK_Ukrainian_JE 0x06b4
#define XK_Macedonia_DSE 0x06b5
#define XK_Ukrainian_I 0x06b6
#define XK_Ukrainian_I 0x06b6
#define XK_Ukrainian_YI 0x06b7
#define XK_Ukrainian_YI 0x06b7
#define XK_Cyrillic_JE 0x06b8
#define XK_Serbian_JE 0x06b8

```

```

#define XK_Cyrillic_LJE 0x06b9
#define XK_Serbian_LJE 0x06b9
#define XK_Cyrillic_NJE 0x06ba
#define XK_Serbian_NJE 0x06ba
#define XK_Serbian_TSHE 0x06bb
#define XK_Macedonia_KJE 0x06bc
#define XK_Ukrainian_GHE_WITH_UPTURN 0x06bd
#define XK_Byelorussian_SHORTU 0x06be
#define XK_Cyrillic_DZHE 0x06bf
#define XK_Serbian_DZE 0x06bf
#define XK_Cyrillic_yu 0x06c0
#define XK_Cyrillic_a 0x06c1
#define XK_Cyrillic_be 0x06c2
#define XK_Cyrillic_tse 0x06c3
#define XK_Cyrillic_de 0x06c4
#define XK_Cyrillic_ie 0x06c5
#define XK_Cyrillic_ef 0x06c6
#define XK_Cyrillic_ghe 0x06c7
#define XK_Cyrillic_ha 0x06c8
#define XK_Cyrillic_i 0x06c9
#define XK_Cyrillic_shorti 0x06ca
#define XK_Cyrillic_ka 0x06cb
#define XK_Cyrillic_el 0x06cc
#define XK_Cyrillic_em 0x06cd
#define XK_Cyrillic_en 0x06ce
#define XK_Cyrillic_o 0x06cf
#define XK_Cyrillic_pe 0x06d0
#define XK_Cyrillic_ya 0x06d1
#define XK_Cyrillic_er 0x06d2
#define XK_Cyrillic_es 0x06d3
#define XK_Cyrillic_te 0x06d4
#define XK_Cyrillic_u 0x06d5
#define XK_Cyrillic_zhe 0x06d6
#define XK_Cyrillic_ve 0x06d7
#define XK_Cyrillic_softsign 0x06d8
#define XK_Cyrillic_yeru 0x06d9
#define XK_Cyrillic_ze 0x06da
#define XK_Cyrillic_sha 0x06db
#define XK_Cyrillic_e 0x06dc
#define XK_Cyrillic_shcha 0x06dd
#define XK_Cyrillic_ache 0x06de
#define XK_Cyrillic_hardsign 0x06df
#define XK_Cyrillic_YU 0x06e0
#define XK_Cyrillic_A 0x06e1
#define XK_Cyrillic_BE 0x06e2
#define XK_Cyrillic_TSE 0x06e3
#define XK_Cyrillic_DE 0x06e4
#define XK_Cyrillic_IE 0x06e5
#define XK_Cyrillic_EF 0x06e6
#define XK_Cyrillic_GHE 0x06e7
#define XK_Cyrillic_HA 0x06e8
#define XK_Cyrillic_I 0x06e9
#define XK_Cyrillic_SHORTI 0x06ea
#define XK_Cyrillic_KA 0x06eb
#define XK_Cyrillic_EL 0x06ec
#define XK_Cyrillic_EM 0x06ed
#define XK_Cyrillic_EN 0x06ee
#define XK_Cyrillic_O 0x06ef
#define XK_Cyrillic_PE 0x06f0
#define XK_Cyrillic_YA 0x06f1
#define XK_Cyrillic_ER 0x06f2
#define XK_Cyrillic_ES 0x06f3
#define XK_Cyrillic_TE 0x06f4
#define XK_Cyrillic_U 0x06f5
#define XK_Cyrillic_ZHE 0x06f6
#define XK_Cyrillic_VE 0x06f7

```

```

#define XK_Cyrillic_SOFTSIGN      0x06f8
#define XK_Cyrillic_YERU          0x06f9
#define XK_Cyrillic_ZE            0x06fa
#define XK_Cyrillic_SHA           0x06fb
#define XK_Cyrillic_E             0x06fc
#define XK_Cyrillic_SHCHA         0x06fd
#define XK_Cyrillic_CHE           0x06fe
#define XK_Cyrillic_HARDSIGN      0x06ff
#define XK_Greek_ALPHAaccent      0x07a1
#define XK_Greek_EPSILONaccent    0x07a2
#define XK_Greek_ETAaccent        0x07a3
#define XK_Greek_IOTAaccent       0x07a4
#define XK_Greek_IOTAdiaeresis    0x07a5
#define XK_Greek_IOTAdieresis     0x07a6
#define XK_Greek_OMICRONaccent    0x07a7
#define XK_Greek_UPSILONaccent    0x07a8
#define XK_Greek_UPSILONdieresis  0x07a9
#define XK_Greek_OMEGAaccent      0x07ab
#define XK_Greek_accentdieresis   0x07ae
#define XK_Greek_horizbar         0x07af
#define XK_Greek_alphaaccent      0x07b1
#define XK_Greek_epsilonaccent    0x07b2
#define XK_Greek_etaaccent        0x07b3
#define XK_Greek_iotaaccent       0x07b4
#define XK_Greek_iotadieresis     0x07b5
#define XK_Greek_iotaaccentdieresis 0x07b6
#define XK_Greek_omicronaccent     0x07b7
#define XK_Greek_upsilonaccent     0x07b8
#define XK_Greek_upsilondieresis   0x07b9
#define XK_Greek_upsilonaccentdieresis 0x07ba
#define XK_Greek_omegaaccent       0x07bb
#define XK_Greek_ALPHA            0x07c1
#define XK_Greek_BETA             0x07c2
#define XK_Greek_GAMMA            0x07c3
#define XK_Greek_DELTA            0x07c4
#define XK_Greek_EPSILON          0x07c5
#define XK_Greek_ZETA             0x07c6
#define XK_Greek_ETA              0x07c7
#define XK_Greek_THETA            0x07c8
#define XK_Greek_IOTA             0x07c9
#define XK_Greek_KAPPA            0x07ca
#define XK_Greek_LAMBDA           0x07cb
#define XK_Greek_LAMDA            0x07cb
#define XK_Greek_MU               0x07cc
#define XK_Greek_NU               0x07cd
#define XK_Greek_XI               0x07ce
#define XK_Greek_OMICRON           0x07cf
#define XK_Greek_PI               0x07d0
#define XK_Greek_RHO              0x07d1
#define XK_Greek_SIGMA            0x07d2
#define XK_Greek_TAU              0x07d4
#define XK_Greek_UPSILON          0x07d5
#define XK_Greek_PHI              0x07d6
#define XK_Greek_CHI              0x07d7
#define XK_Greek_PSI              0x07d8
#define XK_Greek_OMEGA            0x07d9
#define XK_Greek_alpha            0x07e1
#define XK_Greek_beta             0x07e2
#define XK_Greek_gamma            0x07e3
#define XK_Greek_delta            0x07e4
#define XK_Greek_epsilon          0x07e5
#define XK_Greek_zeta             0x07e6
#define XK_Greek_eta              0x07e7
#define XK_Greek_theta            0x07e8
#define XK_Greek_iota             0x07e9
#define XK_Greek_kappa            0x07ea

```

```

#define XK_Greek_lambda 0x07eb
#define XK_Greek_lamda 0x07eb
#define XK_Greek_mu 0x07ec
#define XK_Greek_nu 0x07ed
#define XK_Greek_xi 0x07ee
#define XK_Greek_omicron 0x07ef
#define XK_Greek_pi 0x07f0
#define XK_Greek_rho 0x07f1
#define XK_Greek_sigma 0x07f2
#define XK_Greek_finalsmallsigma 0x07f3
#define XK_Greek_tau 0x07f4
#define XK_Greek_upsilon 0x07f5
#define XK_Greek_phi 0x07f6
#define XK_Greek_chi 0x07f7
#define XK_Greek_psi 0x07f8
#define XK_Greek_omega 0x07f9
#define XK_leftradical 0x08a1
#define XK_topleftradical 0x08a2
#define XK_horizconnector 0x08a3
#define XK_topintegral 0x08a4
#define XK_botintegral 0x08a5
#define XK_vertconnector 0x08a6
#define XK_topleftsqbracket 0x08a7
#define XK_botleftsqbracket 0x08a8
#define XK_toprightsqbracket 0x08a9
#define XK_botrightsqbracket 0x08aa
#define XK_topleftparens 0x08ab
#define XK_botleftparens 0x08ac
#define XK_toprightparens 0x08ad
#define XK_botrightparens 0x08ae
#define XK_leftmiddlecurlybrace 0x08af
#define XK_rightmiddlecurlybrace 0x08b0
#define XK_topleftsummation 0x08b1
#define XK_botleftsummation 0x08b2
#define XK_topvertsummationconnector 0x08b3
#define XK_botvertsummationconnector 0x08b4
#define XK_toprightsummation 0x08b5
#define XK_botrightsummation 0x08b6
#define XK_rightmiddlesummation 0x08b7
#define XK_lessthanequal 0x08bc
#define XK_notequal 0x08bd
#define XK_greaterethanequal 0x08be
#define XK_integral 0x08bf
#define XK_therefore 0x08c0
#define XK_variation 0x08c1
#define XK_infinity 0x08c2
#define XK_nabla 0x08c5
#define XK_approximate 0x08c8
#define XK_similarequal 0x08c9
#define XK_ifonlyif 0x08cd
#define XK_implies 0x08ce
#define XK_identical 0x08cf
#define XK_radical 0x08d6
#define XK_includedin 0x08da
#define XK_includes 0x08db
#define XK_intersection 0x08dc
#define XK_union 0x08dd
#define XK_logicaland 0x08de
#define XK_logicalor 0x08df
#define XK_partialderivative 0x08ef
#define XK_function 0x08f6
#define XK_leftarrow 0x08fb
#define XK_uparrow 0x08fc
#define XK_rightarrow 0x08fd
#define XK_downarrow 0x08fe
#define XK_blank 0x09df

```

```

#define XK_soliddiamond 0x09e0
#define XK_checkerboard 0x09e1
#define XK_ht 0x09e2
#define XK_ff 0x09e3
#define XK_cr 0x09e4
#define XK_lf 0x09e5
#define XK_nl 0x09e8
#define XK_vt 0x09e9
#define XK_lowrightcorner 0x09ea
#define XK_uprightcorner 0x09eb
#define XK_upleftcorner 0x09ec
#define XK_lowleftcorner 0x09ed
#define XK_crossinglines 0x09ee
#define XK_horizlinescan1 0x09ef
#define XK_horizlinescan3 0x09f0
#define XK_horizlinescan5 0x09f1
#define XK_horizlinescan7 0x09f2
#define XK_horizlinescan9 0x09f3
#define XK_leftt 0x09f4
#define XK_rightt 0x09f5
#define XK_bott 0x09f6
#define XK_topt 0x09f7
#define XK_vertbar 0x09f8
#define XK_emspace 0x0aa1
#define XK_enspace 0x0aa2
#define XK_em3space 0x0aa3
#define XK_em4space 0x0aa4
#define XK_digitspace 0x0aa5
#define XK_punctspace 0x0aa6
#define XK_thinspace 0x0aa7
#define XK_hairspace 0x0aa8
#define XK_emdash 0x0aa9
#define XK_endash 0x0aaa
#define XK_signifblank 0x0aac
#define XK_ellipsis 0x0aae
#define XK_doubbaselinedot 0x0aaf
#define XK_onethird 0x0ab0
#define XK_twothirds 0x0ab1
#define XK_onefifth 0x0ab2
#define XK_twofifths 0x0ab3
#define XK_threefifths 0x0ab4
#define XK_fourfifths 0x0ab5
#define XK_onesixth 0x0ab6
#define XK_fivesixths 0x0ab7
#define XK_careof 0x0ab8
#define XK_figdash 0x0abb
#define XK_leftanglebracket 0x0abc
#define XK_decimalpoint 0x0abd
#define XK_rightanglebracket 0x0abe
#define XK_marker 0x0abf
#define XK_oneeighth 0x0ac3
#define XK_threeeighths 0x0ac4
#define XK_fiveeighths 0x0ac5
#define XK_seveneighths 0x0ac6
#define XK_trademark 0x0ac9
#define XK_signaturemark 0x0aca
#define XK_trademarkincircle 0x0acb
#define XK_leftopentriangle 0x0acc
#define XK_rightopentriangle 0x0acd
#define XK_emopencircle 0x0ace
#define XK_emopenrectangle 0x0acf
#define XK_leftsinglequotemark 0x0ad0
#define XK_rightsinglequotemark 0x0ad1
#define XK_leftdoublequotemark 0x0ad2
#define XK_rightdoublequotemark 0x0ad3
#define XK_prescription 0x0ad4

```

```

#define XK_minutes      0x0ad6
#define XK_seconds      0x0ad7
#define XK_latincross   0x0ad9
#define XK_hexagram     0x0ada
#define XK_filledrectbullet 0x0adb
#define XK_filledlefttribullet 0x0adc
#define XK_filledrighttribullet 0x0add
#define XK_emfilledcircle 0x0ade
#define XK_emfilledrect 0x0adf
#define XK_enopencircbullet 0x0ae0
#define XK_enopensquarebullet 0x0ae1
#define XK_openrectbullet 0x0ae2
#define XK_opentribulletup 0x0ae3
#define XK_opentribulletdown 0x0ae4
#define XK_openstar     0x0ae5
#define XK_enfilledcircbullet 0x0ae6
#define XK_enfilledsqbullet 0x0ae7
#define XK_filledtribulletup 0x0ae8
#define XK_filledtribulletdown 0x0ae9
#define XK_leftpointer 0x0aea
#define XK_rightpointer 0x0aeb
#define XK_club 0x0aec
#define XK_diamond 0x0aed
#define XK_heart 0x0aee
#define XK_maltesecross 0x0af0
#define XK_dagger 0x0af1
#define XK_doubledagger 0x0af2
#define XK_checkmark 0x0af3
#define XK_ballotcross 0x0af4
#define XK_musicalsharp 0x0af5
#define XK_musicalflat 0x0af6
#define XK_malesymbol 0x0af7
#define XK_femalesymbol 0x0af8
#define XK_telephone 0x0af9
#define XK_telephonerecorder 0x0afa
#define XK_phonographcopyright 0x0afb
#define XK_caret 0x0afc
#define XK_singlelowquotemark 0x0afd
#define XK_doublelowquotemark 0x0afe
#define XK_cursor 0x0aff
#define XK_leftcaret 0x0ba3
#define XK_rightcaret 0x0ba6
#define XK_downcaret 0x0ba8
#define XK_upcaret 0x0ba9
#define XK_overbar 0x0bc0
#define XK_downtack 0x0bc2
#define XK_upshoe 0x0bc3
#define XK_downstile 0x0bc4
#define XK_underbar 0x0bc6
#define XK_jot 0x0bca
#define XK_quad 0x0bcc
#define XK_uptack 0x0bce
#define XK_circle 0x0bcf
#define XK_upstile 0x0bd3
#define XK_downshoe 0x0bd6
#define XK_rightshoe 0x0bd8
#define XK_leftshoe 0x0bda
#define XK_leftttack 0x0bdc
#define XK_rightttack 0x0bfc
#define XK_hebrew_doublelowline 0x0cdf
#define XK_hebrew_aleph 0x0ce0
#define XK_hebrew_bet 0x0ce1
#define XK_hebrew_beth 0x0ce1
#define XK_hebrew_gimel 0x0ce2
#define XK_hebrew_gimmel 0x0ce2
#define XK_hebrew_dalet 0x0ce3

```

```

#define XK_hebrew_daleth      0x0ce3
#define XK_hebrew_he          0x0ce4
#define XK_hebrew_waw         0x0ce5
#define XK_hebrew_zain        0x0ce6
#define XK_hebrew_zayin       0x0ce6
#define XK_hebrew_chet        0x0ce7
#define XK_hebrew_het         0x0ce7
#define XK_hebrew_tet         0x0ce8
#define XK_hebrew_teth        0x0ce8
#define XK_hebrew_yod         0x0ce9
#define XK_hebrew_finalkaph    0x0cea
#define XK_hebrew_kaph        0x0ceb
#define XK_hebrew_lamed       0x0cec
#define XK_hebrew_finalmem     0x0ced
#define XK_hebrew_mem         0x0cee
#define XK_hebrew_finalnun     0x0cef
#define XK_hebrew_nun         0x0cf0
#define XK_hebrew_samech      0x0cf1
#define XK_hebrew_samekh      0x0cf1
#define XK_hebrew_ayin        0x0cf2
#define XK_hebrew_finalpe     0x0cf3
#define XK_hebrew_pe          0x0cf4
#define XK_hebrew_finalzade    0x0cf5
#define XK_hebrew_finalzadi    0x0cf5
#define XK_hebrew_zade        0x0cf6
#define XK_hebrew_zadi        0x0cf6
#define XK_hebrew_kuf         0x0cf7
#define XK_hebrew_qoph        0x0cf7
#define XK_hebrew_resh        0x0cf8
#define XK_hebrew_shin        0x0cf9
#define XK_hebrew_taf         0x0cfa
#define XK_hebrew_taw         0x0cfa
#define XK_Thai_kokai         0x0da1
#define XK_Thai_khokhai       0x0da2
#define XK_Thai_khokhuat      0x0da3
#define XK_Thai_khokhwai      0x0da4
#define XK_Thai_khokhon       0x0da5
#define XK_Thai_khorakhang     0x0da6
#define XK_Thai_ngongu        0x0da7
#define XK_Thai_chochan       0x0da8
#define XK_Thai_choching      0x0da9
#define XK_Thai_chochang      0x0daa
#define XK_Thai_soso          0x0dab
#define XK_Thai_chochoe       0x0dac
#define XK_Thai_yoying        0x0dad
#define XK_Thai_dochada       0x0dae
#define XK_Thai_topatak       0x0daf
#define XK_Thai_thothan       0x0db0
#define XK_Thai_thonangmontho  0x0db1
#define XK_Thai_thophuthao     0x0db2
#define XK_Thai_nonen         0x0db3
#define XK_Thai_dodek         0x0db4
#define XK_Thai_totao         0x0db5
#define XK_Thai_thothung      0x0db6
#define XK_Thai_thothahan     0x0db7
#define XK_Thai_thothong      0x0db8
#define XK_Thai_nonu         0x0db9
#define XK_Thai_bobaimai      0x0dba
#define XK_Thai_popla         0x0dbb
#define XK_Thai_phophung      0x0dbc
#define XK_Thai_fofa          0x0dbd
#define XK_Thai_phophan       0x0dbe
#define XK_Thai_fofan         0x0dbf
#define XK_Thai_phosamphao     0x0dc0
#define XK_Thai_moma          0x0dc1
#define XK_Thai_yoyak         0x0dc2

```



```

#define XK_Thai_rorua 0x0dc3
#define XK_Thai_ru 0x0dc4
#define XK_Thai_loling 0x0dc5
#define XK_Thai_lu 0x0dc6
#define XK_Thai_wowaen 0x0dc7
#define XK_Thai_sosala 0x0dc8
#define XK_Thai_sorusi 0x0dc9
#define XK_Thai_sosua 0x0dca
#define XK_Thai_hohip 0x0dcb
#define XK_Thai_lochula 0x0dcc
#define XK_Thai_oang 0x0dcd
#define XK_Thai_honokhuk 0x0dce
#define XK_Thai_paiyannoi 0x0dcf
#define XK_Thai_saraa 0x0dd0
#define XK_Thai_maihanakat 0x0dd1
#define XK_Thai_saraaa 0x0dd2
#define XK_Thai_saraam 0x0dd3
#define XK_Thai_sarai 0x0dd4
#define XK_Thai_saraii 0x0dd5
#define XK_Thai_saraue 0x0dd6
#define XK_Thai_sarauee 0x0dd7
#define XK_Thai_sarau 0x0dd8
#define XK_Thai_sarauu 0x0dd9
#define XK_Thai_phinthu 0x0dda
#define XK_Thai_maihanakat_maitho 0x0dde
#define XK_Thai_baht 0x0ddf
#define XK_Thai_sarae 0x0de0
#define XK_Thai_saraae 0x0de1
#define XK_Thai_sarao 0x0de2
#define XK_Thai_saraaimaimuan 0x0de3
#define XK_Thai_saraaimaimalai 0x0de4
#define XK_Thai_lakkhangyao 0x0de5
#define XK_Thai_maiyamok 0x0de6
#define XK_Thai_maitaikhu 0x0de7
#define XK_Thai_maiek 0x0de8
#define XK_Thai_maitho 0x0de9
#define XK_Thai_maitri 0x0dea
#define XK_Thai_maichattawa 0x0deb
#define XK_Thai_thanthakhath 0x0dec
#define XK_Thai_nikhahit 0x0ded
#define XK_Thai_leksun 0x0df0
#define XK_Thai_leknung 0x0df1
#define XK_Thai_leksong 0x0df2
#define XK_Thai_leksam 0x0df3
#define XK_Thai_leksi 0x0df4
#define XK_Thai_lekha 0x0df5
#define XK_Thai_lekhok 0x0df6
#define XK_Thai_lekchet 0x0df7
#define XK_Thai_lekpaet 0x0df8
#define XK_Thai_lekkao 0x0df9
#define XK_Hangul_Kiyeog 0x0ea1
#define XK_Hangul_SsangKiyeog 0x0ea2
#define XK_Hangul_KiyeogSios 0x0ea3
#define XK_Hangul_Nieun 0x0ea4
#define XK_Hangul_NieunJieuj 0x0ea5
#define XK_Hangul_NieunHieuh 0x0ea6
#define XK_Hangul_Dikeud 0x0ea7
#define XK_Hangul_SsangDikeud 0x0ea8
#define XK_Hangul_Rieul 0x0ea9
#define XK_Hangul_RieulKiyeog 0x0eaa
#define XK_Hangul_RieulMieum 0x0eab
#define XK_Hangul_RieulPieub 0x0eac
#define XK_Hangul_RieulSios 0x0ead
#define XK_Hangul_RieulTieut 0x0eae
#define XK_Hangul_RieulPhieuf 0x0eaf
#define XK_Hangul_RieulHieuh 0x0eb0

```



```

#define XK_Hangul_Mieum 0x0eb1
#define XK_Hangul_Pieub 0x0eb2
#define XK_Hangul_SsangPieub 0x0eb3
#define XK_Hangul_PieubSios 0x0eb4
#define XK_Hangul_Sios 0x0eb5
#define XK_Hangul_SsangSios 0x0eb6
#define XK_Hangul_Ieung 0x0eb7
#define XK_Hangul_Jieuj 0x0eb8
#define XK_Hangul_SsangJieuj 0x0eb9
#define XK_Hangul_Cieuc 0x0eba
#define XK_Hangul_Khieuq 0x0ebb
#define XK_Hangul_Tieut 0x0ebc
#define XK_Hangul_Phieuf 0x0ebd
#define XK_Hangul_Hieuh 0x0ebe
#define XK_Hangul_A 0x0ebf
#define XK_Hangul_AE 0x0ec0
#define XK_Hangul_YA 0x0ec1
#define XK_Hangul_YAE 0x0ec2
#define XK_Hangul_EO 0x0ec3
#define XK_Hangul_E 0x0ec4
#define XK_Hangul_YEO 0x0ec5
#define XK_Hangul_YE 0x0ec6
#define XK_Hangul_O 0x0ec7
#define XK_Hangul_WA 0x0ec8
#define XK_Hangul_WAE 0x0ec9
#define XK_Hangul_OE 0x0eca
#define XK_Hangul_YO 0x0ecb
#define XK_Hangul_U 0x0ecc
#define XK_Hangul_WEO 0x0ecd
#define XK_Hangul_WE 0x0ece
#define XK_Hangul_WI 0x0ecf
#define XK_Hangul_YU 0x0ed0
#define XK_Hangul_EU 0x0ed1
#define XK_Hangul_YI 0x0ed2
#define XK_Hangul_I 0x0ed3
#define XK_Hangul_J_Kiyeog 0x0ed4
#define XK_Hangul_J_SsangKiyeog 0x0ed5
#define XK_Hangul_J_KiyeogSios 0x0ed6
#define XK_Hangul_J_Nieun 0x0ed7
#define XK_Hangul_J_NieunJieuj 0x0ed8
#define XK_Hangul_J_NieunHieuh 0x0ed9
#define XK_Hangul_J_Dikeud 0x0eda
#define XK_Hangul_J_Rieul 0x0edb
#define XK_Hangul_J_RieulKiyeog 0x0edc
#define XK_Hangul_J_RieulMieum 0x0edd
#define XK_Hangul_J_RieulPieub 0x0ede
#define XK_Hangul_J_RieulSios 0x0edf
#define XK_Hangul_J_RieulTieut 0x0ee0
#define XK_Hangul_J_RieulPhieuf 0x0ee1
#define XK_Hangul_J_RieulHieuh 0x0ee2
#define XK_Hangul_J_Mieum 0x0ee3
#define XK_Hangul_J_Pieub 0x0ee4
#define XK_Hangul_J_PieubSios 0x0ee5
#define XK_Hangul_J_Sios 0x0ee6
#define XK_Hangul_J_SsangSios 0x0ee7
#define XK_Hangul_J_Ieung 0x0ee8
#define XK_Hangul_J_Jieuj 0x0ee9
#define XK_Hangul_J_Cieuc 0x0eea
#define XK_Hangul_J_Khieuq 0x0eeb
#define XK_Hangul_J_Tieut 0x0eec
#define XK_Hangul_J_Phieuf 0x0eed
#define XK_Hangul_J_Hieuh 0x0eee
#define XK_Hangul_RieulYeorinHieuh 0x0eef
#define XK_Hangul_SunkyeongeumMieum 0x0ef0
#define XK_Hangul_SunkyeongeumPieub 0x0ef1
#define XK_Hangul_PanSios 0x0ef2

```

```

#define XK_Hangul_KkogjiDalrinJeung 0x0ef3
#define XK_Hangul_SunkyeongeumPhieuf 0x0ef4
#define XK_Hangul_YeorinHieuh 0x0ef5
#define XK_Hangul_AraeA 0x0ef6
#define XK_Hangul_AraeAE 0x0ef7
#define XK_Hangul_J_PanSios 0x0ef8
#define XK_Hangul_J_KkogjiDalrinJeung 0x0ef9
#define XK_Hangul_J_YeorinHieuh 0x0efa
#define XK_Korean_Won 0x0eff
#define XK_Ibreve 0x100012c
#define XK_ibreve 0x100012d
#define XK_Wcircumflex 0x1000174
#define XK_wcircumflex 0x1000175
#define XK_Ycircumflex 0x1000176
#define XK_ycircumflex 0x1000177
#define XK_SCHWA 0x100018f
#define XK_Obarred 0x100019f
#define XK_Ohorn 0x10001a0
#define XK_ohorn 0x10001a1
#define XK_Uhorn 0x10001af
#define XK_uhorn 0x10001b0
#define XK_Zstroke 0x10001b5
#define XK_zstroke 0x10001b6
#define XK_Ocaron 0x10001d1
#define XK_ocaron 0x10001d2
#define XK_Gcaron 0x10001e6
#define XK_gcaron 0x10001e7
#define XK_schwa 0x1000259
#define XK_obarred 0x1000275
#define XK_Cyrillic_GHE_bar 0x1000492
#define XK_Cyrillic_ghe_bar 0x1000493
#define XK_Cyrillic_ZHE_descender 0x1000496
#define XK_Cyrillic_zhe_descender 0x1000497
#define XK_Cyrillic_KA_descender 0x100049a
#define XK_Cyrillic_ka_descender 0x100049b
#define XK_Cyrillic_KA_vertstroke 0x100049c
#define XK_Cyrillic_ka_vertstroke 0x100049d
#define XK_Cyrillic_EN_descender 0x10004a2
#define XK_Cyrillic_en_descender 0x10004a3
#define XK_Cyrillic_U_straight 0x10004ae
#define XK_Cyrillic_u_straight 0x10004af
#define XK_Cyrillic_U_straight_bar 0x10004b0
#define XK_Cyrillic_u_straight_bar 0x10004b1
#define XK_Cyrillic_HA_descender 0x10004b2
#define XK_Cyrillic_ha_descender 0x10004b3
#define XK_Cyrillic_CHE_descender 0x10004b6
#define XK_Cyrillic_che_descender 0x10004b7
#define XK_Cyrillic_CHE_vertstroke 0x10004b8
#define XK_Cyrillic_che_vertstroke 0x10004b9
#define XK_Cyrillic_SHHA 0x10004ba
#define XK_Cyrillic_shha 0x10004bb
#define XK_Cyrillic_SCHWA 0x10004d8
#define XK_Cyrillic_schwa 0x10004d9
#define XK_Cyrillic_I_macron 0x10004e2
#define XK_Cyrillic_i_macron 0x10004e3
#define XK_Cyrillic_O_bar 0x10004e8
#define XK_Cyrillic_o_bar 0x10004e9
#define XK_Cyrillic_U_macron 0x10004ee
#define XK_Cyrillic_u_macron 0x10004ef
#define XK_Armenian_AYB 0x1000531
#define XK_Armenian_BEN 0x1000532
#define XK_Armenian_GIM 0x1000533
#define XK_Armenian_DA 0x1000534
#define XK_Armenian_YECH 0x1000535
#define XK_Armenian_ZA 0x1000536
#define XK_Armenian_E 0x1000537

```

```

#define XK_Armenian_AT 0x1000538
#define XK_Armenian_TO 0x1000539
#define XK_Armenian_ZHE 0x100053a
#define XK_Armenian_INI 0x100053b
#define XK_Armenian_LYUN 0x100053c
#define XK_Armenian_KHE 0x100053d
#define XK_Armenian_TSA 0x100053e
#define XK_Armenian_KEN 0x100053f
#define XK_Armenian_HO 0x1000540
#define XK_Armenian_DZA 0x1000541
#define XK_Armenian_GHAT 0x1000542
#define XK_Armenian_TCHE 0x1000543
#define XK_Armenian_MEN 0x1000544
#define XK_Armenian_HI 0x1000545
#define XK_Armenian_NU 0x1000546
#define XK_Armenian_SHA 0x1000547
#define XK_Armenian_VO 0x1000548
#define XK_Armenian_CHA 0x1000549
#define XK_Armenian_PE 0x100054a
#define XK_Armenian_JE 0x100054b
#define XK_Armenian_RA 0x100054c
#define XK_Armenian_SE 0x100054d
#define XK_Armenian_VEV 0x100054e
#define XK_Armenian_TYUN 0x100054f
#define XK_Armenian_RE 0x1000550
#define XK_Armenian_TSO 0x1000551
#define XK_Armenian_VYUN 0x1000552
#define XK_Armenian_PYUR 0x1000553
#define XK_Armenian_KE 0x1000554
#define XK_Armenian_O 0x1000555
#define XK_Armenian_FE 0x1000556
#define XK_Armenian_apostrophe 0x100055a
#define XK_Armenian_accent 0x100055b
#define XK_Armenian_shesht 0x100055b
#define XK_Armenian_amanak 0x100055c
#define XK_Armenian_exclam 0x100055c
#define XK_Armenian_but 0x100055d
#define XK_Armenian_separation_mark 0x100055d
#define XK_Armenian_paruyk 0x100055e
#define XK_Armenian_question 0x100055e
#define XK_Armenian_ayb 0x1000561
#define XK_Armenian_ben 0x1000562
#define XK_Armenian_gim 0x1000563
#define XK_Armenian_da 0x1000564
#define XK_Armenian_yech 0x1000565
#define XK_Armenian_za 0x1000566
#define XK_Armenian_e 0x1000567
#define XK_Armenian_at 0x1000568
#define XK_Armenian_to 0x1000569
#define XK_Armenian_zhe 0x100056a
#define XK_Armenian_ini 0x100056b
#define XK_Armenian_lyun 0x100056c
#define XK_Armenian_khe 0x100056d
#define XK_Armenian_tsa 0x100056e
#define XK_Armenian_ken 0x100056f
#define XK_Armenian_ho 0x1000570
#define XK_Armenian_dza 0x1000571
#define XK_Armenian_ghat 0x1000572
#define XK_Armenian_tche 0x1000573
#define XK_Armenian_men 0x1000574
#define XK_Armenian_hi 0x1000575
#define XK_Armenian_nu 0x1000576
#define XK_Armenian_sha 0x1000577
#define XK_Armenian_vo 0x1000578
#define XK_Armenian_cha 0x1000579
#define XK_Armenian_pe 0x100057a

```

```

#define XK_Armenian_je 0x100057b
#define XK_Armenian_ra 0x100057c
#define XK_Armenian_se 0x100057d
#define XK_Armenian_vev 0x100057e
#define XK_Armenian_tyun 0x100057f
#define XK_Armenian_re 0x1000580
#define XK_Armenian_tso 0x1000581
#define XK_Armenian_vyun 0x1000582
#define XK_Armenian_pyur 0x1000583
#define XK_Armenian_ke 0x1000584
#define XK_Armenian_o 0x1000585
#define XK_Armenian_fe 0x1000586
#define XK_Armenian_ligature_ew 0x1000587
#define XK_Armenian_full_stop 0x1000589
#define XK_Armenian_verjaket 0x1000589
#define XK_Armenian_hyphen 0x100058a
#define XK_Armenian_yentamna 0x100058a
#define XK_Arabic_madda_above 0x1000653
#define XK_Arabic_hamza_above 0x1000654
#define XK_Arabic_hamza_below 0x1000655
#define XK_Arabic_0 0x1000660
#define XK_Arabic_1 0x1000661
#define XK_Arabic_2 0x1000662
#define XK_Arabic_3 0x1000663
#define XK_Arabic_4 0x1000664
#define XK_Arabic_5 0x1000665
#define XK_Arabic_6 0x1000666
#define XK_Arabic_7 0x1000667
#define XK_Arabic_8 0x1000668
#define XK_Arabic_9 0x1000669
#define XK_Arabic_percent 0x100066a
#define XK_Arabic_superscript_alef 0x1000670
#define XK_Arabic_tteh 0x1000679
#define XK_Arabic_peh 0x100067e
#define XK_Arabic_tcheh 0x1000686
#define XK_Arabic_ddal 0x1000688
#define XK_Arabic_rreh 0x1000691
#define XK_Arabic_jeh 0x1000698
#define XK_Arabic_veh 0x10006a4
#define XK_Arabic_kenen 0x10006a9
#define XK_Arabic_gaf 0x10006af
#define XK_Arabic_noon_ghunna 0x10006ba
#define XK_Arabic_heh_doachashmee 0x10006be
#define XK_Arabic_heh_goal 0x10006c1
#define XK_Arabic_farsi_yeh 0x10006cc
#define XK_Farsi_yeh 0x10006cc
#define XK_Arabic_yeh_baree 0x10006d2
#define XK_Arabic_fullstop 0x10006d4
#define XK_Farsi_0 0x10006f0
#define XK_Farsi_1 0x10006f1
#define XK_Farsi_2 0x10006f2
#define XK_Farsi_3 0x10006f3
#define XK_Farsi_4 0x10006f4
#define XK_Farsi_5 0x10006f5
#define XK_Farsi_6 0x10006f6
#define XK_Farsi_7 0x10006f7
#define XK_Farsi_8 0x10006f8
#define XK_Farsi_9 0x10006f9
#define XK_Georgian_an 0x10010d0
#define XK_Georgian_ban 0x10010d1
#define XK_Georgian_gan 0x10010d2
#define XK_Georgian_don 0x10010d3
#define XK_Georgian_en 0x10010d4
#define XK_Georgian_vin 0x10010d5
#define XK_Georgian_zen 0x10010d6
#define XK_Georgian_tan 0x10010d7

```

```

#define XK_Georgian_in 0x10010d8
#define XK_Georgian_kan 0x10010d9
#define XK_Georgian_las 0x10010da
#define XK_Georgian_man 0x10010db
#define XK_Georgian_nar 0x10010dc
#define XK_Georgian_on 0x10010dd
#define XK_Georgian_par 0x10010de
#define XK_Georgian_zhar 0x10010df
#define XK_Georgian_rae 0x10010e0
#define XK_Georgian_san 0x10010e1
#define XK_Georgian_tar 0x10010e2
#define XK_Georgian_un 0x10010e3
#define XK_Georgian_phar 0x10010e4
#define XK_Georgian_khar 0x10010e5
#define XK_Georgian_ghan 0x10010e6
#define XK_Georgian_qar 0x10010e7
#define XK_Georgian_shin 0x10010e8
#define XK_Georgian_chin 0x10010e9
#define XK_Georgian_can 0x10010ea
#define XK_Georgian_jil 0x10010eb
#define XK_Georgian_cil 0x10010ec
#define XK_Georgian_char 0x10010ed
#define XK_Georgian_xan 0x10010ee
#define XK_Georgian_jhan 0x10010ef
#define XK_Georgian_hae 0x10010f0
#define XK_Georgian_he 0x10010f1
#define XK_Georgian_hie 0x10010f2
#define XK_Georgian_we 0x10010f3
#define XK_Georgian_har 0x10010f4
#define XK_Georgian_hoe 0x10010f5
#define XK_Georgian_fi 0x10010f6
#define XK_Babovedot 0x1001e02
#define XK_babovedot 0x1001e03
#define XK_Dabovedot 0x1001e0a
#define XK_dabovedot 0x1001e0b
#define XK_Fabovedot 0x1001e1e
#define XK_fabovedot 0x1001e1f
#define XK_Lbelowdot 0x1001e36
#define XK_lbelowdot 0x1001e37
#define XK_Mabovedot 0x1001e40
#define XK_mabovedot 0x1001e41
#define XK_Pabovedot 0x1001e56
#define XK_pabovedot 0x1001e57
#define XK_Sabovedot 0x1001e60
#define XK_sabovedot 0x1001e61
#define XK_Tabovedot 0x1001e6a
#define XK_tabovedot 0x1001e6b
#define XK_Wgrave 0x1001e80
#define XK_wgrave 0x1001e81
#define XK_Wacute 0x1001e82
#define XK_wacute 0x1001e83
#define XK_Wdiaeresis 0x1001e84
#define XK_wdiaeresis 0x1001e85
#define XK_Xabovedot 0x1001e8a
#define XK_xabovedot 0x1001e8b
#define XK_Abelowdot 0x1001ea0
#define XK_abelowdot 0x1001ea1
#define XK_Ahook 0x1001ea2
#define XK_ahook 0x1001ea3
#define XK_Acircumflexacute 0x1001ea4
#define XK_acircumflexacute 0x1001ea5
#define XK_Acircumflexgrave 0x1001ea6
#define XK_acircumflexgrave 0x1001ea7
#define XK_Acircumflexhook 0x1001ea8
#define XK_acircumflexhook 0x1001ea9
#define XK_Acircumflextilde 0x1001eaa

```

```

#define XK_acircumflextilde      0x1001eab
#define XK_Acircumflexbelowdot  0x1001eac
#define XK_acircumflexbelowdot  0x1001ead
#define XK_Abreveacute          0x1001eae
#define XK_abreveacute          0x1001eaf
#define XK_Abrevegrave          0x1001eb0
#define XK_abrevegrave          0x1001eb1
#define XK_Abrevehook           0x1001eb2
#define XK_abrevehook           0x1001eb3
#define XK_Abrevetilde           0x1001eb4
#define XK_abrevetilde           0x1001eb5
#define XK_Abrevebelowdot       0x1001eb6
#define XK_abrevebelowdot       0x1001eb7
#define XK_Ebelowdot            0x1001eb8
#define XK_ebelowdot            0x1001eb9
#define XK_Ehook                0x1001eba
#define XK_ehook                0x1001ebb
#define XK_Etilde               0x1001ebc
#define XK_etilde               0x1001ebd
#define XK_Ecircumflexacute      0x1001ebe
#define XK_ecircumflexacute      0x1001ebf
#define XK_Ecircumflexgrave      0x1001ec0
#define XK_ecircumflexgrave      0x1001ec1
#define XK_Ecircumflexhook       0x1001ec2
#define XK_ecircumflexhook       0x1001ec3
#define XK_Ecircumflextilde      0x1001ec4
#define XK_ecircumflextilde      0x1001ec5
#define XK_Ecircumflexbelowdot    0x1001ec6
#define XK_ecircumflexbelowdot    0x1001ec7
#define XK_Ihook                0x1001ec8
#define XK_ihook                0x1001ec9
#define XK_Ibelowdot             0x1001eca
#define XK_ibelowdot             0x1001ecb
#define XK_Obelowdot             0x1001ecc
#define XK_obelowdot             0x1001ecd
#define XK_Ohook                0x1001ece
#define XK_ohook                0x1001ecf
#define XK_Ocircumflexacute      0x1001ed0
#define XK_ocircumflexacute      0x1001ed1
#define XK_Ocircumflexgrave      0x1001ed2
#define XK_ocircumflexgrave      0x1001ed3
#define XK_Ocircumflexhook       0x1001ed4
#define XK_ocircumflexhook       0x1001ed5
#define XK_Ocircumflextilde      0x1001ed6
#define XK_ocircumflextilde      0x1001ed7
#define XK_Ocircumflexbelowdot    0x1001ed8
#define XK_ocircumflexbelowdot    0x1001ed9
#define XK_Ohornacute            0x1001eda
#define XK_ohornacute            0x1001edb
#define XK_Ohorngrave            0x1001edc
#define XK_ohorngrave            0x1001edd
#define XK_Ohornhook             0x1001ede
#define XK_ohornhook             0x1001edf
#define XK_Ohorntilde            0x1001ee0
#define XK_ohorntilde            0x1001ee1
#define XK_Ohornbelowdot         0x1001ee2
#define XK_ohornbelowdot         0x1001ee3
#define XK_Ubelowdot             0x1001ee4
#define XK_ubelowdot             0x1001ee5
#define XK_Uhook                0x1001ee6
#define XK_uhook                0x1001ee7
#define XK_Uhornacute            0x1001ee8
#define XK_uhornacute            0x1001ee9
#define XK_Uhorngrave            0x1001eea
#define XK_uhorngrave            0x1001eeb
#define XK_Uhornhook             0x1001eec

```

```

#define XK_uhornhook      0x1001eed
#define XK_Uhorntilde     0x1001eee
#define XK_uhorntilde     0x1001eef
#define XK_Uhornbelowdot  0x1001ef0
#define XK_uhornbelowdot  0x1001ef1
#define XK_Ygrave         0x1001ef2
#define XK_ygrave         0x1001ef3
#define XK_Ybelowdot      0x1001ef4
#define XK_ybelowdot      0x1001ef5
#define XK_Yhook          0x1001ef6
#define XK_yhook          0x1001ef7
#define XK_Ytilde         0x1001ef8
#define XK_ytilde         0x1001ef9
#define XK_zerosuperior   0x1002070
#define XK_foursuperior   0x1002074
#define XK_fivesuperior   0x1002075
#define XK_sixsuperior    0x1002076
#define XK_sevensuperior  0x1002077
#define XK_eightsuperior  0x1002078
#define XK_ninesuperior   0x1002079
#define XK_zerosubscript  0x1002080
#define XK_onesubscript   0x1002081
#define XK_twosubscript   0x1002082
#define XK_threesubscript 0x1002083
#define XK_foursubscript  0x1002084
#define XK_fivesubscript  0x1002085
#define XK_sixsubscript   0x1002086
#define XK_sevensubscript 0x1002087
#define XK_eightsubscript 0x1002088
#define XK_ninesubscript  0x1002089
#define XK_EcuSign        0x10020a0
#define XK_ColonSign      0x10020a1
#define XK_CruzeiroSign   0x10020a2
#define XK_FFrancSign     0x10020a3
#define XK_LiraSign       0x10020a4
#define XK_MillSign       0x10020a5
#define XK_NairaSign      0x10020a6
#define XK_PesetaSign     0x10020a7
#define XK_RupeeSign      0x10020a8
#define XK_WonSign        0x10020a9
#define XK_NewSheqelSign  0x10020aa
#define XK_DongSign       0x10020ab
#define XK_partdifferential 0x1002202
#define XK_emptyset       0x1002205
#define XK_elementof      0x1002208
#define XK_notelementof  0x1002209
#define XK_containsas     0x100220B
#define XK_squareroot     0x100221A
#define XK_cuberoot       0x100221B
#define XK_fourthroot     0x100221C
#define XK_dintegral      0x100222C
#define XK_tintegral      0x100222D
#define XK_because        0x1002235
#define XK_notapproxeq    0x1002247
#define XK_approxeq       0x1002248
#define XK_notidentical   0x1002262
#define XK_stricteq       0x1002263
#define XK_braille_blank  0x1002800
#define XK_braille_dots_1 0x1002801
#define XK_braille_dots_2 0x1002802
#define XK_braille_dots_12 0x1002803
#define XK_braille_dots_3 0x1002804
#define XK_braille_dots_13 0x1002805
#define XK_braille_dots_23 0x1002806
#define XK_braille_dots_123 0x1002807
#define XK_braille_dots_4 0x1002808

```



```

#define XK_braille_dots_14      0x1002809
#define XK_braille_dots_24      0x100280a
#define XK_braille_dots_124     0x100280b
#define XK_braille_dots_34      0x100280c
#define XK_braille_dots_134     0x100280d
#define XK_braille_dots_234     0x100280e
#define XK_braille_dots_1234    0x100280f
#define XK_braille_dots_5       0x1002810
#define XK_braille_dots_15      0x1002811
#define XK_braille_dots_25      0x1002812
#define XK_braille_dots_125     0x1002813
#define XK_braille_dots_35      0x1002814
#define XK_braille_dots_135     0x1002815
#define XK_braille_dots_235     0x1002816
#define XK_braille_dots_1235    0x1002817
#define XK_braille_dots_45      0x1002818
#define XK_braille_dots_145     0x1002819
#define XK_braille_dots_245     0x100281a
#define XK_braille_dots_1245    0x100281b
#define XK_braille_dots_345     0x100281c
#define XK_braille_dots_1345    0x100281d
#define XK_braille_dots_2345    0x100281e
#define XK_braille_dots_12345   0x100281f
#define XK_braille_dots_6       0x1002820
#define XK_braille_dots_16      0x1002821
#define XK_braille_dots_26      0x1002822
#define XK_braille_dots_126     0x1002823
#define XK_braille_dots_36      0x1002824
#define XK_braille_dots_136     0x1002825
#define XK_braille_dots_236     0x1002826
#define XK_braille_dots_1236    0x1002827
#define XK_braille_dots_46      0x1002828
#define XK_braille_dots_146     0x1002829
#define XK_braille_dots_246     0x100282a
#define XK_braille_dots_1246    0x100282b
#define XK_braille_dots_346     0x100282c
#define XK_braille_dots_1346    0x100282d
#define XK_braille_dots_2346    0x100282e
#define XK_braille_dots_12346   0x100282f
#define XK_braille_dots_56      0x1002830
#define XK_braille_dots_156     0x1002831
#define XK_braille_dots_256     0x1002832
#define XK_braille_dots_1256    0x1002833
#define XK_braille_dots_356     0x1002834
#define XK_braille_dots_1356    0x1002835
#define XK_braille_dots_2356    0x1002836
#define XK_braille_dots_12356   0x1002837
#define XK_braille_dots_456     0x1002838
#define XK_braille_dots_1456    0x1002839
#define XK_braille_dots_2456    0x100283a
#define XK_braille_dots_12456   0x100283b
#define XK_braille_dots_3456    0x100283c
#define XK_braille_dots_13456   0x100283d
#define XK_braille_dots_23456   0x100283e
#define XK_braille_dots_123456  0x100283f
#define XK_braille_dots_7       0x1002840
#define XK_braille_dots_17      0x1002841
#define XK_braille_dots_27      0x1002842
#define XK_braille_dots_127     0x1002843
#define XK_braille_dots_37      0x1002844
#define XK_braille_dots_137     0x1002845
#define XK_braille_dots_237     0x1002846
#define XK_braille_dots_1237    0x1002847
#define XK_braille_dots_47      0x1002848
#define XK_braille_dots_147     0x1002849
#define XK_braille_dots_247     0x100284a

```



```

#define XK_braille_dots_1247 0x100284b
#define XK_braille_dots_347 0x100284c
#define XK_braille_dots_1347 0x100284d
#define XK_braille_dots_2347 0x100284e
#define XK_braille_dots_12347 0x100284f
#define XK_braille_dots_57 0x1002850
#define XK_braille_dots_157 0x1002851
#define XK_braille_dots_257 0x1002852
#define XK_braille_dots_1257 0x1002853
#define XK_braille_dots_357 0x1002854
#define XK_braille_dots_1357 0x1002855
#define XK_braille_dots_2357 0x1002856
#define XK_braille_dots_12357 0x1002857
#define XK_braille_dots_457 0x1002858
#define XK_braille_dots_1457 0x1002859
#define XK_braille_dots_2457 0x100285a
#define XK_braille_dots_12457 0x100285b
#define XK_braille_dots_3457 0x100285c
#define XK_braille_dots_13457 0x100285d
#define XK_braille_dots_23457 0x100285e
#define XK_braille_dots_123457 0x100285f
#define XK_braille_dots_67 0x1002860
#define XK_braille_dots_167 0x1002861
#define XK_braille_dots_267 0x1002862
#define XK_braille_dots_1267 0x1002863
#define XK_braille_dots_367 0x1002864
#define XK_braille_dots_1367 0x1002865
#define XK_braille_dots_2367 0x1002866
#define XK_braille_dots_12367 0x1002867
#define XK_braille_dots_467 0x1002868
#define XK_braille_dots_1467 0x1002869
#define XK_braille_dots_2467 0x100286a
#define XK_braille_dots_12467 0x100286b
#define XK_braille_dots_3467 0x100286c
#define XK_braille_dots_13467 0x100286d
#define XK_braille_dots_23467 0x100286e
#define XK_braille_dots_123467 0x100286f
#define XK_braille_dots_567 0x1002870
#define XK_braille_dots_1567 0x1002871
#define XK_braille_dots_2567 0x1002872
#define XK_braille_dots_12567 0x1002873
#define XK_braille_dots_3567 0x1002874
#define XK_braille_dots_13567 0x1002875
#define XK_braille_dots_23567 0x1002876
#define XK_braille_dots_123567 0x1002877
#define XK_braille_dots_4567 0x1002878
#define XK_braille_dots_14567 0x1002879
#define XK_braille_dots_24567 0x100287a
#define XK_braille_dots_124567 0x100287b
#define XK_braille_dots_34567 0x100287c
#define XK_braille_dots_134567 0x100287d
#define XK_braille_dots_234567 0x100287e
#define XK_braille_dots_1234567 0x100287f
#define XK_braille_dots_8 0x1002880
#define XK_braille_dots_18 0x1002881
#define XK_braille_dots_28 0x1002882
#define XK_braille_dots_128 0x1002883
#define XK_braille_dots_38 0x1002884
#define XK_braille_dots_138 0x1002885
#define XK_braille_dots_238 0x1002886
#define XK_braille_dots_1238 0x1002887
#define XK_braille_dots_48 0x1002888
#define XK_braille_dots_148 0x1002889
#define XK_braille_dots_248 0x100288a
#define XK_braille_dots_1248 0x100288b
#define XK_braille_dots_348 0x100288c

```

```

#define XK_braille_dots_1348 0x100288d
#define XK_braille_dots_2348 0x100288e
#define XK_braille_dots_12348 0x100288f
#define XK_braille_dots_58 0x1002890
#define XK_braille_dots_158 0x1002891
#define XK_braille_dots_258 0x1002892
#define XK_braille_dots_1258 0x1002893
#define XK_braille_dots_358 0x1002894
#define XK_braille_dots_1358 0x1002895
#define XK_braille_dots_2358 0x1002896
#define XK_braille_dots_12358 0x1002897
#define XK_braille_dots_458 0x1002898
#define XK_braille_dots_1458 0x1002899
#define XK_braille_dots_2458 0x100289a
#define XK_braille_dots_12458 0x100289b
#define XK_braille_dots_3458 0x100289c
#define XK_braille_dots_13458 0x100289d
#define XK_braille_dots_23458 0x100289e
#define XK_braille_dots_123458 0x100289f
#define XK_braille_dots_68 0x10028a0
#define XK_braille_dots_168 0x10028a1
#define XK_braille_dots_268 0x10028a2
#define XK_braille_dots_1268 0x10028a3
#define XK_braille_dots_368 0x10028a4
#define XK_braille_dots_1368 0x10028a5
#define XK_braille_dots_2368 0x10028a6
#define XK_braille_dots_12368 0x10028a7
#define XK_braille_dots_468 0x10028a8
#define XK_braille_dots_1468 0x10028a9
#define XK_braille_dots_2468 0x10028aa
#define XK_braille_dots_12468 0x10028ab
#define XK_braille_dots_3468 0x10028ac
#define XK_braille_dots_13468 0x10028ad
#define XK_braille_dots_23468 0x10028ae
#define XK_braille_dots_123468 0x10028af
#define XK_braille_dots_568 0x10028b0
#define XK_braille_dots_1568 0x10028b1
#define XK_braille_dots_2568 0x10028b2
#define XK_braille_dots_12568 0x10028b3
#define XK_braille_dots_3568 0x10028b4
#define XK_braille_dots_13568 0x10028b5
#define XK_braille_dots_23568 0x10028b6
#define XK_braille_dots_123568 0x10028b7
#define XK_braille_dots_4568 0x10028b8
#define XK_braille_dots_14568 0x10028b9
#define XK_braille_dots_24568 0x10028ba
#define XK_braille_dots_124568 0x10028bb
#define XK_braille_dots_34568 0x10028bc
#define XK_braille_dots_134568 0x10028bd
#define XK_braille_dots_234568 0x10028be
#define XK_braille_dots_1234568 0x10028bf
#define XK_braille_dots_78 0x10028c0
#define XK_braille_dots_178 0x10028c1
#define XK_braille_dots_278 0x10028c2
#define XK_braille_dots_1278 0x10028c3
#define XK_braille_dots_378 0x10028c4
#define XK_braille_dots_1378 0x10028c5
#define XK_braille_dots_2378 0x10028c6
#define XK_braille_dots_12378 0x10028c7
#define XK_braille_dots_478 0x10028c8
#define XK_braille_dots_1478 0x10028c9
#define XK_braille_dots_2478 0x10028ca
#define XK_braille_dots_12478 0x10028cb
#define XK_braille_dots_3478 0x10028cc
#define XK_braille_dots_13478 0x10028cd
#define XK_braille_dots_23478 0x10028ce

```

```

#define XK_braille_dots_123478 0x10028cf
#define XK_braille_dots_578 0x10028d0
#define XK_braille_dots_1578 0x10028d1
#define XK_braille_dots_2578 0x10028d2
#define XK_braille_dots_12578 0x10028d3
#define XK_braille_dots_3578 0x10028d4
#define XK_braille_dots_13578 0x10028d5
#define XK_braille_dots_23578 0x10028d6
#define XK_braille_dots_123578 0x10028d7
#define XK_braille_dots_4578 0x10028d8
#define XK_braille_dots_14578 0x10028d9
#define XK_braille_dots_24578 0x10028da
#define XK_braille_dots_124578 0x10028db
#define XK_braille_dots_34578 0x10028dc
#define XK_braille_dots_134578 0x10028dd
#define XK_braille_dots_234578 0x10028de
#define XK_braille_dots_1234578 0x10028df
#define XK_braille_dots_678 0x10028e0
#define XK_braille_dots_1678 0x10028e1
#define XK_braille_dots_2678 0x10028e2
#define XK_braille_dots_12678 0x10028e3
#define XK_braille_dots_3678 0x10028e4
#define XK_braille_dots_13678 0x10028e5
#define XK_braille_dots_23678 0x10028e6
#define XK_braille_dots_123678 0x10028e7
#define XK_braille_dots_4678 0x10028e8
#define XK_braille_dots_14678 0x10028e9
#define XK_braille_dots_24678 0x10028ea
#define XK_braille_dots_124678 0x10028eb
#define XK_braille_dots_34678 0x10028ec
#define XK_braille_dots_134678 0x10028ed
#define XK_braille_dots_234678 0x10028ee
#define XK_braille_dots_1234678 0x10028ef
#define XK_braille_dots_5678 0x10028f0
#define XK_braille_dots_15678 0x10028f1
#define XK_braille_dots_25678 0x10028f2
#define XK_braille_dots_125678 0x10028f3
#define XK_braille_dots_35678 0x10028f4
#define XK_braille_dots_135678 0x10028f5
#define XK_braille_dots_235678 0x10028f6
#define XK_braille_dots_1235678 0x10028f7
#define XK_braille_dots_45678 0x10028f8
#define XK_braille_dots_145678 0x10028f9
#define XK_braille_dots_245678 0x10028fa
#define XK_braille_dots_1245678 0x10028fb
#define XK_braille_dots_345678 0x10028fc
#define XK_braille_dots_1345678 0x10028fd
#define XK_braille_dots_2345678 0x10028fe
#define XK_braille_dots_12345678 0x10028ff
#define XK_OE 0x13bc
#define XK_oe 0x13bd
#define XK_Ydiaeresis 0x13be
#define XK_EuroSign 0x20ac
#define XK_ISO_Lock 0xfe01
#define XK_ISO_Level2_Latch 0xfe02
#define XK_ISO_Level3_Shift 0xfe03
#define XK_ISO_Level3_Latch 0xfe04
#define XK_ISO_Level3_Lock 0xfe05
#define XK_ISO_Group_Latch 0xfe06
#define XK_ISO_Group_Lock 0xfe07
#define XK_ISO_Next_Group 0xfe08
#define XK_ISO_Next_Group_Lock 0xfe09
#define XK_ISO_Prev_Group 0xfe0a
#define XK_ISO_Prev_Group_Lock 0xfe0b
#define XK_ISO_First_Group 0xfe0c
#define XK_ISO_First_Group_Lock 0xfe0d

```

```

#define XK_ISO_Last_Group          0xfe0e
#define XK_ISO_Last_Group_Lock    0xfe0f
#define XK_ISO_Level5_Shift        0xfe11
#define XK_ISO_Level5_Latch        0xfe12
#define XK_ISO_Level5_Lock         0xfe13
#define XK_ISO_Left_Tab            0xfe20
#define XK_ISO_Move_Line_Up        0xfe21
#define XK_ISO_Move_Line_Down      0xfe22
#define XK_ISO_Partial_Line_Up     0xfe23
#define XK_ISO_Partial_Line_Down   0xfe24
#define XK_ISO_Partial_Space_Left  0xfe25
#define XK_ISO_Partial_Space_Right 0xfe26
#define XK_ISO_Set_Margin_Left     0xfe27
#define XK_ISO_Set_Margin_Right    0xfe28
#define XK_ISO_Release_Margin_Left 0xfe29
#define XK_ISO_Release_Margin_Right 0xfe2a
#define XK_ISO_Release_Both_Margins 0xfe2b
#define XK_ISO_Fast_Cursor_Left    0xfe2c
#define XK_ISO_Fast_Cursor_Right   0xfe2d
#define XK_ISO_Fast_Cursor_Up      0xfe2e
#define XK_ISO_Fast_Cursor_Down    0xfe2f
#define XK_ISO_Continuous_Underline 0xfe30
#define XK_ISO_Discontinuous_Underline 0xfe31
#define XK_ISO_Emphasize           0xfe32
#define XK_ISO_Center_Object        0xfe33
#define XK_ISO_Enter               0xfe34
#define XK_dead_grave               0xfe50
#define XK_dead_acute               0xfe51
#define XK_dead_circumflex          0xfe52
#define XK_dead_tilde               0xfe53
#define XK_dead_macron              0xfe54
#define XK_dead_breve               0xfe55
#define XK_dead_abovedot            0xfe56
#define XK_dead_diaeresis           0xfe57
#define XK_dead_abovering           0xfe58
#define XK_dead_doubleacute         0xfe59
#define XK_dead_caron               0xfe5a
#define XK_dead_cedilla             0xfe5b
#define XK_dead_ogonek              0xfe5c
#define XK_dead_iota                0xfe5d
#define XK_dead_voiced_sound        0xfe5e
#define XK_dead_semivoiced_sound    0xfe5f
#define XK_dead_belowdot            0xfe60
#define XK_dead_hook                0xfe61
#define XK_dead_horn                0xfe62
#define XK_dead_stroke              0xfe63
#define XK_dead_abovecomma          0xfe64
#define XK_dead_psili               0xfe64
#define XK_dead_abovereverversedcomma 0xfe65
#define XK_dead_dasia              0xfe66
#define XK_dead_belowring           0xfe67
#define XK_dead_belowmacron         0xfe68
#define XK_dead_belowcircumflex     0xfe69
#define XK_dead_belowtilde          0xfe6a
#define XK_dead_belowbreve          0xfe6b
#define XK_dead_belowdiaeresis      0xfe6c
#define XK_AccessX_Enable           0xfe70
#define XK_AccessX_Feedback_Enable 0xfe71
#define XK_RepeatKeys_Enable        0xfe72
#define XK_SlowKeys_Enable          0xfe73
#define XK_BounceKeys_Enable        0xfe74
#define XK_StickyKeys_Enable        0xfe75
#define XK_MouseKeys_Enable         0xfe76
#define XK_MouseKeys_Accel_Enable   0xfe77
#define XK_Overlay1_Enable          0xfe78
#define XK_Overlay2_Enable          0xfe79

```

```

#define XK_AudibleBell_Enable    0xfe7a
#define XK_First_Virtual_Screen  0xfed0
#define XK_Prev_Virtual_Screen   0xfed1
#define XK_Next_Virtual_Screen   0xfed2
#define XK_Last_Virtual_Screen   0xfed4
#define XK_Terminate_Server      0xfed5
#define XK_Pointer_Left 0xfef0
#define XK_Pointer_Right 0xfef1
#define XK_Pointer_Up 0xfef2
#define XK_Pointer_Down 0xfef3
#define XK_Pointer_UpLeft 0xfef4
#define XK_Pointer_UpRight 0xfef5
#define XK_Pointer_DownLeft 0xfef6
#define XK_Pointer_DownRight 0xfef7
#define XK_Pointer_Button_Dflt 0xfef8
#define XK_Pointer_Button1 0xfef9
#define XK_Pointer_Button2 0xfefa
#define XK_Pointer_Button3 0xfefb
#define XK_Pointer_Button4 0xfefc
#define XK_Pointer_Button5 0xfefd
#define XK_Pointer_DblClick_Dflt 0xfefe
#define XK_Pointer_DblClick1 0xfef0
#define XK_Pointer_DblClick2 0xfef1
#define XK_Pointer_DblClick3 0xfef2
#define XK_Pointer_DblClick4 0xfef3
#define XK_Pointer_DblClick5 0xfef4
#define XK_Pointer_Drag_Dflt 0xfef5
#define XK_Pointer_Drag1 0xfef6
#define XK_Pointer_Drag2 0xfef7
#define XK_Pointer_Drag3 0xfef8
#define XK_Pointer_Drag4 0xfef9
#define XK_Pointer_EnableKeys 0xfefa
#define XK_Pointer_Accelerate 0xfefb
#define XK_Pointer_DfltBtnNext 0xfefc
#define XK_Pointer_DfltBtnPrev 0xfefd
#define XK_BackSpace 0xff08
#define XK_Tab 0xff09
#define XK_Linefeed 0xff0a
#define XK_Clear 0xff0b
#define XK_Return 0xff0d
#define XK_Pause 0xff13
#define XK_Scroll_Lock 0xff14
#define XK_Sys_Req 0xff15
#define XK_Escape 0xff1b
#define XK_Multi_key 0xff20
#define XK_Kanji 0xff21
#define XK_Muhenkan 0xff22
#define XK_Henkan 0xff23
#define XK_Henkan_Mode 0xff23
#define XK_Romaji 0xff24
#define XK_Hiragana 0xff25
#define XK_Katakana 0xff26
#define XK_Hiragana_Katakana 0xff27
#define XK_Zenkaku 0xff28
#define XK_Hankaku 0xff29
#define XK_Zenkaku_Hankaku 0xff2a
#define XK_Touroku 0xff2b
#define XK_Massyo 0xff2c
#define XK_Kana_Lock 0xff2d
#define XK_Kana_Shift 0xff2e
#define XK_Eisu_Shift 0xff2f
#define XK_Eisu_toggle 0xff30
#define XK_Hangul 0xff31
#define XK_Hangul_Start 0xff32
#define XK_Hangul_End 0xff33

```

```

#define XK_Hangul_Hanja 0xff34
#define XK_Hangul_Jamo 0xff35
#define XK_Hangul_Romaja 0xff36
#define XK_Codeinput 0xff37
#define XK_Hangul_Codeinput 0xff37
#define XK_Kanji_Bangou 0xff37
#define XK_Hangul_Jeonja 0xff38
#define XK_Hangul_Banja 0xff39
#define XK_Hangul_PreHanja 0xff3a
#define XK_Hangul_PostHanja 0xff3b
#define XK_Hangul_SingleCandidate 0xff3c
#define XK_SingleCandidate 0xff3c
#define XK_Hangul_MultipleCandidate 0xff3d
#define XK_MultipleCandidate 0xff3d
#define XK_Zen_Koho 0xff3d
#define XK_Hangul_PreviousCandidate 0xff3e
#define XK_Mae_Koho 0xff3e
#define XK_PreviousCandidate 0xff3e
#define XK_Hangul_Special 0xff3f
#define XK_Home 0xff50
#define XK_Left 0xff51
#define XK_Up 0xff52
#define XK_Right 0xff53
#define XK_Down 0xff54
#define XK_Page_Up 0xff55
#define XK_Prior 0xff55
#define XK_Next 0xff56
#define XK_Page_Down 0xff56
#define XK_End 0xff57
#define XK_Begin 0xff58
#define XK_Select 0xff60
#define XK_Print 0xff61
#define XK_Execute 0xff62
#define XK_Insert 0xff63
#define XK_Undo 0xff65
#define XK_Redo 0xff66
#define XK_Menu 0xff67
#define XK_Find 0xff68
#define XK_Cancel 0xff69
#define XK_Help 0xff6a
#define XK_Break 0xff6b
#define XK_Arabic_switch 0xff7e
#define XK_Greek_switch 0xff7e
#define XK_Hangul_switch 0xff7e
#define XK_Hebrew_switch 0xff7e
#define XK_ISO_Group_Shift 0xff7e
#define XK_Mode_switch 0xff7e
#define XK_kana_switch 0xff7e
#define XK_script_switch 0xff7e
#define XK_Num_Lock 0xff7f
#define XK_KP_Space 0xff80
#define XK_KP_Tab 0xff89
#define XK_KP_Enter 0xff8d
#define XK_KP_F1 0xff91
#define XK_KP_F2 0xff92
#define XK_KP_F3 0xff93
#define XK_KP_F4 0xff94
#define XK_KP_Home 0xff95
#define XK_KP_Left 0xff96
#define XK_KP_Up 0xff97
#define XK_KP_Right 0xff98
#define XK_KP_Down 0xff99
#define XK_KP_Page_Up 0xff9a
#define XK_KP_Prior 0xff9a
#define XK_KP_Next 0xff9b
#define XK_KP_Page_Down 0xff9b

```

```

#define XK_KP_End          0xff9c
#define XK_KP_Begin        0xff9d
#define XK_KP_Insert       0xff9e
#define XK_KP_Delete       0xff9f
#define XK_KP_Multiply     0xffaa
#define XK_KP_Add          0xffab
#define XK_KP_Separator    0xffac
#define XK_KP_Subtract     0xffad
#define XK_KP_Decimal      0xffae
#define XK_KP_Divide       0xffaf
#define XK_KP_0            0xffb0
#define XK_KP_1            0xffb1
#define XK_KP_2            0xffb2
#define XK_KP_3            0xffb3
#define XK_KP_4            0xffb4
#define XK_KP_5            0xffb5
#define XK_KP_6            0xffb6
#define XK_KP_7            0xffb7
#define XK_KP_8            0xffb8
#define XK_KP_9            0xffb9
#define XK_KP_Equal        0xffbd
#define XK_F1              0xffbe
#define XK_F2              0xffbf
#define XK_F3              0xffc0
#define XK_F4              0xffc1
#define XK_F5              0xffc2
#define XK_F6              0xffc3
#define XK_F7              0xffc4
#define XK_F8              0xffc5
#define XK_F9              0xffc6
#define XK_F10             0xffc7
#define XK_F11             0xffc8
#define XK_L1              0xffc8
#define XK_F12             0xffc9
#define XK_L2              0xffc9
#define XK_F13             0xffca
#define XK_L3              0xffca
#define XK_F14             0xffcb
#define XK_L4              0xffcb
#define XK_F15             0xffcc
#define XK_L5              0xffcc
#define XK_F16             0xffcd
#define XK_L6              0xffcd
#define XK_F17             0xffce
#define XK_L7              0xffce
#define XK_F18             0xffcf
#define XK_L8              0xffcf
#define XK_F19             0xffd0
#define XK_L9              0xffd0
#define XK_F20             0xffd1
#define XK_L10             0xffd1
#define XK_F21             0xffd2
#define XK_R1              0xffd2
#define XK_F22             0xffd3
#define XK_R2              0xffd3
#define XK_F23             0xffd4
#define XK_R3              0xffd4
#define XK_F24             0xffd5
#define XK_R4              0xffd5
#define XK_F25             0xffd6
#define XK_R5              0xffd6
#define XK_F26             0xffd7
#define XK_R6              0xffd7
#define XK_F27             0xffd8
#define XK_R7              0xffd8
#define XK_F28             0xffd9

```



```

#define XK_R8      0xffd9
#define XK_F29     0xffda
#define XK_R9      0xffda
#define XK_F30     0xffdb
#define XK_R10     0xffdb
#define XK_F31     0xffdc
#define XK_R11     0xffdc
#define XK_F32     0xffdd
#define XK_R12     0xffdd
#define XK_F33     0xffde
#define XK_R13     0xffde
#define XK_F34     0xffdf
#define XK_R14     0xffdf
#define XK_F35     0xffe0
#define XK_R15     0xffe0
#define XK_Shift_L  0xffe1
#define XK_Shift_R  0xffe2
#define XK_Control_L 0xffe3
#define XK_Control_R 0xffe4
#define XK_Caps_Lock 0xffe5
#define XK_Shift_Lock 0xffe6
#define XK_Meta_L   0xffe7
#define XK_Meta_R   0xffe8
#define XK_Alt_L    0xffe9
#define XK_Alt_R    0xffea
#define XK_Super_L  0xffeb
#define XK_Super_R  0xffec
#define XK_Hyper_L  0xffed
#define XK_Hyper_R  0xffee
#define XK_braille_dot_1 0xffff1
#define XK_braille_dot_2 0xffff2
#define XK_braille_dot_3 0xffff3
#define XK_braille_dot_4 0xffff4
#define XK_braille_dot_5 0xffff5
#define XK_braille_dot_6 0xffff6
#define XK_braille_dot_7 0xffff7
#define XK_braille_dot_8 0xffff8
#define XK_braille_dot_9 0xffff9
#define XK_braille_dot_10 0xffffa
#define XK_Delete   0xfffff
#define XK_VoidSymbol 0xfffffff /* Void symbol */

```

6.3 Interface Definitions for libX11

The interfaces defined on the following pages are included in libX11 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 6.1 shall behave as described in the referenced base document.

6.4 Interfaces for libSM

Table 6-3 defines the library name and shared object name for the libSM library

Table 6-3 libSM Definition

Library:	libSM
SONAME:	libSM.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[XSM] X11 Session Management

6.4.1 Session Management Functions

6.4.1.1 Interfaces for Session Management Functions

An LSB conforming implementation shall provide the generic functions for Session Management Functions specified in Table 6-4, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-4 libSM - Session Management Functions Function Interfaces

SmFreeProperty [XSM]	SmFreeReasons [XSM]	SmcClientID [XSM]	SmcCloseConnec tion [XSM]
SmcDeletePropert ies [XSM]	SmcGetIceConne ction [XSM]	SmcGetProperty es [XSM]	SmcInteractDone [XSM]
SmcInteractRequ est [XSM]	SmcModifyCallb acks [XSM]	SmcOpenConnec tion [XSM]	SmcProtocolRevi sion [XSM]
SmcProtocolVers ion [XSM]	SmcRelease [XSM]	SmcRequestSave Yourself [XSM]	SmcRequestSave YourselfPhase2 [XSM]
SmcSaveYourself Done [XSM]	SmcSetErrorHan dler [XSM]	SmcSetProperties [XSM]	SmcVendor [XSM]
SmsCleanUp [XSM]	SmsClientHostN ame [XSM]	SmsClientID [XSM]	SmsDie [XSM]
SmsGenerateClie ntID [XSM]	SmsGetIceConne ction [XSM]	SmsInitialize [XSM]	SmsInteract [XSM]
SmsProtocolRevi sion [XSM]	SmsProtocolVers ion [XSM]	SmsRegisterClie ntReply [XSM]	SmsReturnPrope rties [XSM]
SmsSaveComple te [XSM]	SmsSaveYourself [XSM]	SmsSaveYourself Phase2 [XSM]	SmsSetErrorHan dler [XSM]
SmsShutdownCa nelled [XSM]			

6.5 Data Definitions for libSM

This section defines global identifiers and their values that are associated with interfaces contained in libSM. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.5.1 X11/SM/SM.h

```

#define SM_Error 0
#define SmDialogError 0
#define SmInteractStyleNone 0
#define SmProtoMinor 0
#define SmRestartIfRunning 0
#define SmSaveGlobal 0
#define SM_RegisterClient 1
#define SmDialogNormal 1
#define SmInteractStyleErrors 1
#define SmProtoMajor 1
#define SmRestartAnyway 1
#define SmSaveLocal 1
#define SM_ShutdownCancelled 10
#define SM_CloseConnection 11
#define SM_SetProperties 12
#define SM_DeleteProperties 13
#define SM_GetProperties 14
#define SM_PropertiesReply 15
#define SM_SaveYourselfPhase2Request 16
#define SM_SaveYourselfPhase2 17
#define SM_SaveComplete 18
#define SM_RegisterClientReply 2
#define SmInteractStyleAny 2
#define SmRestartImmediately 2
#define SmSaveBoth 2
#define SM_SaveYourself 3
#define SmRestartNever 3
#define SM_SaveYourselfRequest 4
#define SM_InteractRequest 5
#define SM_Interact 6
#define SM_InteractDone 7
#define SM_SaveYourselfDone 8
#define SM_Die 9
#define SmARRAY8 "ARRAY8"
#define SmCARD8 "CARD8"
#define SmCloneCommand "CloneCommand"
#define SmCurrentDirectory "CurrentDirectory"
#define SmDiscardCommand "DiscardCommand"
#define SmEnvironment "Environment"
#define SmLISTOfARRAY8 "LISTOfARRAY8"
#define SmProcessID "ProcessID"
#define SmProgram "Program"
#define SmResignCommand "ResignCommand"
#define SmRestartCommand "RestartCommand"
#define SmRestartStyleHint "RestartStyleHint"
#define SmShutdownCommand "ShutdownCommand"
#define SmUserID "UserID"

```

6.5.2 X11/SM/SMlib.h

```

#define SmcSaveYourselfProcMask (1L << 0)
#define SmsRegisterClientProcMask (1L << 0)
#define SmcDieProcMask (1L << 1)
#define SmsInteractRequestProcMask (1L << 1)
#define SmcSaveCompleteProcMask (1L << 2)
#define SmsInteractDoneProcMask (1L << 2)
#define SmcShutdownCancelledProcMask (1L << 3)
#define SmsSaveYourselfRequestProcMask (1L << 3)
#define SmsSaveYourselfP2RequestProcMask (1L << 4)
#define SmsSaveYourselfDoneProcMask (1L << 5)
#define SmsCloseConnectionProcMask (1L << 6)

```

```

#define SmsSetPropertiesProcMask      (1L << 7)
#define SmsDeletePropertiesProcMask  (1L << 8)
#define SmsGetPropertiesProcMask     (1L << 9)

typedef void *SmPointer;
typedef struct {
    int length;
    SmPointer value;
} SmPropValue;
typedef struct {
    char *name;
    char *type;
    int num_vals;
    SmPropValue *vals;
} SmProp;
struct _SmcConn;
typedef struct _SmcConn *SmcConn;
typedef void (*SmcSaveYourselfProc) (SmcConn, SmPointer, int, int,
int,
int);
typedef void (*SmcDieProc) (SmcConn, SmPointer);
typedef void (*SmcSaveCompleteProc) (SmcConn, SmPointer);
typedef void (*SmcShutdownCancelledProc) (SmcConn, SmPointer);
typedef struct {
    struct {
        SmcSaveYourselfProc callback;
        SmPointer client_data;
    } save_yourself;
    struct {
        SmcDieProc callback;
        SmPointer client_data;
    } die;
    struct {
        SmcSaveCompleteProc callback;
        SmPointer client_data;
    } save_complete;
    struct {
        SmcShutdownCancelledProc callback;
        SmPointer client_data;
    } shutdown_cancelled;
} SmcCallbacks;
struct _SmsConn;
typedef struct _SmsConn *SmsConn;
typedef void (*SmsErrorHandler) (SmsConn, int, int, long unsigned
int, int,
int, SmPointer);
typedef void (*SmcErrorHandler) (SmcConn, int, int, long unsigned
int, int,
int, SmPointer);
typedef int (*SmsRegisterClientProc) (SmsConn, SmPointer, char *);
typedef void (*SmsInteractRequestProc) (SmsConn, SmPointer, int);
typedef void (*SmsInteractDoneProc) (SmsConn, SmPointer, int);
typedef void (*SmsSaveYourselfRequestProc) (SmsConn, SmPointer, int,
int,
int, int, int);
typedef void (*SmsSaveYourselfPhase2RequestProc) (SmsConn,
SmPointer);
typedef void (*SmsSaveYourselfDoneProc) (SmsConn, SmPointer, int);
typedef void (*SmsCloseConnectionProc) (SmsConn, SmPointer, int,
char **);
typedef void (*SmsSetPropertiesProc) (SmsConn, SmPointer, int,
SmProp *);
typedef void (*SmsDeletePropertiesProc) (SmsConn, SmPointer, int,
char **);
typedef void (*SmsGetPropertiesProc) (SmsConn, SmPointer);
typedef struct {

```

```

    struct {
        SmsRegisterClientProc callback;
        SmPointer manager_data;
    } register_client;
    struct {
        SmsInteractRequestProc callback;
        SmPointer manager_data;
    } interact_request;
    struct {
        SmsInteractDoneProc callback;
        SmPointer manager_data;
    } interact_done;
    struct {
        SmsSaveYourselfRequestProc callback;
        SmPointer manager_data;
    } save_yourself_request;
    struct {
        SmsSaveYourselfPhase2RequestProc callback;
        SmPointer manager_data;
    } save_yourself_phase2_request;
    struct {
        SmsSaveYourselfDoneProc callback;
        SmPointer manager_data;
    } save_yourself_done;
    struct {
        SmsCloseConnectionProc callback;
        SmPointer manager_data;
    } close_connection;
    struct {
        SmsSetPropertiesProc callback;
        SmPointer manager_data;
    } set_properties;
    struct {
        SmsDeletePropertiesProc callback;
        SmPointer manager_data;
    } delete_properties;
    struct {
        SmsGetPropertiesProc callback;
        SmPointer manager_data;
    } get_properties;
} SmsCallbacks;
typedef int (*SmsNewClientProc) (SmsConn, SmPointer, long unsigned
int *,
                                SmsCallbacks *, char **);
typedef void (*SmcPropReplyProc) (SmcConn, SmPointer, int, SmProp
* *);
typedef void (*SmcInteractProc) (SmcConn, SmPointer);
typedef void (*SmcSaveYourselfPhase2Proc) (SmcConn, SmPointer);
typedef enum {
    SmcClosedNow = 0,
    SmcClosedASAP = 1,
    SmcConnectionInUse = 2
} SmcCloseStatus;
extern void SmFreeProperty(SmProp *);
extern void SmFreeReasons(int, char **);
extern char *SmcClientID(SmcConn);
extern SmcCloseStatus SmcCloseConnection(SmcConn, int, char **);
extern void SmcDeleteProperties(SmcConn, int, char **);
extern IceConn SmcGetIceConnection(SmcConn);
extern int SmcGetProperties(SmcConn, SmcPropReplyProc, SmPointer);
extern void SmcInteractDone(SmcConn, int);
extern int SmcInteractRequest(SmcConn, int, SmcInteractProc,
SmPointer);
extern void SmcModifyCallbacks(SmcConn, long unsigned int,
SmsCallbacks *);
extern SmcConn SmcOpenConnection(char *, SmPointer, int, int,

```

```

        long unsigned int, SmcCallbacks *, char
*,
        char **, int, char *);
extern int SmcProtocolRevision(SmcConn);
extern int SmcProtocolVersion(SmcConn);
extern char *SmcRelease(SmcConn);
extern void SmcRequestSaveYourself(SmcConn, int, int, int, int,
int);
extern          int          SmcRequestSaveYourselfPhase2(SmcConn,
SmcSaveYourselfPhase2Proc,
                        SmPointer);
extern void SmcSaveYourselfDone(SmcConn, int);
extern SmcErrorHandler SmcSetErrorHandler(SmcErrorHandler);
extern void SmcSetProperties(SmcConn, int, SmProp * *);
extern char *SmcVendor(SmcConn);
extern void SmsCleanUp(SmsConn);
extern char *SmsClientHostName(SmsConn);
extern char *SmsClientID(SmsConn);
extern void SmsDie(SmsConn);
extern char *SmsGenerateClientID(SmsConn);
extern IceConn SmsGetIceConnection(SmsConn);
extern int  SmsInitialize(char *, char *,  SmsNewClientProc,
SmPointer,
                        IceHostBasedAuthProc, int, char *);
extern void SmsInteract(SmsConn);
extern int  SmsProtocolRevision(SmsConn);
extern int  SmsProtocolVersion(SmsConn);
extern int  SmsRegisterClientReply(SmsConn, char *);
extern void SmsReturnProperties(SmsConn, int, SmProp * *);
extern void SmsSaveComplete(SmsConn);
extern void SmsSaveYourself(SmsConn, int, int, int, int);
extern void SmsSaveYourselfPhase2(SmsConn);
extern SmsErrorHandler SmsSetErrorHandler(SmsErrorHandler);
extern void SmsShutdownCancelled(SmsConn);

```

6.5.3 X11/SM/SMproto.h

```

#define sz_smSaveYourselfMsg      16
#define sz_smSaveYourselfRequestMsg      16
#define sz_smCloseConnectionMsg 8
#define sz_smDeletePropertiesMsg      8
#define sz_smDieMsg      8
#define sz_smGetPropertiesMsg      8
#define sz_smInteractDoneMsg      8
#define sz_smInteractMsg      8
#define sz_smInteractRequestMsg 8
#define sz_smPropertiesReplyMsg 8
#define sz_smRegisterClientMsg 8
#define sz_smRegisterClientReplyMsg      8
#define sz_smSaveCompleteMsg      8
#define sz_smSaveYourselfDoneMsg      8
#define sz_smSaveYourselfPhase2Msg      8
#define sz_smSaveYourselfPhase2RequestMsg      8
#define sz_smSetPropertiesMsg      8
#define sz_smShutdownCancelledMsg      8

```

6.6 Interfaces for libICE

Table 6-5 defines the library name and shared object name for the libICE library

Table 6-5 libICE Definition

Library:	libICE

SONAME:	libICE.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

[XICE] X11 Inter-Client Exchange

6.6.1 ICE Functions

6.6.1.1 Interfaces for ICE Functions

An LSB conforming implementation shall provide the generic functions for ICE Functions specified in Table 6-6, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-6 libICE - ICE Functions Function Interfaces

IceAcceptConnection [XICE]	IceAddConnectionWatch [XICE]	IceAllocScratch [XICE]	IceAppLockConn [XICE]
IceAppUnlockConn [XICE]	IceAuthFileName [XICE]	IceCheckShutdownNegotiation [XICE]	IceCloseConnection [XICE]
IceComposeNetworkIdList [XICE]	IceConnectionNumber [XICE]	IceConnectionStatus [XICE]	IceConnectionString [XICE]
IceFlush [XICE]	IceFreeAuthFileEntry [XICE]	IceFreeListenObjects [XICE]	IceGenerateMagicCookie [XICE]
IceGetAuthFileEntry [XICE]	IceGetConnectionContext [XICE]	IceGetInBufSize [XICE]	IceGetListenConnectionNumber [XICE]
IceGetListenConnectionString [XICE]	IceGetOutBufSize [XICE]	IceInitThreads [XICE]	IceLastReceivedSequenceNumber [XICE]
IceLastSentSequenceNumber [XICE]	IceListenForConnections [XICE]	IceListenForWellKnownConnections [XICE]	IceLockAuthFile [XICE]
IceOpenConnection [XICE]	IcePing [XICE]	IceProcessMessages [XICE]	IceProtocolRevision [XICE]
IceProtocolSetup [XICE]	IceProtocolShutdown [XICE]	IceProtocolVersion [XICE]	IceReadAuthFileEntry [XICE]
IceRegisterForProtocolReply [XICE]	IceRegisterForProtocolSetup [XICE]	IceRelease [XICE]	IceRemoveConnectionWatch [XICE]
IceSetErrorHandler [XICE]	IceSetHostBasedAuthProc [XICE]	IceSetIOErrorHandler [XICE]	IceSetPaAuthData [XICE]
IceSetShutdownNegotiation [XICE]	IceSwapping [XICE]	IceUnlockAuthFile [XICE]	IceVendor [XICE]

IceWriteAuthFile Entry [XICE]	_IceRead [LSB]	_IceReadSkip [LSB]	_IceWrite [LSB]

An LSB conforming implementation shall provide the generic deprecated functions for ICE Functions specified in Table 6-7, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 6-7 libICE - ICE Functions Deprecated Function Interfaces

IceAppLockCon n [XICE]	IceAppUnlockCo nn [XICE]	IceInitThreads [XICE]	

6.7 Data Definitions for libICE

This section defines global identifiers and their values that are associated with interfaces contained in libICE. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.7.1 X11/ICE/ICE.h

```
#define ICE_Error 0
#define IceBadMajor 0
#define IceCanContinue 0
#define IceLSBfirst 0
#define IceProtoMinor 0
#define IceBadMinor 0x8000
#define IceBadState 0x8001
#define IceBadLength 0x8002
#define IceBadValue 0x8003
#define ICE_ByteOrder 1
#define IceFatalToProtocol 1
#define IceMSBfirst 1
#define IceNoAuth 1
#define IceProtoMajor 1
#define ICE_PingReply 10
#define ICE_WantToClose 11
#define ICE_NoClose 12
#define ICE_ConnectionSetup 2
#define IceFatalToConnection 2
#define IceNoVersion 2
#define ICE_AuthRequired 3
#define IceSetupFailed 3
#define ICE_AuthReply 4
#define IceAuthRejected 4
```

```

#define ICE_AuthNextPhase      5
#define IceAuthFailed      5
#define ICE_ConnectionReply    6
#define IceProtocolDuplicate    6
#define ICE_ProtocolSetup      7
#define IceMajorOpcodeDuplicate 7
#define ICE_ProtocolReply      8
#define IceUnknownProtocol      8
#define ICE_Ping                9

```

6.7.2 X11/ICE/ICEconn.h

```

struct _XtransConnInfo;
typedef struct {
    char *vendor;
    char *release;
    int version_count;
    IcePoVersionRec *version_recs;
    int auth_count;
    char **auth_names;
    IcePaAuthProc *auth_procs;
    IceIOErrorProc io_error_proc;
} _IcePoProtocol;
typedef struct {
    char *vendor;
    char *release;
    int version_count;
    IcePaVersionRec *version_recs;
    IceProtocolSetupProc protocol_setup_proc;
    IceProtocolActivateProc protocol_activate_proc;
    int auth_count;
    char **auth_names;
    IcePaAuthProc *auth_procs;
    IceHostBasedAuthProc host_based_auth_proc;
    IceIOErrorProc io_error_proc;
} _IcePaProtocol;
typedef struct {
    char *protocol_name;
    _IcePoProtocol *orig_client;
    _IcePaProtocol *accept_client;
} _IceProtocol;
typedef struct {
    int in_use;
    int my_opcode;
    _IceProtocol *protocol;
    IcePointer client_data;
    int accept_flag;
    union {
        IcePaProcessMsgProc accept_client;
        IcePoProcessMsgProc orig_client;
    } process_msg_proc;
} _IceProcessMsgInfo;
typedef struct _IceSavedReplyWait {
    IceReplyWaitInfo *reply_wait;
    int reply_ready;
    _IceSavedReplyWait *next;
} _IceSavedReplyWait;
typedef struct _IcePingWait {
    IcePingReplyProc ping_reply_proc;
    IcePointer client_data;
    _IcePingWait *next;
} _IcePingWait;
typedef struct {
    int auth_active;
    char my_auth_index;

```



```

    IcePointer my_auth_state;
} _IceConnectToYouInfo;
typedef struct {
    int my_opcode;
    int my_auth_count;
    int *my_auth_indices;
    int auth_active;
    char my_auth_index;
    IcePointer my_auth_state;
} _IceProtoSetupToYouInfo;
typedef struct {
    int his_version_index;
    int my_version_index;
    char *his_vendor;
    char *his_release;
    char my_auth_index;
    IcePointer my_auth_state;
    int must_authenticate;
} _IceConnectToMeInfo;
typedef struct {
    int his_opcode;
    int my_opcode;
    int his_version_index;
    int my_version_index;
    char *his_vendor;
    char *his_release;
    char my_auth_index;
    IcePointer my_auth_state;
    int must_authenticate;
} _IceProtoSetupToMeInfo;
struct _IceConn {
    unsigned int io_ok:1;
    unsigned int swap:1;
    unsigned int waiting_for_byteorder:1;
    unsigned int skip_want_to_close:1;
    unsigned int want_to_close:1;
    unsigned int free_asap:1;
    unsigned int unused1:2;
    unsigned int unused2:8;
    IceConnectStatus connection_status;
    unsigned char my_ice_version_index;
    struct _XtransConnInfo *trans_conn;
    long unsigned int send_sequence;
    long unsigned int receive_sequence;
    char *connection_string;
    char *vendor;
    char *release;
    char *inbuf;
    char *inbufptr;
    char *inbufmax;
    char *outbuf;
    char *outbufptr;
    char *outbufmax;
    char *scratch;
    long unsigned int scratch_size;
    int dispatch_level;
    IcePointer context;
    _IceProcessMsgInfo *process_msg_info;
    char his_min_opcode;
    char his_max_opcode;
    unsigned char open_ref_count;
    unsigned char proto_ref_count;
    IceListenObj listen_obj;
    _IceSavedReplyWait *saved_reply_waits;
    _IcePingWait *ping_waits;
    _IceConnectToYouInfo *connect_to_you;

```

```

    _IceProtoSetupToYouInfo *protosetup_to_you;
    _IceConnectToMeInfo *connect_to_me;
    _IceProtoSetupToMeInfo *protosetup_to_me;
};

```

6.7.3 X11/ICE/ICElib.h

```

typedef void *IcePointer;
typedef struct _IceConn *IceConn;
typedef void (*IcePaProcessMsgProc) (IceConn, IcePointer, int,
                                     long unsigned int, int);

typedef struct {
    int major_version;
    int minor_version;
    IcePaProcessMsgProc process_msg_proc;
} IcePaVersionRec;
typedef struct {
    long unsigned int sequence_of_request;
    int major_opcode_of_request;
    int minor_opcode_of_request;
    IcePointer reply;
} IceReplyWaitInfo;
typedef void (*IcePoProcessMsgProc) (IceConn, IcePointer, int,
                                     long unsigned int, int,
                                     IceReplyWaitInfo *, int *);

typedef struct {
    int major_version;
    int minor_version;
    IcePoProcessMsgProc process_msg_proc;
} IcePoVersionRec;
typedef IcePoAuthStatus (*IcePoAuthProc) (IceConn, IcePointer *, int,
int,
                                     int, IcePointer, int *,
                                     IcePointer *, char **);
typedef IcePaAuthStatus (*IcePaAuthProc) (IceConn, IcePointer *, int,
int,
                                     IcePointer, int *, IcePointer *,
                                     char **);

typedef enum {
    IceAcceptSuccess = 0,
    IceAcceptFailure = 1,
    IceAcceptBadMalloc = 2
} IceAcceptStatus;
struct _IceListenObj;
typedef struct _IceListenObj *IceListenObj;
typedef enum {
    IceConnectPending = 0,
    IceConnectAccepted = 1,
    IceConnectRejected = 2,
    IceConnectIOError = 3
} IceConnectStatus;
typedef void (*IceIOErrorProc) (IceConn);
typedef int (*IceProtocolSetupProc) (IceConn, int, int, char *,
char *,
                                     IcePointer *, char **);
typedef void (*IceProtocolActivateProc) (IceConn, IcePointer);
typedef int (*IceHostBasedAuthProc) (char *);
typedef void (*IcePingReplyProc) (IceConn, IcePointer);
typedef enum {
    IceProcessMessagesSuccess = 0,
    IceProcessMessagesIOError = 1,
    IceProcessMessagesConnectionClosed = 2
} IceProcessMessagesStatus;
typedef enum {
    IceClosedNow = 0,

```

```

IceClosedASAP = 1,
IceConnectionInUse = 2,
IceStartedShutdownNegotiation = 3
} IceCloseStatus;
typedef enum {
IceProtocolSetupSuccess = 0,
IceProtocolSetupFailure = 1,
IceProtocolSetupIOError = 2,
IceProtocolAlreadyActive = 3
} IceProtocolSetupStatus;
typedef void (*IceWatchProc) (IceConn, IcePointer, int, IcePointer
*);
typedef void (*IceIOErrorHandler) (IceConn);
typedef void (*IceErrorHandler) (IceConn, int, int, long unsigned
int, int,
int, IcePointer);
typedef enum {
IcePoAuthHaveReply,
IcePoAuthRejected,
IcePoAuthFailed,
IcePoAuthDoneCleanup
} IcePoAuthStatus;
typedef enum {
IcePaAuthContinue,
IcePaAuthAccepted,
IcePaAuthRejected,
IcePaAuthFailed
} IcePaAuthStatus;
extern IceConn IceAcceptConnection(IceListenObj, IceAcceptStatus
*);
extern int IceAddConnectionWatch(IceWatchProc, IcePointer);
extern char *IceAllocScratch(IceConn, long unsigned int);
extern void IceAppLockConn(IceConn);
extern void IceAppUnlockConn(IceConn);
extern int IceCheckShutdownNegotiation(IceConn);
extern IceCloseStatus IceCloseConnection(IceConn);
extern char *IceComposeNetworkIdList(int, IceListenObj *);
extern int IceConnectionNumber(IceConn);
extern IceConnectStatus IceConnectionStatus(IceConn);
extern char *IceConnectionString(IceConn);
extern int IceFlush(IceConn);
extern void IceFreeListenObjs(int, IceListenObj *);
extern IcePointer IceGetConnectionContext(IceConn);
extern int IceGetInBufSize(IceConn);
extern int IceGetListenConnectionNumber(IceListenObj);
extern char *IceGetListenConnectionString(IceListenObj);
extern int IceGetOutBufSize(IceConn);
extern int IceInitThreads(void);
extern long unsigned int IceLastReceivedSequenceNumber(IceConn);
extern long unsigned int IceLastSentSequenceNumber(IceConn);
extern int IceListenForConnections(int *, IceListenObj * *, int,
char *);
extern int IceListenForWellKnownConnections(char *, int *,
IceListenObj * *, int, char *);
extern IceConn IceOpenConnection(char *, IcePointer, int, int, int,
char *);
extern int IcePing(IceConn, IcePingReplyProc, IcePointer);
extern IceProcessMessagesStatus IceProcessMessages(IceConn,
IceReplyWaitInfo *,
int *);
extern int IceProtocolRevision(IceConn);
extern IceProtocolSetupStatus IceProtocolSetup(IceConn, int,
IcePointer,
int, int *, int *, char **,
char **, int, char *);
extern int IceProtocolShutdown(IceConn, int);

```

```

extern int IceProtocolVersion(IceConn);
extern int IceRegisterForProtocolReply(char *, char *, char *, int,
                                       IcePaVersionRec *, int,
                                       const char **, IcePaAuthProc *,
                                       IceHostBasedAuthProc,
                                       IceProtocolSetupProc,
                                       IceProtocolActivateProc,
                                       IceIOErrorProc);
extern int IceRegisterForProtocolSetup(char *, char *, char *, int,
                                       IcePoVersionRec *, int,
                                       const char **, IcePoAuthProc *,
                                       IceIOErrorProc);

extern char *IceRelease(IceConn);
extern void IceRemoveConnectionWatch(IceWatchProc, IcePointer);
extern IceErrorHandler IceSetErrorHandler(IceErrorHandler);
extern void IceSetHostBasedAuthProc(IceListenObj,
IceHostBasedAuthProc);
extern IceIOErrorHandler IceSetIOErrorHandler(IceIOErrorHandler);
extern void IceSetShutdownNegotiation(IceConn, int);
extern int IceSwapping(IceConn);
extern char *IceVendor(IceConn);

```

6.7.4 X11/ICE/ICEmsg.h

```

#define IceWriteData16(_iceConn, _bytes, _data) IceWriteData
(_iceConn, _bytes, (char *) _data)
#define IceWriteData32(_iceConn, _bytes, _data) IceWriteData
(_iceConn, _bytes, (char *) _data)
#define IceDisposeCompleteMessage(_iceConn, _pData) if ((char
*) _pData < _iceConn->inbuf || (char *) _pData >= _iceConn->inbufmax)
\
    free ((char *) _pData);
#define
IceGetHeaderExtra(_iceConn, _major, _minor, _headerSize, _extra, _msgT
ype, _pMsg, _pData) if ((_iceConn->outbufptr + _headerSize +
(((_extra) << 3)) > _iceConn->outbufmax) IceFlush (_iceConn);\
    _pMsg = (_msgType *) _iceConn->outbufptr;\
    if ((_iceConn->outbufptr + _headerSize + (((_extra) << 3)) <=
_iceConn->outbufmax) _pData = (char *) _pMsg + _headerSize;\
    else _pData = ((void *)0);\
    _pMsg->majorOpcode = _major;\
    _pMsg->minorOpcode = _minor;\
    _pMsg->length = ((_headerSize - SIZEOF (iceMsg)) >> 3) + (_extra);\
    _iceConn->outbufptr += (_headerSize + (((_extra) << 3));\
    _iceConn->send_sequence++
#define
IceGetHeader(_iceConn, _major, _minor, _headerSize, _msgType, _pMsg) if
(((_iceConn->outbufptr + _headerSize) > _iceConn->outbufmax)
IceFlush (_iceConn);\
    _pMsg = (_msgType *) _iceConn->outbufptr;\
    _pMsg->majorOpcode = _major;\
    _pMsg->minorOpcode = _minor;\
    _pMsg->length = (_headerSize - SIZEOF (iceMsg)) >> 3;\
    _iceConn->outbufptr += _headerSize;\
    _iceConn->send_sequence++
#define IceValidIO(_iceConn) _iceConn->io_ok
#define IceReadData(_iceConn, _bytes, _pData) _IceRead (_iceConn,
(unsigned long) (_bytes), (char *) _pData);
#define IceReadSimpleMessage(_iceConn, _msgType, _pMsg) _pMsg =
(_msgType *) (_iceConn->inbuf);
#define IceReadPad(_iceConn, _bytes) { char _dummy[7]; _IceRead
(_iceConn, (unsigned long) (_bytes), _dummy); }
#define
IceErrorHandler(_iceConn, _offendingMajorOpcode, _offendingMinorOpco

```

```

de, _offendingSequenceNum, _severity, _errorClass, _dataLength)
{ iceErrorMsg * _pMsg; \
  IceGetHeader (_iceConn, _offendingMajorOpcode, ICE_Error, SIZEOF
(iceErrorMsg), iceErrorMsg, _pMsg); \
  _pMsg->length += (_dataLength); \
  _pMsg->offendingMinorOpcode = _offendingMinorOpcode; \
  _pMsg->severity = _severity; \
  _pMsg->offendingSequenceNum = _offendingSequenceNum; \
  _pMsg->errorClass = _errorClass; \
}
#define IceSimpleMessage(_iceConn, _major, _minor) { iceMsg
*_pMsg; IceGetHeader (_iceConn, _major, _minor, SIZEOF (iceMsg),
iceMsg, _pMsg) }
#define IceWritePad(_iceConn, _bytes) { if ((_iceConn->outbufptr
+ (_bytes)) > _iceConn->outbufmax) { char _dummy[7]; \
  IceFlush (_iceConn); \
  _IceWrite (_iceConn, (unsigned long) (_bytes), _dummy); \
} else { _iceConn->outbufptr += (_bytes); \
} }
#define IceWriteData(_iceConn, _bytes, _data) { if ((_iceConn-
>outbufptr + (_bytes)) > _iceConn->outbufmax) { IceFlush
(_iceConn); \
  _IceWrite (_iceConn, (unsigned long) (_bytes), _data); \
} else { memcpy (_iceConn->outbufptr, _data, _bytes); \
  _iceConn->outbufptr += (_bytes); \
} }
#define IceSendData(_iceConn, _bytes, _data) { if (_iceConn-
>outbufptr > _iceConn->outbuf) IceFlush (_iceConn); \
  _IceWrite (_iceConn, (unsigned long) (_bytes), _data); \
}
#define
IceReadCompleteMessage(_iceConn, _headerSize, _msgType, _pMsg, _pData)
{ \
  unsigned long _bytes; \
  IceReadMessageHeader (_iceConn, _headerSize, _msgType, _pMsg);
\
  _bytes = (_pMsg->length < 3) - (_headerSize - SIZEOF (iceMsg));
\
  if ((_iceConn->inbufmax - _iceConn->inbufptr) >= _bytes) \
  { \
    _IceRead (_iceConn, _bytes, _iceConn->inbufptr); \
    _pData = _iceConn->inbufptr; \
    _iceConn->inbufptr += _bytes; \
  } \
  else \
  { \
    _pData = (char *) malloc ((unsigned) _bytes); \
    if (_pData) \
      _IceRead (_iceConn, _bytes, _pData); \
    else \
      _IceReadSkip (_iceConn, _bytes); \
  } \
}
#define IceReadData16(_iceConn, _swap, _bytes, _pData) { _IceRead
(_iceConn, (unsigned long) (_bytes), (char *) _pData); }
#define IceReadData32(_iceConn, _swap, _bytes, _pData) { _IceRead
(_iceConn, (unsigned long) (_bytes), (char *) _pData); }
#define IceReadMessageHeader(_iceConn, _headerSize, _msgType, _pMsg)
{ _IceRead (_iceConn, (unsigned long) (_headerSize - SIZEOF
(iceMsg)), _iceConn->inbufptr); \
  _pMsg = (_msgType *) (_iceConn->inbuf); \
  _iceConn->inbufptr += (_headerSize - SIZEOF (iceMsg)); \
}

extern int _IceRead(IceConn, unsigned long int, char *);
extern void _IceReadSkip(IceConn, unsigned long int);

```

```
extern void _IceWrite(IceConn, unsigned long int, char *);
```

6.7.5 X11/ICE/ICEproto.h

```
#define sz_iceAuthNextPhaseMsg 16
#define sz_iceAuthReplyMsg 16
#define sz_iceAuthRequiredMsg 16
#define sz_iceConnectionSetupMsg 16
#define sz_iceErrorMsg 16
#define sz_iceProtocolSetupMsg 16
#define sz_iceByteOrderMsg 8
#define sz_iceConnectionReplyMsg 8
#define sz_iceMsg 8
#define sz_iceNoCloseMsg 8
#define sz_icePingMsg 8
#define sz_icePingReplyMsg 8
#define sz_iceProtocolReplyMsg 8
#define sz_iceWantToCloseMsg 8

typedef iceErrorMsg icePingMsg;
typedef iceErrorMsg icePingReplyMsg;
typedef iceErrorMsg iceWantToCloseMsg;
typedef iceErrorMsg iceNoCloseMsg;
```

6.7.6 X11/ICE/ICEutil.h

```
#define IceAuthLockSuccess 0
#define IceAuthLockError 1
#define IceAuthLockTimeout 2

typedef struct {
    char *protocol_name;
    char *network_id;
    char *auth_name;
    short unsigned int auth_data_length;
    char *auth_data;
} IceAuthDataEntry;
typedef struct {
    char *protocol_name;
    short unsigned int protocol_data_length;
    char *protocol_data;
    char *network_id;
    char *auth_name;
    short unsigned int auth_data_length;
    char *auth_data;
} IceAuthFileEntry;
extern char *IceAuthFileName(void);
extern void IceFreeAuthFileEntry(IceAuthFileEntry *);
extern char *IceGenerateMagicCookie(int);
extern IceAuthFileEntry *IceGetAuthFileEntry(char *, char *, char *);
extern int IceLockAuthFile(char *, int, int, long int);
extern IceAuthFileEntry *IceReadAuthFileEntry(FILE *);
extern void IceSetPaAuthData(int, IceAuthDataEntry *);
extern void IceUnlockAuthFile(char *);
extern int IceWriteAuthFileEntry(FILE *, IceAuthFileEntry *);
```

6.8 Interface Definitions for libICE

The interfaces defined on the following pages are included in libICE and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 6.6 shall behave as described in the referenced base document.

_IceRead

Name

`_IceRead` — read the specified number of bytes from an ICE connection

Synopsis

```
#include <X11/ICE/ICEmsg.h>
int _IceRead(IceConn iceConn, unsigned long int nbytes, char * ptr);
```

Description

The interface `_IceRead()` shall read the specified number of bytes from an Inter-Client Exchange connection.

The parameter *iceConn* shall specify the Inter-Client Exchange connection.

The parameter *nbytes* shall specify the number of bytes to read.

The output parameter *ptr* shall contain the data that was read.

Return Value

`_IceRead()` shall return 0 if it detects that the connection has closed and this was expected.

In the case of a fatal I/O error, `_IceRead()` shall return 1.

See Also

`_IceReadSkip()`, `_IceWrite()`.

_IceReadSkip

Name

`_IceReadSkip` — advance to the end of the message when a bad header is read

Synopsis

```
#include <X11/ICE/ICEmsg.h>
void _IceReadSkip(IceConn iceConn, unsigned long int nbytes);
```

Description

The interface `_IceReadSkip()` shall advance to the end of the message when a header with a bad major or minor opcode is read, so the next message can be processed correctly.

The parameter *iceConn* shall specify the Inter-Client Exchange connection.

The parameter *nbytes* shall specify the number of bytes to skip. If the number of bytes specified is less than 512, the interface shall assume 512 bytes should be skipped.

See Also

`_IceRead()`, `_IceWrite()`.

_IceWrite

Name

`_IceWrite` — write the specified number of bytes of data to an ICE connection

Synopsis

```
#include <X11/ICE/ICEmsg.h>
void _IceWrite(IceConn iceConn, unsigned long int nbytes, char * ptr);
```

Description

The interface `_IceWrite()` shall write the specified number of bytes to an Inter-Client Exchange connection.

The parameter `iceConn` shall specify the Inter-Client Exchange connection.

The parameter `nbytes` shall specify the number of bytes of the specified data to write.

The parameter `ptr` shall contain the data to write.

See Also

`_IceRead()`, `_IceReadSkip()`.

6.9 Interfaces for libXt

Table 6-8 defines the library name and shared object name for the libXt library

Table 6-8 libXt Definition

Library:	libXt
SONAME:	libXt.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

[Xt] X11 Toolkit Intrinsics

6.9.1 X Toolkit

6.9.1.1 Interfaces for X Toolkit

An LSB conforming implementation shall provide the generic functions for X Toolkit specified in Table 6-9, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-9 libXt - X Toolkit Function Interfaces

XtAddActions [Xt]	XtAddCallback [Xt]	XtAddCallbacks [Xt]	XtAddConverter [Xt]
XtAddEventHandler [Xt]	XtAddExposure ToRegion [Xt]	XtAddGrab [Xt]	XtAddInput [Xt]
XtAddRawEvent Handler [Xt]	XtAddSignal [Xt]	XtAddTimeOut [Xt]	XtAddWorkProc [Xt]

XtAllocateGC [Xt]	XtAppAddActionHook [Xt]	XtAppAddActions [Xt]	XtAppAddBlockHook [Xt]
XtAppAddConverter [Xt]	XtAppAddInput [Xt]	XtAppAddSignal [Xt]	XtAppAddTimeout [Xt]
XtAppAddWorkProc [Xt]	XtAppCreateShell [Xt]	XtAppError [Xt]	XtAppErrorMsg [Xt]
XtAppGetErrorDatabase [Xt]	XtAppGetErrorDatabaseText [Xt]	XtAppGetExitFlag [Xt]	XtAppGetSelectionTimeout [Xt]
XtAppInitialize [Xt]	XtAppLock [Xt]	XtAppMainLoop [Xt]	XtAppNextEvent [Xt]
XtAppPeekEvent [Xt]	XtAppPending [Xt]	XtAppProcessEvent [Xt]	XtAppReleaseCacheRefs [Xt]
XtAppSetErrorHandler [Xt]	XtAppSetErrorMsgHandler [Xt]	XtAppSetExitFlag [Xt]	XtAppSetFallbackResources [Xt]
XtAppSetSelectionTimeout [Xt]	XtAppSetTypeConverter [Xt]	XtAppSetWarningHandler [Xt]	XtAppSetWarningMsgHandler [Xt]
XtAppUnlock [Xt]	XtAppWarning [Xt]	XtAppWarningMsg [Xt]	XtAugmentTranslations [Xt]
XtBuildEventMask [Xt]	XtCallAcceptFocus [Xt]	XtCallActionProc [Xt]	XtCallCallbackList [Xt]
XtCallCallbacks [Xt]	XtCallConverter [Xt]	XtCallbackExclusive [Xt]	XtCallbackNone [Xt]
XtCallbackNoneExclusive [Xt]	XtCallbackPopdown [Xt]	XtCallbackReleaseCacheRef [Xt]	XtCallbackReleaseCacheRefList [Xt]
XtCalloc [Xt]	XtCancelSelectionRequest [Xt]	XtChangeManagedSet [Xt]	XtClass [Xt]
XtCloseDisplay [Xt]	XtConfigureWidget [Xt]	XtConvert [Xt]	XtConvertAndStore [Xt]
XtConvertCase [Xt]	XtCreateApplicationContext [Xt]	XtCreateApplicationShell [Xt]	XtCreateManagedWidget [Xt]
XtCreatePopupShell [Xt]	XtCreateSelectionRequest [Xt]	XtCreateWidget [Xt]	XtCreateWindow [Xt]
XtCvtColorToPixel [Xt]	XtCvtIntToBool [Xt]	XtCvtIntToBoolean [Xt]	XtCvtIntToColor [Xt]
XtCvtIntToFloat [Xt]	XtCvtIntToFont [Xt]	XtCvtIntToPixel [Xt]	XtCvtIntToPixmap [Xt]
XtCvtIntToShort [Xt]	XtCvtIntToUnsignedChar [Xt]	XtCvtStringToAcceleratorTable [Xt]	XtCvtStringToAtom [Xt]

XtCvtStringToBo ol [Xt]	XtCvtStringToBo olean [Xt]	XtCvtStringToCo mmandArgArra y [Xt]	XtCvtStringToCu rsor [Xt]
XtCvtStringToDi mension [Xt]	XtCvtStringToDi rectoryString [Xt]	XtCvtStringToDi splay [Xt]	XtCvtStringToFil e [Xt]
XtCvtStringToFl oat [Xt]	XtCvtStringToFo nt [Xt]	XtCvtStringToFo ntSet [Xt]	XtCvtStringToFo ntStruct [Xt]
XtCvtStringToGr avity [Xt]	XtCvtStringToIni tialState [Xt]	XtCvtStringToInt [Xt]	XtCvtStringToPi xel [Xt]
XtCvtStringToRe startStyle [Xt]	XtCvtStringToSh ort [Xt]	XtCvtStringToTr anslationTable [Xt]	XtCvtStringToU nsignedChar [Xt]
XtCvtStringToVi sual [Xt]	XtDatabase [Xt]	XtDestroyApplic ationContext [Xt]	XtDestroyGC [Xt]
XtDestroyWidget [Xt]	XtDirectConvert [Xt]	XtDisownSelecti on [Xt]	XtDispatchEvent [Xt]
XtDispatchEvent ToWidget [Xt]	XtDisplay [Xt]	XtDisplayInitiali ze [Xt]	XtDisplayOfObj ect [Xt]
XtDisplayString ConversionWarn ing [Xt]	XtDisplayToApp licationContext [Xt]	XtError [Xt]	XtErrorMsg [Xt]
XtFindFile [Xt]	XtFree [Xt]	XtGetActionKeys ym [Xt]	XtGetActionList [Xt]
XtGetApplicatio nNameAndClass [Xt]	XtGetApplicatio nResources [Xt]	XtGetClassExten sion [Xt]	XtGetConstraint ResourceList [Xt]
XtGetDisplays [Xt]	XtGetErrorDatab ase [Xt]	XtGetErrorDatab aseText [Xt]	XtGetGC [Xt]
XtGetKeyboardF ocusWidget [Xt]	XtGetKeysymTa ble [Xt]	XtGetMultiClick Time [Xt]	XtGetResourceLi st [Xt]
XtGetSelectionPa rameters [Xt]	XtGetSelectionRe quest [Xt]	XtGetSelectionTi meout [Xt]	XtGetSelectionVa lue [Xt]
XtGetSelectionVa lueIncremental [Xt]	XtGetSelectionVa lues [Xt]	XtGetSelectionVa luesIncremental [Xt]	XtGetSubresourc es [Xt]
XtGetSubvalues [Xt]	XtGetValues [Xt]	XtGrabButton [Xt]	XtGrabKey [Xt]
XtGrabKeyboard [Xt]	XtGrabPointer [Xt]	XtHasCallbacks [Xt]	XtHooksOfDispl ay [Xt]
XtInitialize [Xt]	XtInitializeWidg etClass [Xt]	XtInsertEventHa ndler [Xt]	XtInsertEventTy peHandler [Xt]
XtInsertRawEven tHandler [Xt]	XtInstallAccelera tors [Xt]	XtInstallAllAccel erators [Xt]	XtIsApplicationS hell [Xt]

XtIsComposite [Xt]	XtIsConstraint [Xt]	XtIsManaged [Xt]	XtIsObject [Xt]
XtIsOverrideShell [Xt]	XtIsRealized [Xt]	XtIsRectObj [Xt]	XtIsSensitive [Xt]
XtIsSessionShell [Xt]	XtIsShell [Xt]	XtIsSubclass [Xt]	XtIsTopLevelShell [Xt]
XtIsTransientShell [Xt]	XtIsVendorShell [Xt]	XtIsWMShell [Xt]	XtIsWidget [Xt]
XtKeysymToKeyCodeList [Xt]	XtLastEventProcessed [Xt]	XtLastTimestampProcessed [Xt]	XtMainLoop [Xt]
XtMakeGeometryRequest [Xt]	XtMakeResizeRequest [Xt]	XtMalloc [Xt]	XtManageChild [Xt]
XtManageChildren [Xt]	XtMapWidget [Xt]	XtMenuPopupAction [Xt]	XtMergeArgLists [Xt]
XtMoveWidget [Xt]	XtName [Xt]	XtNameToWidget [Xt]	XtNewString [Xt]
XtNextEvent [Xt]	XtNoticeSignal [Xt]	XtOpenApplication [Xt]	XtOpenDisplay [Xt]
XtOverrideTranslations [Xt]	XtOwnSelection [Xt]	XtOwnSelectionIncremental [Xt]	XtParent [Xt]
XtParseAcceleratorTable [Xt]	XtParseTranslationTable [Xt]	XtPeekEvent [Xt]	XtPending [Xt]
XtPopdown [Xt]	XtPopup [Xt]	XtPopupSpringLoaded [Xt]	XtProcessEvent [Xt]
XtProcessLock [Xt]	XtProcessUnlock [Xt]	XtQueryGeometry [Xt]	XtRealizeWidget [Xt]
XtRealloc [Xt]	XtRegisterCaseConverter [Xt]	XtRegisterDrawable [Xt]	XtRegisterExtensionSelector [Xt]
XtRegisterGrabAction [Xt]	XtReleaseGC [Xt]	XtReleasePropertyAtom [Xt]	XtRemoveActionHook [Xt]
XtRemoveAllCallbacks [Xt]	XtRemoveBlockHook [Xt]	XtRemoveCallback [Xt]	XtRemoveCallbacks [Xt]
XtRemoveEventHandler [Xt]	XtRemoveEventTypeHandler [Xt]	XtRemoveGrab [Xt]	XtRemoveInput [Xt]
XtRemoveRawEventHandler [Xt]	XtRemoveSignal [Xt]	XtRemoveTimeout [Xt]	XtRemoveWorkProc [Xt]
XtReservePropertyAtom [Xt]	XtResizeWidget [Xt]	XtResizeWindow [Xt]	XtResolvePathname [Xt]
XtScreen [Xt]	XtScreenDatabase [Xt]	XtScreenOfObject [Xt]	XtSendSelectionRequest [Xt]

XtSessionGetToken [Xt]	XtSessionReturnToken [Xt]	XtSetErrorHandler [Xt]	XtSetErrorMsgHandler [Xt]
XtSetEventDispatcher [Xt]	XtSetKeyTranslator [Xt]	XtSetKeyboardFocus [Xt]	XtSetLanguageProc [Xt]
XtSetMappedWhenManaged [Xt]	XtSetMultiClickTime [Xt]	XtSetSelectionParameters [Xt]	XtSetSelectionTimeout [Xt]
XtSetSensitive [Xt]	XtSetSubvalues [Xt]	XtSetTypeConverter [Xt]	XtSetValues [Xt]
XtSetWMColormapWindows [Xt]	XtSetWarningHandler [Xt]	XtSetWarningMsgHandler [Xt]	XtStringConversionWarning [Xt]
XtSuperclass [Xt]	XtToolkitInitialize [Xt]	XtToolkitThreadInitialize [Xt]	XtTranslateCoords [Xt]
XtTranslateKey [Xt]	XtTranslateKeyCode [Xt]	XtUngrabButton [Xt]	XtUngrabKey [Xt]
XtUngrabKeyboard [Xt]	XtUngrabPointer [Xt]	XtUninstallTranslations [Xt]	XtUnmanageChild [Xt]
XtUnmanageChildren [Xt]	XtUnmapWidget [Xt]	XtUnrealizeWidget [Xt]	XtUnregisterDrawable [Xt]
XtVaAppCreateShell [Xt]	XtVaAppInitialize [Xt]	XtVaCreateArgsList [Xt]	XtVaCreateManagedWidget [Xt]
XtVaCreatePopupShell [Xt]	XtVaCreateWidget [Xt]	XtVaGetApplicationResources [Xt]	XtVaGetSubresources [Xt]
XtVaGetSubvalues [Xt]	XtVaGetValues [Xt]	XtVaOpenApplication [Xt]	XtVaSetSubvalues [Xt]
XtVaSetValues [Xt]	XtWarning [Xt]	XtWarningMsg [Xt]	XtWidgetToApplicationContext [Xt]
XtWindow [Xt]	XtWindowOfObject [Xt]	XtWindowToWidget [Xt]	_XtCheckSubclassFlag [Xt]
_XtCopyFromArg [LSB]	_XtInherit [LSB]	_XtIsSubclassOf [LSB]	

An LSB conforming implementation shall provide the generic data interfaces for X Toolkit specified in Table 6-10, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-10 libXt - X Toolkit Data Interfaces

XtCXtToolkitError [Xt]	XtShellStrings [Xt]	XtStrings [Xt]	_XtInheritTranslations [LSB]
applicationShellClassRec [Xt]	applicationShellWidgetClass [Xt]	colorConvertArgs [Xt]	compositeClassRec [Xt]
compositeWidgetClass [Xt]	constraintClassRec [Xt]	constraintWidgetClass [Xt]	coreWidgetClass [Xt]

objectClass [Xt]	objectClassRec [Xt]	overrideShellClassRec [Xt]	overrideShellWidgetClass [Xt]
rectObjClass [Xt]	rectObjClassRec [Xt]	sessionShellClassRec [Xt]	sessionShellWidgetClass [Xt]
shellClassRec [Xt]	shellWidgetClass [Xt]	topLevelShellClassRec [Xt]	topLevelShellWidgetClass [Xt]
transientShellClassRec [Xt]	transientShellWidgetClass [Xt]	widgetClass [Xt]	widgetClassRec [Xt]
wmShellClassRec [Xt]	wmShellWidgetClass [Xt]		

6.10 Data Definitions for libXt

This section defines global identifiers and their values that are associated with interfaces contained in libXt. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.10.1 X11/Composite.h

```
#define _XtComposite_h 1

typedef void (*XtDoChangeProc) (Widget, WidgetList, Cardinal *,
                                WidgetList,
                                Cardinal *, XtPointer);
typedef struct _CompositeClassRec *CompositeWidgetClass;
extern void XtChangeManagedSet(WidgetList, Cardinal,
                                XtDoChangeProc,
                                XtPointer, WidgetList, Cardinal);
extern void XtManageChild(Widget);
extern void XtManageChildren(WidgetList, Cardinal);
extern void XtUnmanageChild(Widget);
extern void XtUnmanageChildren(WidgetList, Cardinal);
extern WidgetClass compositeWidgetClass;
```

6.10.2 X11/CompositeP.h

```
#define XtInheritGeometryManager ((XtGeometryHandler)
_XtInherit)
#define XtInheritChangeManaged ((XtWidgetProc) _XtInherit)
#define XtInheritDeleteChild ((XtWidgetProc) _XtInherit)
#define XtInheritInsertChild ((XtWidgetProc) _XtInherit)
#define _XtCompositeP_h 1
```

```

#define XtCompositeExtensionVersion    2L

typedef struct _CompositePart {
    WidgetList children;
    Cardinal num_children;
    Cardinal num_slots;
    XtOrderProc insert_position;
} CompositePart;
typedef struct _CompositeClassPart {
    XtGeometryHandler geometry_manager;
    XtWidgetProc change_managed;
    XtWidgetProc insert_child;
    XtWidgetProc delete_child;
    XtPointer extension;
} CompositeClassPart;
typedef struct _CompositePart *CompositePtr;
typedef struct _CompositeRec {
    CorePart core;
    CompositePart composite;
} CompositeRec;
typedef struct _CompositeClassPart *CompositePartPtr;
typedef struct {
    XtPointer next_extension;
    XrmQuark record_type;
    long int version;
    Cardinal record_size;
    Boolean accepts_objects;
    Boolean allows_change_managed_set;
} CompositeClassExtensionRec;
typedef CompositeClassExtensionRec *CompositeClassExtension;

typedef struct _CompositeClassRec CompositeClassRec;
extern CompositeClassRec compositeClassRec;

```

6.10.3 X11/ConstrainP.h

```

#define XtConstraintExtensionVersion    1L

typedef struct _ConstraintPart {
    XtPointer mumble; /* No new fields, keep C compiler
happy */
} ConstraintPart;
typedef struct _ConstraintClassRec *ConstraintClassRec;
typedef struct _ConstraintClassPart {
    XtResourceList resources;
    Cardinal num_resources;
    Cardinal constraint_size;
    XtInitProc initialize;
    XtWidgetProc destroy;
    XtSetValuesFunc set_values;
    XtPointer extension;
} ConstraintClassPart;
struct _ConstraintClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ConstraintClassPart constraint_class;
};
typedef struct _ConstraintRec {
    CorePart core;
    CompositePart composite;
    ConstraintPart constraint;
} ConstraintRec;
typedef struct _ConstraintRec *ConstraintWidget;
typedef struct {
    XtPointer next_extension;

```

```

    XrmQuark record_type;
    long int version;
    Cardinal record_size;
    XtArgsProc get_values_hook;
} ConstraintClassExtensionRec;
typedef ConstraintClassExtensionRec *ConstraintClassExtension;
extern ConstraintClassRec constraintClassRec;

```

6.10.4 X11/Constraint.h

```

#define _XtConstraint_h 1

typedef struct _ConstraintClassRec *ConstraintWidgetClass;
extern WidgetClass constraintWidgetClass;

```

6.10.5 X11/Core.h

```

#define _XtCore_h 1

typedef struct _WidgetClassRec *CoreWidgetClass;
typedef struct _WidgetClassRec *CoreWidget;
extern WidgetClass coreWidgetClass;
extern WidgetClass widgetClass;

```

6.10.6 X11/CoreP.h

```

#define XtInheritTranslations ((String) &_XtInheritTranslations)
#define XtInheritAcceptFocus ((XtAcceptFocusProc) _XtInherit)
#define XtInheritSetValuesAlmost ((XtAlmostProc) _XtInherit)
#define XtInheritExpose ((XtExposeProc) _XtInherit)
#define XtInheritQueryGeometry ((XtGeometryHandler) _XtInherit)
#define XtInheritRealize ((XtRealizeProc) _XtInherit)
#define XtInheritDisplayAccelerator ((XtStringProc) _XtInherit)
#define XtInheritResize ((XtWidgetProc) _XtInherit)
#define XtCoreP_h 1

typedef struct _CorePart {
    Widget self;
    WidgetClass widget_class;
    Widget parent;
    XrmName xrm_name;
    Boolean being_destroyed;
    XtCallbackList destroy_callbacks;
    XtPointer constraints;
    Position x;
    Position y;
    Dimension width;
    Dimension height;
    Dimension border_width;
    Boolean managed;
    Boolean sensitive;
    Boolean ancestor_sensitive;
    XtEventTable event_table;
    XtTMRc tm;
    XtTranslations accelerators;
    Pixel border_pixel;
    Pixmap border_pixmap;
    WidgetList popup_list;
    Cardinal num_popups;
    String name;
    Screen *screen;
    Colormap colormap;
}

```

```

        Window window;
        Cardinal depth;
        Pixel background_pixel;
        Pixmap background_pixmap;
        Boolean visible;
        Boolean mapped_when_managed;
    } CorePart;
typedef struct _CoreClassPart {
    WidgetClass superclass;
    String class_name;
    Cardinal widget_size;
    XtProc class_initialize;
    XtWidgetClassProc class_part_initialize;
    XtEnum class_initied;
    XtInitProc initialize;
    XtArgsProc initialize_hook;
    XtRealizeProc realize;
    XtActionList actions;
    Cardinal num_actions;
    XtResourceList resources;
    Cardinal num_resources;
    XrmClass xrm_class;
    Boolean compress_motion;
    XtEnum compress_exposure;
    Boolean compress_enterleave;
    Boolean visible_interest;
    XtWidgetProc destroy;
    XtWidgetProc resize;
    XtExposeProc expose;
    XtSetValuesFunc set_values;
    XtArgsFunc set_values_hook;
    XtAlmostProc set_values_almost;
    XtArgsProc get_values_hook;
    XtAcceptFocusProc accept_focus;
    XtVersionType version;
    XtPointer callback_private;
    String tm_table;
    XtGeometryHandler query_geometry;
    XtStringProc display_accelerator;
    XtPointer extension;
} CoreClassPart;
typedef struct _WidgetRec {
    CorePart core;
} WidgetRec;
typedef struct _WidgetRec {
    CorePart core;
} CoreRec;
typedef struct _WidgetClassRec {
    CoreClassPart core_class;
} WidgetClassRec;
typedef struct _WidgetClassRec {
    CoreClassPart core_class;
} CoreClassRec;
extern int _XtInheritTranslations;
extern WidgetClassRec widgetClassRec;

```

6.10.7 X11/Eventl.h

```

typedef struct _XtGrabRec *XtGrabList;
typedef struct _XtGrabRec {
    XtGrabList next;
    Widget widget;
    unsigned int exclusive:1;
    unsigned int spring_loaded:1;
} XtGrabRec;

```



```
typedef struct _XtEventRec {
    XtEventTable next;
    EventMask mask;
    XtEventHandler proc;
    XtPointer closure;
    unsigned int select:1;
    unsigned int has_type_specifier:1;
    unsigned int async:1;
} XtEventRec;
```

6.10.8 X11/Intrinsic.h

```
#define XtOffset(p_type, field) ((Cardinal) (((char *)
(&((p_type)NULL->field))) - ((char *) NULL)))
#define XtNumber(arr) ((Cardinal) (sizeof(arr) / sizeof(arr[0])))
#define XtAllEvents ((EventMask) -1L)
#define XtUnspecifiedPixmap ((Pixmap)2)
#define XtNew(type) ((type *) XtMalloc((unsigned) sizeof(type)))
#define XtSetArg(arg,n,d) ((void) ( (arg).name = (n),
(arg).value = (XtArgVal) (d) ))
#define XtUnspecifiedWindow ((Window)2)
#define XtUnspecifiedWindowGroup ((Window)3)
#define XtUnspecifiedShellInt (-1)
#define XtCWQueryOnly (1 << 7)
#define XtInputReadMask (1L<<0)
#define XtInputWriteMask (1L<<1)
#define XtInputExceptMask (1L<<2)
#define XT_CONVERT_FAIL (Atom)0x80000001
#define XtIMAll (XtIMXEvent | XtIMTimer | XtIMAlternateInput |
XtIMSignal)
#define XtIsRectObj(object) (_XtCheckSubclassFlag(object,
(XtEnum)0x02))
#define XtIsWidget(object) (_XtCheckSubclassFlag(object,
(XtEnum)0x04))
#define XtIsComposite(widget) (_XtCheckSubclassFlag(widget,
(XtEnum)0x08))
#define XtIsConstraint(widget) (_XtCheckSubclassFlag(widget,
(XtEnum)0x10))
#define XtIsShell(widget) (_XtCheckSubclassFlag(widget,
(XtEnum)0x20))
#define XtIsWMShell(widget) (_XtCheckSubclassFlag(widget,
(XtEnum)0x40))
#ifndef FALSE
#define FALSE 0
#endif
#define XtInputNoneMask 0L
#define XtCacheNone 0x001
#define XtCacheAll 0x002
#define XtCacheByDisplay 0x003
#define XtCacheRefCount 0x100
#ifndef TRUE
#define TRUE 1
#endif
#define XtIMXEvent 1
#define _XtIntrinsic_h 1
#define XtIMTimer 2
#define XtIMAlternateInput 4
#define XtSMDontChange 5
#define XtSpecificationRelease 6
#define XtIMSignal 8
#define _XtString char*
#define externalref extern
#define _XtBoolean int
#define _XtPosition int
#define XtOffsetOf(s_type, field) offsetof(s_type, field)
```

```

#define _XtDimension      unsigned int
#define _XtKeyCode        unsigned int
#define _XtXtEnum         unsigned int
#define XtCurrentDirectory "XtCurrentDirectory"
#define XtCvtPixelToColor  XtCvtIntToColor
#define XtCvtIntToDimension XtCvtIntToShort
#define XtCvtIntToPosition XtCvtIntToShort
#define XtCvtStringToPosition XtCvtStringToShort
#define XtDefaultBackground "XtDefaultBackground"
#define XtDefaultFont      "XtDefaultFont"
#define XtDefaultFontSet   "XtDefaultFontSet"
#define XtDefaultForeground "XtDefaultForeground"
#define XtVaNestedList     "XtVaNestedList"
#define XtVaTypedArg       "XtVaTypedArg"

typedef unsigned int Modifiers;
typedef void *XtCacheRef;
struct _XtAppStruct;
typedef struct _XtAppStruct *XtAppContext;
typedef int XtCacheType;
typedef long unsigned int XtGCMask;
typedef long unsigned int XtSignalId;
typedef long unsigned int XtWorkProcId;
typedef long unsigned int XtIntervalId;
typedef struct _TranslationData *XtAccelerators;
typedef enum {
    XtAddress = 0,
    XtBaseOffset = 1,
    XtImmediate = 2,
    XtResourceString = 3,
    XtResourceQuark = 4,
    XtWidgetBaseOffset = 5,
    XtProcedureArg = 6
} XtAddressMode;
typedef struct {
    XtAddressMode address_mode;
    XtPointer address_id;
    Cardinal size;
} XtConvertArgRec;
typedef XtConvertArgRec *XtConvertArgList;
typedef void (*XtSignalCallbackProc) (XtPointer, XtSignalId *);
typedef enum {
    XtCallbackNoList = 0,
    XtCallbackHasNone = 1,
    XtCallbackHasSome = 2
} XtCallbackStatus;
typedef void (*XtInputCallbackProc) (XtPointer, int *, XtInputId *);
typedef void (*XtTimerCallbackProc) (XtPointer, XtIntervalId *);
typedef long unsigned int XtInputMask;
typedef enum {
    XtListHead = 0,
    XtListTail = 1
} XtListPosition;
typedef void *XtVarArgsList;
typedef void (*XtEventHandler) (Widget, XtPointer, XEvent *, Boolean *);
typedef void (*XtCaseProc) (Display *, KeySym, KeySym *, KeySym *);
typedef void (*XtKeyProc) (Display *, unsigned int, Modifiers, Modifiers *, KeySym *);
typedef void (*XtBlockHookProc) (XtPointer);
typedef long unsigned int XtBlockHookId;
typedef void (*XtActionHookProc) (Widget, XtPointer, String, XEvent *, String *, Cardinal *);

```

```

typedef void *XtActionHookId;
typedef void (*XtDestructor) (XtAppContext, XrmValue *, XtPointer,
                             XrmValue *, Cardinal *);
typedef Boolean(*XtTypeConverter) (Display *, XrmValue *, Cardinal
*,
                                XrmValue *, XrmValue *, XtPointer *);
typedef void (*XtConverter) (XrmValue *, Cardinal *, XrmValue *,
                             XrmValue *);
typedef void *XtRequestId;
typedef Boolean(*XtFilePredicate) (String);
typedef struct {
    char match;
    String substitution;
} SubstitutionRec;
typedef SubstitutionRec *Substitution;
typedef void (*XtSelectionCallbackProc) (Widget, XtPointer, Atom *,
Atom *,
                                XtPointer, long unsigned int *,
                                int *);
typedef Boolean(*XtWorkProc) (XtPointer);
typedef void (*XtSelectionDoneProc) (Widget, Atom *, Atom *);
typedef void (*XtLoseSelectionProc) (Widget, Atom *);
typedef void (*XtErrorHandler) (String);
typedef Boolean(*XtConvertSelectionProc) (Widget, Atom *, Atom *,
Atom *,
                                XtPointer *, long unsigned int
*,
                                int *);
typedef void (*XtErrorMsgHandler) (String, String, String, String,
String *, Cardinal *);
typedef String(*XtLanguageProc) (Display *, String, XtPointer);
typedef struct _XtCheckpointTokenRec *XtCheckpointToken;
typedef void (*XtExtensionSelectProc) (Widget, int *, XtPointer *,
int,
                                XtPointer);
typedef Boolean(*XtEventDispatchProc) (XEvent *);
typedef void (*XtCancelConvertSelectionProc) (Widget, Atom *, Atom
*,
                                XtRequestId *, XtPointer);
typedef Boolean(*XtConvertSelectionIncrProc) (Widget, Atom *, Atom
*,
                                Atom *, XtPointer *,
                                long unsigned int *, int *,
                                long unsigned int *,
                                XtPointer, XtRequestId *);
typedef void (*XtSelectionDoneIncrProc) (Widget, Atom *, Atom *,
                                XtRequestId *, XtPointer);
typedef void (*XtLoseSelectionIncrProc) (Widget, Atom *, XtPointer);
typedef struct _XtResource XtResource;
typedef struct _XtCallbackRec XtCallbackRec;
typedef struct _XtCheckpointTokenRec {
    int save_type;
    int interact_style;
    Boolean shutdown;
    Boolean fast;
    Boolean cancel_shutdown;
    int phase;
    int interact_dialog_type;
    Boolean request_cancel;
    Boolean request_next_phase;
    Boolean save_success;
    int type;
    Widget widget;
} XtCheckpointTokenRec;
typedef struct _XtActionsRec XtActionsRec;
typedef struct {

```

```

    Widget shell_widget;
    Widget enable_widget;
} XtPopdownIDRec;
typedef XtPopdownIDRec *XtPopdownID;
typedef struct {
    String type;
    Widget widget;
    ArgList args;
    Cardinal num_args;
} XtCreateHookDataRec;
typedef XtCreateHookDataRec *XtCreateHookData;
typedef struct {
    String type;
    Widget widget;
    XtPointer event_data;
    Cardinal num_event_data;
} XtChangeHookDataRec;
typedef XtChangeHookDataRec *XtChangeHookData;
typedef struct {
    Widget old;
    Widget req;
    ArgList args;
    Cardinal num_args;
} XtChangeHookSetValuesDataRec;
typedef XtChangeHookSetValuesDataRec *XtChangeHookSetValuesData;
typedef struct {
    String type;
    Widget widget;
    XtGeometryMask changeMask;
    XWindowChanges changes;
} XtConfigureHookDataRec;
typedef XtConfigureHookDataRec *XtConfigureHookData;
typedef struct {
    String type;
    Widget widget;
    XtWidgetGeometry *request;
    XtWidgetGeometry *reply;
    XtGeometryResult result;
} XtGeometryHookDataRec;
typedef XtGeometryHookDataRec *XtGeometryHookData;
typedef struct {
    String type;
    Widget widget;
} XtDestroyHookDataRec;
typedef XtDestroyHookDataRec *XtDestroyHookData;
extern void XtAddActions(XtActionList, Cardinal);
extern void XtAddCallback(Widget, const char *, XtCallbackProc,
XtPointer);
extern void XtAddCallbacks(Widget, const char *, XtCallbackList);
extern void XtAddConverter(const char *, const char *, XtConverter,
XtConvertArgList, Cardinal);
extern void XtAddEventHandler(Widget, EventMask, int,
XtEventHandler,
XtPointer);
extern void XtAddExposureToRegion(XEvent *, Region);
extern void XtAddGrab(Widget, int, int);
extern XtInputId XtAddInput(int, XtPointer, XtInputCallbackProc,
XtPointer);
extern void XtAddRawEventHandler(Widget, EventMask, int,
XtEventHandler,
XtPointer);
extern XtSignalId XtAddSignal(XtSignalCallbackProc, XtPointer);
extern XtIntervalId XtAddTimeout(long unsigned int,
XtTimerCallbackProc,
XtPointer);
extern XtWorkProcId XtAddWorkProc(XtWorkProc, XtPointer);

```

```

extern GC XtAllocateGC(Widget, Cardinal, XtGCMask, XGCValues *,
XtGCMask,
XtGCMask);
extern XtActionHookId XtAppAddActionHook(XtAppContext,
XtActionHookProc,
XtPointer);
extern void XtAppAddActions(XtAppContext, XtActionList, Cardinal);
extern XtBlockHookId XtAppAddBlockHook(XtAppContext,
XtBlockHookProc,
XtPointer);
extern void XtAppAddConverter(XtAppContext, const char *, const
char *,
XtConverter, XtConvertArgList, Cardinal);
extern XtInputId XtAppAddInput(XtAppContext, int, XtPointer,
XtInputCallbackProc, XtPointer);
extern XtSignalId XtAppAddSignal(XtAppContext,
XtSignalCallbackProc,
XtPointer);
extern XtIntervalId XtAppAddTimeOut(XtAppContext, long unsigned int,
XtTimerCallbackProc, XtPointer);
extern XtWorkProcId XtAppAddWorkProc(XtAppContext, XtWorkProc,
XtPointer);
extern Widget XtAppCreateShell(const char *, const char *,
WidgetClass,
Display *, ArgList, Cardinal);
extern void XtAppError(XtAppContext, const char *);
extern void XtAppErrorMsg(XtAppContext, const char *, const char *,
const char *, const char *, String *,
Cardinal *);
extern XrmDatabase *XtAppGetErrorDatabase(XtAppContext);
extern void XtAppGetErrorDatabaseText(XtAppContext, const char *,
const char *, const char *,
const char *, String, int,
XrmDatabase);
extern Boolean XtAppGetExitFlag(XtAppContext);
extern long unsigned int XtAppGetSelectionTimeout(XtAppContext);
extern Widget XtAppInitialize(XtAppContext *, const char *,
XrmOptionDescList, Cardinal, int *,
String *,
String *, ArgList, Cardinal);
extern void XtAppLock(XtAppContext);
extern void XtAppMainLoop(XtAppContext);
extern void XtAppNextEvent(XtAppContext, XEvent *);
extern Boolean XtAppPeekEvent(XtAppContext, XEvent *);
extern XtInputMask XtAppPending(XtAppContext);
extern void XtAppProcessEvent(XtAppContext, XtInputMask);
extern void XtAppReleaseCacheRefs(XtAppContext, XtCacheRef *);
extern XtErrorHandler XtAppSetErrorHandler(XtAppContext,
XtErrorHandler);
extern XtErrorMsgHandler XtAppSetErrorMsgHandler(XtAppContext,
XtErrorMsgHandler);
extern void XtAppSetExitFlag(XtAppContext);
extern void XtAppSetFallbackResources(XtAppContext, String *);
extern void XtAppSetSelectionTimeout(XtAppContext, long unsigned
int);
extern void XtAppSetTypeConverter(XtAppContext, const char *, const
char *,
XtTypeConverter, XtConvertArgList,
Cardinal, XtCacheType, XtDestructor);
extern XtErrorHandler XtAppSetWarningHandler(XtAppContext,
XtErrorHandler);
extern XtErrorMsgHandler XtAppSetWarningMsgHandler(XtAppContext,
XtErrorMsgHandler);
extern void XtAppUnlock(XtAppContext);
extern void XtAppWarning(XtAppContext, const char *);

```

```

extern void XtAppWarningMsg(XtAppContext, const char *, const char
*,
                        const char *, const char *, String *,
                        Cardinal *);
extern void XtAugmentTranslations(Widget, XtTranslations);
extern EventMask XtBuildEventMask(Widget);
extern Boolean XtCallAcceptFocus(Widget, Time *);
extern void XtCallActionProc(Widget, const char *, XEvent *, String
*,
                        Cardinal);
extern void XtCallCallbackList(Widget, XtCallbackList, XtPointer);
extern void XtCallCallbacks(Widget, const char *, XtPointer);
extern Boolean XtCallConverter(Display *, XtTypeConverter,
XrmValuePtr,
                        Cardinal, XrmValuePtr, XrmValue *,
                        XtCacheRef *);
extern void XtCallbackExclusive(Widget, XtPointer, XtPointer);
extern void XtCallbackNone(Widget, XtPointer, XtPointer);
extern void XtCallbackNonexclusive(Widget, XtPointer, XtPointer);
extern void XtCallbackPopdown(Widget, XtPointer, XtPointer);
extern void XtCallbackReleaseCacheRef(Widget, XtPointer,
XtPointer);
extern void XtCallbackReleaseCacheRefList(Widget, XtPointer,
XtPointer);
extern char *XtCalloc(Cardinal, Cardinal);
extern void XtCancelSelectionRequest(Widget, Atom);
extern WidgetClass XtClass(Widget);
extern void XtCloseDisplay(Display *);
extern void XtConvert(Widget, const char *, XrmValue *, const char
*,
                        XrmValue *);
extern Boolean XtConvertAndStore(Widget, const char *, XrmValue *,
                        const char *, XrmValue *);
extern void XtConvertCase(Display *, KeySym, KeySym *, KeySym *);
extern XtAppContext XtCreateApplicationContext(void);
extern Widget XtCreateApplicationShell(const char *, WidgetClass,
ArgList,
                        Cardinal);
extern Widget XtCreateManagedWidget(const char *, WidgetClass,
Widget,
                        ArgList, Cardinal);
extern Widget XtCreatePopupShell(const char *, WidgetClass, Widget,
ArgList, Cardinal);
extern void XtCreateSelectionRequest(Widget, Atom);
extern Widget XtCreateWidget(const char *, WidgetClass, Widget,
ArgList,
                        Cardinal);
extern Boolean XtCvtColorToPixel(Display *, XrmValuePtr, Cardinal
*,
                        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToBool(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToBoolean(Display *, XrmValuePtr, Cardinal
*,
                        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToColor(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToFloat(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToFont(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToPixel(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToPixmap(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToShort(Display *, XrmValuePtr, Cardinal *,
XrmValuePtr, XrmValuePtr, XtPointer *);

```

```

        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtIntToUnsignedChar(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr,
        XtPointer *);
extern Boolean XtCvtStringToAcceleratorTable(Display *,
XrmValuePtr,
        Cardinal *, XrmValuePtr,
        XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToAtom(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToBool(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToBoolean(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr, XtPointer
*);
extern Boolean XtCvtStringToCommandArgArray(Display *, XrmValuePtr,
        Cardinal *, XrmValuePtr,
        XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToCursor(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer
*);
extern Boolean XtCvtStringToDimension(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr,
        XtPointer *);
extern Boolean XtCvtStringToDirectoryString(Display *, XrmValuePtr,
        Cardinal *, XrmValuePtr,
        XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToDisplay(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr, XtPointer
*);
extern Boolean XtCvtStringToFile(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToFloat(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToFont(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToFontSet(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr, XtPointer
*);
extern Boolean XtCvtStringToFontStruct(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr,
        XtPointer *);
extern Boolean XtCvtStringToGravity(Display *, XrmValuePtr,
Cardinal *,
        XrmValuePtr, XrmValuePtr, XtPointer
*);
extern Boolean XtCvtStringToInitialState(Display *, XrmValuePtr,
        Cardinal *, XrmValuePtr,
        XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToInt(Display *, XrmValuePtr, Cardinal *,
        XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToPixel(Display *, XrmValuePtr, Cardinal
*,
        XrmValuePtr, XrmValuePtr, XtPointer *);

```

```

extern Boolean XtCvtStringToRestartStyle(Display *, XrmValuePtr,
                                           Cardinal *, XrmValuePtr,
                                           XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToShort(Display *, XrmValuePtr, Cardinal
*,
                                   XrmValuePtr, XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToTranslationTable(Display *,
XrmValuePtr,
                                           Cardinal *, XrmValuePtr,
                                           XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToUnsignedChar(Display *, XrmValuePtr,
                                           Cardinal *, XrmValuePtr,
                                           XrmValuePtr, XtPointer *);
extern Boolean XtCvtStringToVisual(Display *, XrmValuePtr, Cardinal
*,
                                   XrmValuePtr, XrmValuePtr, XtPointer
*);
extern XrmDatabase XtDatabase(Display *);
extern void XtDestroyApplicationContext(XtAppContext);
extern void XtDestroyGC(GC);
extern void XtDestroyWidget(Widget);
extern void XtDirectConvert(XtConverter, XrmValuePtr, Cardinal,
XrmValuePtr, XrmValue *);
extern void XtDisownSelection(Widget, Atom, Time);
extern Boolean XtDispatchEvent(XEvent *);
extern Boolean XtDispatchEventToWidget(Widget, XEvent *);
extern Display *XtDisplay(Widget);
extern void XtDisplayInitialize(XtAppContext, Display *, const char
*,
                                const char *, XrmOptionDescRec *,
                                Cardinal,
                                int *, char **);
extern Display *XtDisplayOfObject(Widget);
extern void XtDisplayStringConversionWarning(Display *, const char
*,
                                const char *);
extern XtAppContext XtDisplayToApplicationContext(Display *);
extern void XtError(const char *);
extern void XtErrorMsg(const char *, const char *, const char *,
const char *, String *, Cardinal *);
extern String XtFindFile(const char *, Substitution, Cardinal,
XtFilePredicate);
extern void XtFree(char *);
extern KeySym XtGetActionKeysym(XEvent *, Modifiers *);
extern void XtGetActionList(WidgetClass, XtActionList *, Cardinal
*);
extern void XtGetApplicationNameAndClass(Display *, String *,
String *);
extern void XtGetApplicationResources(Widget, XtPointer,
XtResourceList,
                                   Cardinal, ArgList, Cardinal);
extern XtPointer XtGetClassExtension(WidgetClass, Cardinal,
XrmQuark,
                                   long int, Cardinal);
extern void XtGetConstraintResourceList(WidgetClass,
XtResourceList *,
                                   Cardinal *);
extern void XtGetDisplays(XtAppContext, Display * **, Cardinal *);
extern XrmDatabase *XtGetErrorDatabase(void);
extern void XtGetErrorDatabaseText(const char *, const char *,
const char *, const char *, String,
int);
extern GC XtGetGC(Widget, XtGCMask, XGCValues *);
extern Widget XtGetKeyboardFocusWidget(Widget);
extern KeySym *XtGetKeysymTable(Display *, KeyCode *, int *);
extern int XtGetMultiClickTime(Display *);

```



```

extern void XtGetResourceList(WidgetClass, XtResourceList *,
Cardinal *);
extern void XtGetSelectionParameters(Widget, Atom, XtRequestId,
Atom *,
                                XtPointer *, long unsigned int *,
                                int *);
extern XSelectionRequestEvent *XtGetSelectionRequest(Widget, Atom,
XtRequestId);
extern long unsigned int XtGetSelectionTimeout(void);
extern void XtGetSelectionValue(Widget, Atom, Atom,
                                XtSelectionCallbackProc, XtPointer,
Time);
extern void XtGetSelectionValueIncremental(Widget, Atom, Atom,
XtSelectionCallbackProc,
XtPointer, Time);
extern void XtGetSelectionValues(Widget, Atom, Atom *, int,
XtSelectionCallbackProc, XtPointer *,
Time);
extern void XtGetSelectionValuesIncremental(Widget, Atom, Atom *,
int,
                                XtSelectionCallbackProc,
                                XtPointer *, Time);
extern void XtGetSubresources(Widget, XtPointer, const char *,
const char *, XtResourceList, Cardinal,
ArgList, Cardinal);
extern void XtGetSubvalues(XtPointer, XtResourceList, Cardinal,
ArgList,
Cardinal);
extern void XtGetValues(Widget, ArgList, Cardinal);
extern void XtGrabButton(Widget, int, Modifiers, int, unsigned int,
int,
int, Window, Cursor);
extern void XtGrabKey(Widget, unsigned int, Modifiers, int, int,
int);
extern int XtGrabKeyboard(Widget, int, int, int, Time);
extern int XtGrabPointer(Widget, int, unsigned int, int, int,
Window,
Cursor, Time);
extern XtCallbackStatus XtHasCallbacks(Widget, const char *);
extern Widget XtHooksOfDisplay(Display *);
extern Widget XtInitialize(const char *, const char *,
XrmOptionDescRec *,
Cardinal, int *, char **);
extern void XtInitializeWidgetClass(WidgetClass);
extern void XtInsertEventHandler(Widget, EventMask, int,
XtEventHandler,
XtPointer, XtListPosition);
extern void XtInsertEventTypeHandler(Widget, int, XtPointer,
XtEventHandler, XtPointer,
XtListPosition);
extern void XtInsertRawEventHandler(Widget, EventMask, int,
XtEventHandler,
XtPointer, XtListPosition);
extern void XtInstallAccelerators(Widget, Widget);
extern void XtInstallAllAccelerators(Widget, Widget);
extern Boolean XtIsApplicationShell(Widget);
extern Boolean XtIsManaged(Widget);
extern Boolean XtIsObject(Widget);
extern Boolean XtIsOverrideShell(Widget);
extern Boolean XtIsRealized(Widget);
extern Boolean XtIsSensitive(Widget);
extern Boolean XtIsSessionShell(Widget);
extern Boolean XtIsSubclass(Widget, WidgetClass);
extern Boolean XtIsTransientShell(Widget);
extern Boolean XtIsVendorShell(Widget);
extern void XtKeysymToKeycodeList(Display *, KeySym, KeyCode * *,

```

```

        Cardinal *);
extern XEvent *XtLastEventProcessed(Display *);
extern Time XtLastTimestampProcessed(Display *);
extern void XtMainLoop(void);
extern XtGeometryResult XtMakeGeometryRequest(Widget,
XtWidgetGeometry *,
        XtWidgetGeometry *);
extern XtGeometryResult XtMakeResizeRequest(Widget, unsigned int,
        unsigned int, Dimension *,
        Dimension *);

extern char *XtMalloc(Cardinal);
extern void XtMapWidget(Widget);
extern void XtMenuPopupAction(Widget, XEvent *, String *, Cardinal
*);
extern ArgList XtMergeArgLists(ArgList, Cardinal, ArgList,
Cardinal);
extern String XtName(Widget);
extern Widget XtNameToWidget(Widget, const char *);
extern String XtNewString(String);
extern void XtNextEvent(XEvent *);
extern void XtNoticeSignal(XtSignalId);
extern Widget XtOpenApplication(XtAppContext *, const char *,
        XrmOptionDescList, Cardinal, int *,
        String *, String *, WidgetClass, ArgList,
        Cardinal);
extern Display *XtOpenDisplay(XtAppContext, const char *, const
char *,
        const char *, XrmOptionDescRec *, Cardinal,
        int *, char **);
extern void XtOverrideTranslations(Widget, XtTranslations);
extern Boolean XtOwnSelection(Widget, Atom, Time,
XtConvertSelectionProc,
        XtLoseSelectionProc, XtSelectionDoneProc);
extern Boolean XtOwnSelectionIncremental(Widget, Atom, Time,
        XtConvertSelectionIncrProc,
        XtLoseSelectionIncrProc,
        XtSelectionDoneIncrProc,
        XtCancelConvertSelectionProc,
        XtPointer);

extern Widget XtParent(Widget);
extern XtAccelerators XtParseAcceleratorTable(const char *);
extern XtTranslations XtParseTranslationTable(const char *);
extern Boolean XtPeekEvent(XEvent *);
extern Boolean XtPending(void);
extern void XtPopdown(Widget);
extern void XtPopup(Widget, XtGrabKind);
extern void XtPopupSpringLoaded(Widget);
extern void XtProcessEvent(XtInputMask);
extern XtGeometryResult XtQueryGeometry(Widget, XtWidgetGeometry *,
        XtWidgetGeometry *);
extern void XtRealizeWidget(Widget);
extern char *XtRealloc(char *, Cardinal);
extern void XtRegisterCaseConverter(Display *, XtCaseProc, KeySym,
KeySym);
extern void XtRegisterDrawable(Display *, Drawable, Widget);
extern void XtRegisterExtensionSelector(Display *, int, int,
        XtExtensionSelectProc,
        XtPointer);
extern void XtRegisterGrabAction(XtActionProc, int, unsigned int,
int,
        int);
extern void XtReleaseGC(Widget, GC);
extern void XtReleasePropertyAtom(Widget, Atom);
extern void XtRemoveActionHook(XtActionHookId);
extern void XtRemoveAllCallbacks(Widget, const char *);
extern void XtRemoveBlockHook(XtBlockHookId);

```

```

extern void XtRemoveCallback(Widget, const char *, XtCallbackProc,
                             XtPointer);
extern void XtRemoveCallbacks(Widget, const char *, XtCallbackList);
extern void XtRemoveEventHandler(Widget, EventMask, int,
                                XtEventHandler,
                                XtPointer);
extern void XtRemoveEventHandler(Widget, int, XtPointer,
                                XtEventHandler, XtPointer);
extern void XtRemoveGrab(Widget);
extern void XtRemoveInput(XtInputId);
extern void XtRemoveRawEventHandler(Widget, EventMask, int,
                                    XtEventHandler,
                                    XtPointer);
extern void XtRemoveSignal(XtSignalId);
extern void XtRemoveTimeout(XtIntervalId);
extern void XtRemoveWorkProc(XtWorkProcId);
extern Atom XtReservePropertyAtom(Widget);
extern String XtResolvePathname(Display *, const char *, const char
*,
                                const char *, const char *, Substitution,
                                Cardinal, XtFilePredicate);
extern Screen *XtScreen(Widget);
extern XrmDatabase XtScreenDatabase(Screen *);
extern Screen *XtScreenOfObject(Widget);
extern void XtSendSelectionRequest(Widget, Atom, Time);
extern XtCheckpointToken XtSessionGetToken(Widget);
extern void XtSessionReturnToken(XtCheckpointToken);
extern void XtSetErrorHandler(XtErrorHandler);
extern void XtSetErrorMsgHandler(XtErrorMsgHandler);
extern XtEventDispatchProc XtSetEventDispatcher(Display *, int,
                                                XtEventDispatchProc);
extern void XtSetKeyTranslator(Display *, XtKeyProc);
extern void XtSetKeyboardFocus(Widget, Widget);
extern XtLanguageProc XtSetLanguageProc(XtAppContext,
                                         XtLanguageProc,
                                         XtPointer);
extern void XtSetMappedWhenManaged(Widget, int);
extern void XtSetMultiClickTime(Display *, int);
extern void XtSetSelectionParameters(Widget, Atom, Atom, XtPointer,
                                     long unsigned int, int);
extern void XtSetSelectionTimeout(long unsigned int);
extern void XtSetSensitive(Widget, int);
extern void XtSetSubvalues(XtPointer, XtResourceList, Cardinal,
                           ArgList,
                           Cardinal);
extern void XtSetTypeConverter(const char *, const char *,
                               XtTypeConverter,
                               XtConvertArgList, Cardinal, XtCacheType,
                               XtDestructor);
extern void XtSetValues(Widget, ArgList, Cardinal);
extern void XtSetWMColormapWindows(Widget, Widget *, Cardinal);
extern void XtSetWarningHandler(XtErrorHandler);
extern void XtSetWarningMsgHandler(XtErrorMsgHandler);
extern void XtStringConversionWarning(const char *, const char *);
extern WidgetClass XtSuperclass(Widget);
extern void XtToolkitInitialize(void);
extern Boolean XtToolkitThreadInitialize(void);
extern void XtTranslateCoords(Widget, int, int, Position *,
                              Position *);
extern void XtTranslateKey(Display *, unsigned int, Modifiers,
                           Modifiers *,
                           KeySym *);
extern void XtTranslateKeyCode(Display *, unsigned int, Modifiers,
                              Modifiers *, KeySym *);
extern void XtUngrabButton(Widget, unsigned int, Modifiers);
extern void XtUngrabKey(Widget, unsigned int, Modifiers);

```

```

extern void XtUngrabKeyboard(Widget, Time);
extern void XtUngrabPointer(Widget, Time);
extern void XtUninstallTranslations(Widget);
extern void XtUnmapWidget(Widget);
extern void XtUnrealizeWidget(Widget);
extern void XtUnregisterDrawable(Display *, Drawable);
extern Widget XtVaAppCreateShell(const char *, const char *,
WidgetClass,
                                Display *, ...);
extern Widget XtVaAppInitialize(XtAppContext *, const char *,
                                XrmOptionDescList, Cardinal, int *,
                                String *, String *, ...);
extern XtVarArgsList XtVaCreateArgsList(XtPointer, ...);
extern Widget XtVaCreateManagedWidget(const char *, WidgetClass,
Widget,
                                ...);
extern Widget XtVaCreatePopupShell(const char *, WidgetClass,
Widget, ...);
extern Widget XtVaCreateWidget(const char *, WidgetClass,
Widget, ...);
extern void XtVaGetApplicationResources(Widget, XtPointer,
XtResourceList,
                                Cardinal, ...);
extern void XtVaGetSubresources(Widget, XtPointer, const char *,
                                const char *, XtResourceList, Cardinal,
                                ...);
extern void XtVaGetSubvalues(XtPointer, XtResourceList,
Cardinal, ...);
extern void XtVaGetValues(Widget, ...);
extern Widget XtVaOpenApplication(XtAppContext *, const char *,
                                XrmOptionDescList, Cardinal, int *,
                                String *, String *, WidgetClass, ...);
extern void XtVaSetSubvalues(XtPointer, XtResourceList,
Cardinal, ...);
extern void XtVaSetValues(Widget, ...);
extern void XtWarning(const char *);
extern void XtWarningMsg(const char *, const char *, const char *,
                                const char *, String *, Cardinal *);
extern XtAppContext XtWidgetToApplicationContext(Widget);
extern Window XtWindow(Widget);
extern Window XtWindowOfObject(Widget);
extern Widget XtWindowToWidget(Display *, Window);
extern Boolean _XtCheckSubclassFlag(Widget, unsigned int);
extern Boolean _XtIsSubclassOf(Widget, WidgetClass, WidgetClass,
                                unsigned int);
extern const XtConvertArgRec colorConvertArgs[];

```

6.10.9 X11/Intrinsic.h

```

#define CALLGEOTAT(f)
#define XTERROR_PREFIX ""
#define XTWARNING_PREFIX ""
#define XtStackAlloc(size, stack_cache_array) ((size) <=
sizeof(stack_cache_array) ? (XtPointer)(stack_cache_array) :
XtMalloc((unsigned)(size)))
#define XtWindowOfObject(object) ((XtIsWidget(object) ?
(object) : _XtWindowedAncestor(object)) ->core.window)
#define XtIsSensitive(object) (XtIsRectObj(object) ? ((object)-
>core.sensitive && (object)->core.ancestor_sensitive) : False)
#define XtIsManaged(object) (XtIsRectObj(object) ? (object)-
>core.managed : False)
#define XtScreenOfObject(object) (XtIsWidget(object) ?
(object)->core.screen : _XtIsHookObject(object) ?
((HookObject)(object))->hooks.screen :
_XtWindowedAncestor(object)->core.screen)

```

```

#define XtDisplayOfObject(object)          (XtIsWidget(object) ?
(object)->core.screen->display : _XtIsHookObject(object) ?
((HookObject)(object))->hooks.screen->display :
_XtWindowedAncestor(object)->core.screen->display)
#define XFILESEARCHPATHDEFAULT
"/usr/lib/X11/%L/%T/%N%S:/usr/lib/X11/%l/%T/%N%S:/usr/lib/X11/%T/
%N%S"
#define ERRORDB "/usr/lib/X11/XtErrorDB"
#define RectObjClassFlag 0x02
#define WidgetClassFlag 0x04
#define CompositeClassFlag 0x08
#define ConstraintClassFlag 0x10
#define ShellClassFlag 0x20
#define WMShellClassFlag 0x40
#define TopLevelClassFlag 0x80
#define _XtintrinsicI_h 1
#define XtBZero(dst,size) bzero((char *) (dst), (int) (size))
#define XtMemmove(dst,src,size) if ((char *) (dst) != (char *) (src))
{ (void) memcpy((char *) (dst), (char *) (src), (int) (size)); }
#define XtMemcmp(b1,b2,size) memcmp((char *) (b1), (char *) (b2),
(int) (size))
#define _XBCOPYFUNC _XtBcopy
#define XtStackFree(pointer,stack_cache_array) { if ((pointer) !=
((XtPointer)(stack_cache_array))) XtFree(pointer); }

extern String XtCXtToolkitError;

```

6.10.10 X11/IntrinsicP.h

```

#define XtCheckSubclass(w,widget_class,message)
#define XtExposeNoCompress ((XtEnum)False)
#define XtExposeCompressSeries ((XtEnum)True)
#define XtVersion (XT_VERSION * 1000 + XT_REVISION)
#define XtVersionDontCheck 0
#define XtExposeGraphicsExpose 0x10
#define XtExposeGraphicsExposeMerged 0x20
#define XtExposeNoExpose 0x40
#define XtExposeNoRegion 0x80
#define _XtintrinsicP_h 1
#define XT_VERSION 11
#define XtExposeCompressMultiple 2
#define XtExposeCompressMaximal 3
#define XT_REVISION 6

typedef struct {
    long int xrm_name;
    long int xrm_class;
    long int xrm_type;
    Cardinal xrm_size;
    int xrm_offset;
    long int xrm_default_type;
    XtPointer xrm_default_addr;
} XrmResource;
typedef XrmResource *XrmResourceList;
typedef struct _XtTMRec *XtTM;
typedef struct {
    String name;
    String type;
    XtArgVal value;
    int size;
} XtTypedArg;
typedef XtTypedArg *XtTypedArgList;
typedef void (*XtProc) (void);
typedef void (*XtWidgetClassProc) (WidgetClass);
typedef void (*XtInitProc) (Widget, Widget, ArgList, Cardinal *);

```

```

typedef void (*XtArgsProc) (Widget, ArgList, Cardinal *);
typedef void (*XtRealizeProc) (Widget, XtValueMask *,
                               XSetWindowAttributes *);
typedef void (*XtWidgetProc) (Widget);
typedef void (*XtExposeProc) (Widget, XEvent *, Region);
typedef Boolean (*XtArgsFunc) (Widget, ArgList, Cardinal *);
typedef void (*XtAlmostProc) (Widget, Widget, XtWidgetGeometry *,
                              XtWidgetGeometry *);
typedef Boolean (*XtAcceptFocusProc) (Widget, Time *);
typedef long unsigned int XtVersionType;
typedef XtGeometryResult (*XtGeometryHandler) (Widget,
XtWidgetGeometry *,
                               XtWidgetGeometry *);
typedef void (*XtAllocateProc) (WidgetClass, Cardinal *, Cardinal
*,
                               ArgList, Cardinal *, XtTypedArgList,
                               Cardinal *, struct _WidgetRec * *,
                               void **);
typedef void (*XtDeallocateProc) (struct _WidgetRec *, void *);
typedef struct _XtTMRc {
    XtTranslations translations;
    XtBoundActions proc_table;
    struct _XtStateRec *current_state;
    long unsigned int lastEventTime;
} XtTMRc;

typedef void (*XtStringProc) (Widget, String);
typedef Boolean (*XtSetValuesFunc) (Widget, Widget, Widget, ArgList,
                                   Cardinal *);
extern void XtConfigureWidget(Widget, int, int, unsigned int,
                              unsigned int);
extern void XtCreateWindow(Widget, unsigned int, Visual *,
XtValueMask,
                              XSetWindowAttributes *);
extern Boolean XtIsComposite(Widget);
extern Boolean XtIsConstraint(Widget);
extern Boolean XtIsRectObj(Widget);
extern Boolean XtIsShell(Widget);
extern Boolean XtIsTopLevelShell(Widget);
extern Boolean XtIsWMShell(Widget);
extern Boolean XtIsWidget(Widget);
extern void XtMoveWidget(Widget, int, int);
extern void XtProcessLock(void);
extern void XtProcessUnlock(void);
extern void XtResizeWidget(Widget, unsigned int, unsigned int,
                           unsigned int);
extern void XtResizeWindow(Widget);
extern void _XtInherit(void);

```

6.10.11 X11/Object.h

```

#define _XtObject_h 1

typedef struct _ObjectClassRec *ObjectClass;
typedef struct _ObjectClassRec *Object;
extern WidgetClass objectClass;

```

6.10.12 X11/ObjectP.h

```

#define XtInheritAllocate ((XtAllocateProc) _XtInherit)
#define XtInheritDeallocate ((XtDeallocateProc) _XtInherit)
#define _Xt_ObjectP_h_ 1

```

```

#define XtObjectExtensionVersion      1L

typedef struct _ObjectPart {
    Widget self;
    WidgetClass widget_class;
    Widget parent;
    XrmName xrm_name;
    Boolean being_destroyed;
    XtCallbackList destroy_callbacks;
    XtPointer constraints;
} ObjectPart;
typedef struct _ObjectRec {
    ObjectPart object;
} ObjectRec;
typedef struct _ObjectClassPart {
    WidgetClass superclass;
    String class_name;
    Cardinal widget_size;
    XtProc class_initialize;
    XtWidgetClassProc class_part_initialize;
    XtEnum class_inited;
    XtInitProc initialize;
    XtArgsProc initialize_hook;
    XtProc obj1;
    XtPointer obj2;
    Cardinal obj3;
    XtResourceList resources;
    Cardinal num_resources;
    XrmClass xrm_class;
    Boolean obj4;
    XtEnum obj5;
    Boolean obj6;
    Boolean obj7;
    XtWidgetProc destroy;
    XtProc obj8;
    XtProc obj9;
    XtSetValuesFunc set_values;
    XtArgsFunc set_values_hook;
    XtProc obj10;
    XtArgsProc get_values_hook;
    XtProc obj11;
    XtVersionType version;
    XtPointer callback_private;
    String obj12;
    XtProc obj13;
    XtProc obj14;
    XtPointer extension;
} ObjectClassPart;
typedef struct {
    XtPointer next_extension;
    XrmQuark record_type;
    long int version;
    Cardinal record_size;
    XtAllocateProc allocate;
    XtDeallocateProc deallocate;
} ObjectClassExtensionRec;
typedef ObjectClassExtensionRec *ObjectClassExtension;

typedef struct _ObjectClassRec {
    ObjectClassPart object_class;
} ObjectClassRec;
extern ObjectClassRec objectClassRec;

```

6.10.13 X11/RectObj.h

```
typedef struct _RectObjRec *RectObj;
typedef struct _RectObjClassRec *RectObjClass;
extern WidgetClass rectObjClass;
```

6.10.14 X11/RectObjP.h

```
typedef struct _RectObjPart {
    Position x;
    Position y;
    Dimension width;
    Dimension height;
    Dimension border_width;
    Boolean managed;
    Boolean sensitive;
    Boolean ancestor_sensitive;
} RectObjPart;

typedef struct _RectObjRec {
    ObjectPart object;
    RectObjPart rectangle;
} RectObjRec;

typedef struct _RectObjClassPart {
    WidgetClass superclass;
    String class_name;
    Cardinal widget_size;
    XtProc class_initialize;
    XtWidgetClassProc class_part_initialize;
    XtEnum class_inited;
    XtInitProc initialize;
    XtArgsProc initialize_hook;
    XtProc rect1;
    XtPointer rect2;
    Cardinal rect3;
    XtResourceList resources;
    Cardinal num_resources;
    XrmClass xrm_class;
    Boolean rect4;
    XtEnum rect5;
    Boolean rect6;
    Boolean rect7;
    XtWidgetProc destroy;
    XtWidgetProc resize;
    XtExposeProc expose;
    XtSetValuesFunc set_values;
    XtArgsFunc set_values_hook;
    XtAlmostProc set_values_almost;
    XtArgsProc get_values_hook;
    XtProc rect9;
    XtVersionType version;
    XtPointer callback_private;
    String rect10;
    XtGeometryHandler query_geometry;
    XtProc rect11;
    XtPointer extension;
} RectObjClassPart;

typedef struct _RectObjClassRec {
    RectObjClassPart rect_class;
} RectObjClassRec;
extern RectObjClassRec rectObjClassRec;
```


6.10.15 X11/Resource.h

```
#define _XtresourceI_h
#define StringToClass(string)    XrmStringToQuark(string)
#define StringToName(string)     XrmStringToQuark(string)
#define StringToQuark(string)    XrmStringToQuark(string)

extern void _XtCopyFromArg(XtArgVal, char *, unsigned int);
```

6.10.16 X11/Shell.h

```
#define XtNiconName      ((char*)&XtShellStrings[0])
#define XtCDiscardCommand ((char*)&XtShellStrings[1001])
#define XtNenvironment  ((char*)&XtShellStrings[1016])
#define XtCEnvironment  ((char*)&XtShellStrings[1028])
#define XtNinteractCallback ((char*)&XtShellStrings[1040])
#define XtNvisual        ((char*)&XtShellStrings[104])
#define XtNjoinSession   ((char*)&XtShellStrings[1057])
#define XtCJoinSession   ((char*)&XtShellStrings[1069])
#define XtNprogramPath   ((char*)&XtShellStrings[1081])
#define XtCProgramPath   ((char*)&XtShellStrings[1093])
#define XtNresignCommand ((char*)&XtShellStrings[1105])
#define XtCResignCommand ((char*)&XtShellStrings[1119])
#define XtCvisual        ((char*)&XtShellStrings[111])
#define XtNrestartCommand ((char*)&XtShellStrings[1133])
#define XtCRestartCommand ((char*)&XtShellStrings[1148])
#define XtNrestartStyle   ((char*)&XtShellStrings[1163])
#define XtCRestartStyle   ((char*)&XtShellStrings[1176])
#define XtNsaveCallback   ((char*)&XtShellStrings[1189])
#define XtNtitleEncoding  ((char*)&XtShellStrings[118])
#define XtNsaveCompleteCallback ((char*)&XtShellStrings[1202])
#define XtNsessionId      ((char*)&XtShellStrings[1223])
#define XtCsessionId      ((char*)&XtShellStrings[1233])
#define XtNshutdownCommand ((char*)&XtShellStrings[1243])
#define XtCShutdownCommand ((char*)&XtShellStrings[1259])
#define XtNerrorCallback   ((char*)&XtShellStrings[1275])
#define XtCtitleEncoding   ((char*)&XtShellStrings[132])
#define XtNsaveUnder      ((char*)&XtShellStrings[146])
#define XtCSaveUnder      ((char*)&XtShellStrings[156])
#define XtNtransient       ((char*)&XtShellStrings[166])
#define XtCTransient       ((char*)&XtShellStrings[176])
#define XtNoverrideRedirect ((char*)&XtShellStrings[186])
#define XtNiconPixmap      ((char*)&XtShellStrings[18])
#define XtCOVERRIDERedirect ((char*)&XtShellStrings[203])
#define XtNtransientFor    ((char*)&XtShellStrings[220])
#define XtCTransientFor    ((char*)&XtShellStrings[233])
#define XtNiconNameEncoding ((char*)&XtShellStrings[246])
#define XtCIconNameEncoding ((char*)&XtShellStrings[263])
#define XtNallowShellResize ((char*)&XtShellStrings[280])
#define XtCallowShellResize ((char*)&XtShellStrings[297])
#define XtCIconPixmap      ((char*)&XtShellStrings[29])
#define XtNcreatePopupChildProc ((char*)&XtShellStrings[314])
#define XtCCreatePopupChildProc ((char*)&XtShellStrings[335])
#define XtNtitle           ((char*)&XtShellStrings[356])
#define XtCtitle           ((char*)&XtShellStrings[362])
#define XtNargc             ((char*)&XtShellStrings[373])
#define XtCargc             ((char*)&XtShellStrings[378])
#define XtNargv             ((char*)&XtShellStrings[383])
#define XtCargv             ((char*)&XtShellStrings[388])
#define XtNiconX            ((char*)&XtShellStrings[393])
#define XtCIconX            ((char*)&XtShellStrings[399])
#define XtNiconY            ((char*)&XtShellStrings[405])
#define XtNiconWindow       ((char*)&XtShellStrings[40])
```

```

#define XtCIconY          ((char*)&XtShellStrings[411])
#define XtNinput          ((char*)&XtShellStrings[417])
#define XtCInput          ((char*)&XtShellStrings[423])
#define XtNiconic         ((char*)&XtShellStrings[429])
#define XtCIconic         ((char*)&XtShellStrings[436])
#define XtNInitialState   ((char*)&XtShellStrings[443])
#define XtCInitialState   ((char*)&XtShellStrings[456])
#define XtNgeometry       ((char*)&XtShellStrings[469])
#define XtCGeometry       ((char*)&XtShellStrings[478])
#define XtNbaseWidth      ((char*)&XtShellStrings[487])
#define XtCBaseWidth      ((char*)&XtShellStrings[497])
#define XtNbaseHeight     ((char*)&XtShellStrings[507])
#define XtCBaseHeight     ((char*)&XtShellStrings[518])
#define XtCIconWindow     ((char*)&XtShellStrings[51])
#define XtNwinGravity      ((char*)&XtShellStrings[529])
#define XtCwinGravity      ((char*)&XtShellStrings[540])
#define XtNminWidth       ((char*)&XtShellStrings[551])
#define XtCminWidth       ((char*)&XtShellStrings[560])
#define XtNminHeight      ((char*)&XtShellStrings[569])
#define XtCminHeight      ((char*)&XtShellStrings[579])
#define XtNmaxWidth       ((char*)&XtShellStrings[589])
#define XtCmaxWidth       ((char*)&XtShellStrings[598])
#define XtNmaxHeight      ((char*)&XtShellStrings[607])
#define XtCmaxHeight      ((char*)&XtShellStrings[617])
#define XtNwidthInc       ((char*)&XtShellStrings[627])
#define XtNiconMask       ((char*)&XtShellStrings[62])
#define XtCwidthInc       ((char*)&XtShellStrings[636])
#define XtNheightInc      ((char*)&XtShellStrings[645])
#define XtCheightInc      ((char*)&XtShellStrings[655])
#define XtNminAspectY     ((char*)&XtShellStrings[665])
#define XtCminAspectY     ((char*)&XtShellStrings[676])
#define XtNmaxAspectY     ((char*)&XtShellStrings[687])
#define XtCmaxAspectY     ((char*)&XtShellStrings[698])
#define XtNminAspectX     ((char*)&XtShellStrings[709])
#define XtCiconMask       ((char*)&XtShellStrings[71])
#define XtCminAspectX     ((char*)&XtShellStrings[720])
#define XtNmaxAspectX     ((char*)&XtShellStrings[731])
#define XtCmaxAspectX     ((char*)&XtShellStrings[742])
#define XtNwmTimeout      ((char*)&XtShellStrings[753])
#define XtCwmTimeout      ((char*)&XtShellStrings[763])
#define XtNwaitForWm      ((char*)&XtShellStrings[773])
#define XtCwaitForWm      ((char*)&XtShellStrings[783])
#define XtNwaitforwm      ((char*)&XtShellStrings[793])
#define XtCwaitforwm      ((char*)&XtShellStrings[803])
#define XtNwindowGroup    ((char*)&XtShellStrings[80])
#define XtNclientLeader   ((char*)&XtShellStrings[813])
#define XtCclientLeader   ((char*)&XtShellStrings[826])
#define XtNwindowRole     ((char*)&XtShellStrings[839])
#define XtCwindowRole     ((char*)&XtShellStrings[850])
#define XtNurgency        ((char*)&XtShellStrings[861])
#define XtCurgency        ((char*)&XtShellStrings[869])
#define XtNcancelCallback  ((char*)&XtShellStrings[877])
#define XtNcloneCommand   ((char*)&XtShellStrings[892])
#define XtCcloneCommand   ((char*)&XtShellStrings[905])
#define XtNconnection     ((char*)&XtShellStrings[918])
#define XtCconnection     ((char*)&XtShellStrings[929])
#define XtCwindowGroup    ((char*)&XtShellStrings[92])
#define XtNcurrentDirectory ((char*)&XtShellStrings[940])
#define XtCcurrentDirectory ((char*)&XtShellStrings[957])
#define XtNdieCallback     ((char*)&XtShellStrings[974])
#define XtNdiscardCommand  ((char*)&XtShellStrings[986])
#define XtCiconName       ((char*)&XtShellStrings[9])
#define _XtShell_h        1

extern char XtShellStrings[];
extern WidgetClass applicationShellWidgetClass;

```

```
extern WidgetClass overrideShellWidgetClass;
extern WidgetClass sessionShellWidgetClass;
extern WidgetClass shellWidgetClass;
extern WidgetClass topLevelShellWidgetClass;
extern WidgetClass transientShellWidgetClass;
extern WidgetClass wmShellWidgetClass;
```

6.10.17 X11/ShellP.h

```
struct _OldXSizeHints {
    long int flags;
    int x;
    int y;
    int width;
    int height;
    int min_width;
    int min_height;
    int max_width;
    int max_height;
    int width_inc;
    int height_inc;
    struct {
        int x;
        int y;
    } min_aspect;
    struct {
        int x;
        int y;
    } max_aspect;
};

#define _XtShellPositionValid ((Boolean)(1<<0))
#define _XtShellNotReparented ((Boolean)(1<<1))
#define _XtShellPPositionOK ((Boolean)(1<<2))
#define _XtShellGeometryParsed ((Boolean)(1<<3))
#define XtInheritRootGeometryManager
((XtGeometryHandler) XtInherit)
#define XtShellExtensionVersion 1L

typedef struct {
    XtPointer extension;
} ShellClassPart;
typedef struct {
    XtPointer next_extension;
    XrmQuark record_type;
    long int version;
    Cardinal record_size;
    XtGeometryHandler root_geometry_manager;
} ShellClassExtensionRec;
typedef ShellClassExtensionRec *ShellClassExtension;
typedef struct _ShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
} ShellClassRec;
typedef struct {
    char *geometry;
    XtCreatePopupChildProc create_popup_child_proc;
    XtGrabKind grab_kind;
    Boolean spring_loaded;
    Boolean popped_up;
    Boolean allow_shell_resize;
    Boolean client_specified;
    Boolean save_under;
    Boolean override_redirect;
```

```

        XtCallbackList popup_callback;
        XtCallbackList popdown_callback;
        Visual *visual;
    } ShellPart;
typedef struct {
    CorePart core;
    CompositePart composite;
    ShellPart shell;
} ShellRec;
typedef ShellRec *ShellWidget;
typedef struct {
    XtPointer extension;
} OverrideShellClassPart;
typedef struct _OverrideShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
    OverrideShellClassPart override_shell_class;
} OverrideShellClassRec;
typedef struct {
    int frabjous;
} OverrideShellPart;
typedef struct {
    CorePart core;
    CompositePart composite;
    ShellPart shell;
    OverrideShellPart override;
} OverrideShellRec;
typedef OverrideShellRec *OverrideShellWidget;
typedef struct {
    XtPointer extension;
} WMShellClassPart;
typedef struct _WMShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
    WMShellClassPart wm_shell_class;
} WMShellClassRec;
typedef struct {
    char *title;
    int wm_timeout;
    Boolean wait_for_wm;
    Boolean transient;
    Boolean urgency;
    Widget client_leader;
    String window_role;
    struct _OldXSizeHints size_hints;
    XWMHints wm_hints;
    int base_width;
    int base_height;
    int win_gravity;
    Atom title_encoding;
} WMShellPart;
typedef struct {
    CorePart core;
    CompositePart composite;
    ShellPart shell;
    WMShellPart wm;
} WMShellRec;
typedef WMShellRec *WMShellWidget;
typedef struct {
    XtPointer extension;
} TransientShellClassPart;
typedef struct _TransientShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;

```

```

    ShellClassPart shell_class;
    WMShellClassPart wm_shell_class;
    VendorShellClassPart vendor_shell_class;
    TransientShellClassPart transient_shell_class;
} TransientShellClassRec;
typedef struct {
    Widget transient_for;
} TransientShellPart;
typedef struct {
    CorePart core;
    CompositePart composite;
    ShellPart shell;
    WMShellPart wm;
    VendorShellPart vendor;
    TransientShellPart transient;
} TransientShellRec;
typedef TransientShellRec *TransientShellWidget;
typedef struct {
    XtPointer extension;
} TopLevelShellClassPart;
typedef struct _TopLevelShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
    WMShellClassPart wm_shell_class;
    VendorShellClassPart vendor_shell_class;
    TopLevelShellClassPart top_level_shell_class;
} TopLevelShellClassRec;
typedef struct {
    char *icon_name;
    Boolean iconic;
    Atom icon_name_encoding;
} TopLevelShellPart;
typedef struct {
    CorePart core;
    CompositePart composite;
    ShellPart shell;
    WMShellPart wm;
    VendorShellPart vendor;
    TopLevelShellPart topLevel;
} TopLevelShellRec;
typedef TopLevelShellRec *TopLevelShellWidget;
typedef struct {
    XtPointer extension;
} ApplicationShellClassPart;
typedef struct _ApplicationShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
    WMShellClassPart wm_shell_class;
    VendorShellClassPart vendor_shell_class;
    TopLevelShellClassPart top_level_shell_class;
    ApplicationShellClassPart application_shell_class;
} ApplicationShellClassRec;
typedef struct {
#ifdef __cplusplus || defined(c_plusplus)
    char *c_class;
#else
    char *class;
#endif
    XrmClass xrm_class;
    int argc;
    char **argv;
} ApplicationShellPart;
typedef struct {
    CorePart core;

```

```

        CompositePart composite;
        ShellPart shell;
        WMShellPart wm;
        VendorShellPart vendor;
        TopLevelShellPart topLevel;
        ApplicationShellPart application;
    } ApplicationShellRec;
typedef ApplicationShellRec *ApplicationShellWidget;
typedef struct {
    XtPointer extension;
} SessionShellClassPart;
typedef struct _SessionShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
    WMShellClassPart wm_shell_class;
    VendorShellClassPart vendor_shell_class;
    TopLevelShellClassPart top_level_shell_class;
    ApplicationShellClassPart application_shell_class;
    SessionShellClassPart session_shell_class;
} SessionShellClassRec;
typedef struct _XtSaveYourselfRec *XtSaveYourself;
typedef struct {
    SmcConn connection;
    String session_id;
    char **restart_command;
    char **clone_command;
    char **discard_command;
    char **resign_command;
    char **shutdown_command;
    char **environment;
    String current_dir;
    String program_path;
    unsigned char restart_style;
    unsigned char checkpoint_state;
    Boolean join_session;
    XtCallbackList save_callbacks;
    XtCallbackList interact_callbacks;
    XtCallbackList cancel_callbacks;
    XtCallbackList save_complete_callbacks;
    XtCallbackList die_callbacks;
    XtCallbackList error_callbacks;
    XtSaveYourself save;
    XtInputId input_id;
    XtPointer ses20;
    XtPointer ses19;
    XtPointer ses18;
    XtPointer ses17;
    XtPointer ses16;
    XtPointer ses15;
    XtPointer ses14;
    XtPointer ses13;
    XtPointer ses12;
    XtPointer ses11;
    XtPointer ses10;
    XtPointer ses9;
    XtPointer ses8;
    XtPointer ses7;
    XtPointer ses6;
    XtPointer ses5;
    XtPointer ses4;
    XtPointer ses3;
    XtPointer ses2;
    XtPointer ses1;
} SessionShellPart;
typedef struct {

```

```

CorePart core;
CompositePart composite;
ShellPart shell;
WMShellPart wm;
VendorShellPart vendor;
TopLevelShellPart topLevel;
ApplicationShellPart application;
SessionShellPart session;
} SessionShellRec;
typedef SessionShellRec *SessionShellWidget;
extern ApplicationShellClassRec applicationShellClassRec;
extern OverrideShellClassRec overrideShellClassRec;
extern SessionShellClassRec sessionShellClassRec;
extern ShellClassRec shellClassRec;
extern TopLevelShellClassRec topLevelShellClassRec;
extern TransientShellClassRec transientShellClassRec;
extern WMShellClassRec wmShellClassRec;

```

6.10.18 X11/StringDefs.h

```

#define XtNaccelerators ((char*)&XtStrings[0])
#define XtCFile ((char*)&XtStrings[1012])
#define XtCFont ((char*)&XtStrings[1017])
#define XtCForeground ((char*)&XtStrings[1022])
#define XtCFraction ((char*)&XtStrings[1033])
#define XtCFunction ((char*)&XtStrings[1042])
#define XtCHeight ((char*)&XtStrings[1051])
#define XtCHSpace ((char*)&XtStrings[1058])
#define XtCIndex ((char*)&XtStrings[1065])
#define XtCInitialResourcesPersistent ((char*)&XtStrings[1071])
#define XtCInsertPosition ((char*)&XtStrings[1098])
#define XtCInterval ((char*)&XtStrings[1113])
#define XtNborderPixmap ((char*)&XtStrings[111])
#define XtCJustify ((char*)&XtStrings[1122])
#define XtCKnobIndent ((char*)&XtStrings[1130])
#define XtCKnobPixel ((char*)&XtStrings[1141])
#define XtCLabel ((char*)&XtStrings[1151])
#define XtCLength ((char*)&XtStrings[1157])
#define XtCMappedWhenManaged ((char*)&XtStrings[1164])
#define XtCMargin ((char*)&XtStrings[1182])
#define XtCMenuEntry ((char*)&XtStrings[1189])
#define XtCNotify ((char*)&XtStrings[1199])
#define XtCOrientation ((char*)&XtStrings[1206])
#define XtCParameter ((char*)&XtStrings[1218])
#define XtCPixmap ((char*)&XtStrings[1228])
#define XtCPosition ((char*)&XtStrings[1235])
#define XtCReadOnly ((char*)&XtStrings[1244])
#define XtNborderWidth ((char*)&XtStrings[124])
#define XtCResize ((char*)&XtStrings[1253])
#define XtCReverseVideo ((char*)&XtStrings[1260])
#define XtCScreen ((char*)&XtStrings[1273])
#define XtCScrollProc ((char*)&XtStrings[1280])
#define XtCScrollDCursor ((char*)&XtStrings[1291])
#define XtCScrollHCursor ((char*)&XtStrings[1305])
#define XtCScrollLCursor ((char*)&XtStrings[1319])
#define XtCScrollRCursor ((char*)&XtStrings[1333])
#define XtCScrollUCursor ((char*)&XtStrings[1347])
#define XtCScrollVCursor ((char*)&XtStrings[1361])
#define XtNcallback ((char*)&XtStrings[136])
#define XtCSelection ((char*)&XtStrings[1375])
#define XtCSensitive ((char*)&XtStrings[1385])
#define XtCSelectionArray ((char*)&XtStrings[1395])
#define XtNallowHoriz ((char*)&XtStrings[13])
#define XtCSpace ((char*)&XtStrings[1410])
#define XtCString ((char*)&XtStrings[1416])

```



```

#define XtCTextOptions ((char*)&XtStrings[1423])
#define XtCTextPosition ((char*)&XtStrings[1435])
#define XtCTextSink ((char*)&XtStrings[1448])
#define XtCTextSource ((char*)&XtStrings[1457])
#define XtNchildren ((char*)&XtStrings[145])
#define XtCThickness ((char*)&XtStrings[1468])
#define XtCThumb ((char*)&XtStrings[1478])
#define XtCTranslations ((char*)&XtStrings[1484])
#define XtCValue ((char*)&XtStrings[1497])
#define XtCVSpace ((char*)&XtStrings[1503])
#define XtCWidth ((char*)&XtStrings[1510])
#define XtCWindow ((char*)&XtStrings[1516])
#define XtCX ((char*)&XtStrings[1523])
#define XtCY ((char*)&XtStrings[1525])
#define XtRAcceleratorTable ((char*)&XtStrings[1527])
#define XtRAtom ((char*)&XtStrings[1544])
#define XtRBitmap ((char*)&XtStrings[1549])
#define XtNcolormap ((char*)&XtStrings[154])
#define XtRBool ((char*)&XtStrings[1556])
#define XtRBoolean ((char*)&XtStrings[1561])
#define XtRCallback ((char*)&XtStrings[1569])
#define XtRCallProc ((char*)&XtStrings[1578])
#define XtRCardinal ((char*)&XtStrings[1587])
#define XtRColor ((char*)&XtStrings[1596])
#define XtRColormap ((char*)&XtStrings[1602])
#define XtRCursor ((char*)&XtStrings[1611])
#define XtRDimension ((char*)&XtStrings[1618])
#define XtRDisplay ((char*)&XtStrings[1628])
#define XtREditMode ((char*)&XtStrings[1636])
#define XtNdepth ((char*)&XtStrings[163])
#define XtREnum ((char*)&XtStrings[1645])
#define XtRFile ((char*)&XtStrings[1650])
#define XtRFloat ((char*)&XtStrings[1655])
#define XtRFont ((char*)&XtStrings[1661])
#define XtRFontStruct ((char*)&XtStrings[1666])
#define XtRFunction ((char*)&XtStrings[1677])
#define XtRGeometry ((char*)&XtStrings[1686])
#define XtRImmediate ((char*)&XtStrings[1695])
#define XtNdestroyCallback ((char*)&XtStrings[169])
#define XtRInitialState ((char*)&XtStrings[1705])
#define XtRInt ((char*)&XtStrings[1718])
#define XtRJustify ((char*)&XtStrings[1722])
#define XtRLongBoolean ((char*)&XtStrings[1730])
#define XtRObject ((char*)&XtStrings[1735])
#define XtROrientation ((char*)&XtStrings[1742])
#define XtRPixel ((char*)&XtStrings[1754])
#define XtRPixmap ((char*)&XtStrings[1760])
#define XtRPointer ((char*)&XtStrings[1767])
#define XtRPosition ((char*)&XtStrings[1775])
#define XtRScreen ((char*)&XtStrings[1784])
#define XtRShort ((char*)&XtStrings[1791])
#define XtRString ((char*)&XtStrings[1797])
#define XtRStringArray ((char*)&XtStrings[1804])
#define XtRStringTable ((char*)&XtStrings[1816])
#define XtRUnsignedChar ((char*)&XtStrings[1828])
#define XtRTranslationTable ((char*)&XtStrings[1841])
#define XtRVisual ((char*)&XtStrings[1858])
#define XtNeditType ((char*)&XtStrings[185])
#define XtRWidget ((char*)&XtStrings[1865])
#define XtRWidgetClass ((char*)&XtStrings[1872])
#define XtRWidgetList ((char*)&XtStrings[1884])
#define XtRWindow ((char*)&XtStrings[1895])
#define XtEoff ((char*)&XtStrings[1902])
#define XtEfalse ((char*)&XtStrings[1906])
#define XtEno ((char*)&XtStrings[1912])
#define XtEon ((char*)&XtStrings[1915])

```



```

#define XtEtrue ((char*)&XtStrings[1918])
#define XtEyes ((char*)&XtStrings[1923])
#define XtEvertical ((char*)&XtStrings[1927])
#define XtEhorizontal ((char*)&XtStrings[1936])
#define XtEtextRead ((char*)&XtStrings[1947])
#define XtNfile ((char*)&XtStrings[194])
#define XtEtextAppend ((char*)&XtStrings[1952])
#define XtEtextEdit ((char*)&XtStrings[1959])
#define XtExtdefaultbackground ((char*)&XtStrings[1964])
#define XtExtdefaultforeground ((char*)&XtStrings[1984])
#define XtNfont ((char*)&XtStrings[199])
#define XtExtdefaultfont ((char*)&XtStrings[2004])
#define XtNfontSet ((char*)&XtStrings[2018])
#define XtRFontSet ((char*)&XtStrings[2026])
#define XtCFontSet ((char*)&XtStrings[2034])
#define XtRGravity ((char*)&XtStrings[2042])
#define XtNforceBars ((char*)&XtStrings[204])
#define XtNcreateHook ((char*)&XtStrings[2050])
#define XtNchangeHook ((char*)&XtStrings[2061])
#define XtNconfigureHook ((char*)&XtStrings[2072])
#define XtNgeometryHook ((char*)&XtStrings[2086])
#define XtNdestroyHook ((char*)&XtStrings[2099])
#define XtNshells ((char*)&XtStrings[2111])
#define XtNnumShells ((char*)&XtStrings[2118])
#define XtRCommandArgArray ((char*)&XtStrings[2128])
#define XtRDirectoryString ((char*)&XtStrings[2144])
#define XtNforeground ((char*)&XtStrings[214])
#define XtREnvironmentArray ((char*)&XtStrings[2160])
#define XtRRestartStyle ((char*)&XtStrings[2177])
#define XtRSmcConn ((char*)&XtStrings[2190])
#define XtHcreate ((char*)&XtStrings[2198])
#define XtHsetValues ((char*)&XtStrings[2207])
#define XtHmanageChildren ((char*)&XtStrings[2219])
#define XtHunmanageChildren ((char*)&XtStrings[2236])
#define XtHmanageSet ((char*)&XtStrings[2255])
#define XtNfunction ((char*)&XtStrings[225])
#define XtHunmanageSet ((char*)&XtStrings[2267])
#define XtHrealizeWidget ((char*)&XtStrings[2281])
#define XtHunrealizeWidget ((char*)&XtStrings[2297])
#define XtHaddCallback ((char*)&XtStrings[2315])
#define XtHaddCallbacks ((char*)&XtStrings[2329])
#define XtHremoveCallback ((char*)&XtStrings[2344])
#define XtNheight ((char*)&XtStrings[234])
#define XtHremoveCallbacks ((char*)&XtStrings[2361])
#define XtHremoveAllCallbacks ((char*)&XtStrings[2379])
#define XtHaugmentTranslations ((char*)&XtStrings[2400])
#define XtNhighlight ((char*)&XtStrings[241])
#define XtHoverrideTranslations ((char*)&XtStrings[2422])
#define XtHuninstallTranslations ((char*)&XtStrings[2445])
#define XtHsetKeyboardFocus ((char*)&XtStrings[2469])
#define XtHsetWMColormapWindows ((char*)&XtStrings[2488])
#define XtNallowVert ((char*)&XtStrings[24])
#define XtHsetMappedWhenManaged ((char*)&XtStrings[2511])
#define XtNhSpace ((char*)&XtStrings[251])
#define XtHmapWidget ((char*)&XtStrings[2534])
#define XtHunmapWidget ((char*)&XtStrings[2546])
#define XtHpopup ((char*)&XtStrings[2560])
#define XtHpopupSpringLoaded ((char*)&XtStrings[2568])
#define XtHpopdown ((char*)&XtStrings[2588])
#define XtNindex ((char*)&XtStrings[258])
#define XtHconfigure ((char*)&XtStrings[2598])
#define XtHpreGeometry ((char*)&XtStrings[2610])
#define XtHpostGeometry ((char*)&XtStrings[2624])
#define XtHdestroy ((char*)&XtStrings[2639])
#define XtNinitialResourcesPersistent ((char*)&XtStrings[264])
#define XtNinnerHeight ((char*)&XtStrings[291])

```

```

#define XtNinnerWidth ((char*)&XtStrings[303])
#define XtNinnerWindow ((char*)&XtStrings[314])
#define XtNinsertPosition ((char*)&XtStrings[326])
#define XtNinternalHeight ((char*)&XtStrings[341])
#define XtNancestorSensitive ((char*)&XtStrings[34])
#define XtNinternalWidth ((char*)&XtStrings[356])
#define XtNjumpProc ((char*)&XtStrings[370])
#define XtNjustify ((char*)&XtStrings[379])
#define XtNknobHeight ((char*)&XtStrings[387])
#define XtNknobIndent ((char*)&XtStrings[398])
#define XtNknobPixel ((char*)&XtStrings[409])
#define XtNknobWidth ((char*)&XtStrings[419])
#define XtNlabel ((char*)&XtStrings[429])
#define XtNlength ((char*)&XtStrings[435])
#define XtNlowerRight ((char*)&XtStrings[442])
#define XtNmappedWhenManaged ((char*)&XtStrings[453])
#define XtNmenuItem ((char*)&XtStrings[471])
#define XtNname ((char*)&XtStrings[481])
#define XtNnotify ((char*)&XtStrings[486])
#define XtNnumChildren ((char*)&XtStrings[493])
#define XtNorientation ((char*)&XtStrings[505])
#define XtNparameter ((char*)&XtStrings[517])
#define XtNpixmap ((char*)&XtStrings[527])
#define XtNbackground ((char*)&XtStrings[52])
#define XtNpopupCallback ((char*)&XtStrings[534])
#define XtNpopdownCallback ((char*)&XtStrings[548])
#define XtNresize ((char*)&XtStrings[564])
#define XtNreverseVideo ((char*)&XtStrings[571])
#define XtNscreen ((char*)&XtStrings[584])
#define XtNscrollProc ((char*)&XtStrings[591])
#define XtNscrollDCursor ((char*)&XtStrings[602])
#define XtNscrollHCursor ((char*)&XtStrings[616])
#define XtNscrollLCursor ((char*)&XtStrings[630])
#define XtNbackgroundPixmap ((char*)&XtStrings[63])
#define XtNscrollRCursor ((char*)&XtStrings[644])
#define XtNscrollUCursor ((char*)&XtStrings[658])
#define XtNscrollVCursor ((char*)&XtStrings[672])
#define XtNselection ((char*)&XtStrings[686])
#define XtNselectionArray ((char*)&XtStrings[696])
#define XtNsensitive ((char*)&XtStrings[711])
#define XtNshown ((char*)&XtStrings[721])
#define XtNspace ((char*)&XtStrings[727])
#define XtNstring ((char*)&XtStrings[733])
#define XtNtextOptions ((char*)&XtStrings[740])
#define XtNtextSink ((char*)&XtStrings[752])
#define XtNtextSource ((char*)&XtStrings[761])
#define XtNthickness ((char*)&XtStrings[772])
#define XtNthumb ((char*)&XtStrings[782])
#define XtNthumbProc ((char*)&XtStrings[788])
#define XtNtop ((char*)&XtStrings[798])
#define XtNtranslations ((char*)&XtStrings[802])
#define XtNbitmap ((char*)&XtStrings[80])
#define XtNunrealizeCallback ((char*)&XtStrings[815])
#define XtNupdate ((char*)&XtStrings[833])
#define XtNuseBottom ((char*)&XtStrings[840])
#define XtNuseRight ((char*)&XtStrings[850])
#define XtNvalue ((char*)&XtStrings[859])
#define XtNvSpace ((char*)&XtStrings[865])
#define XtNwidth ((char*)&XtStrings[872])
#define XtNwindow ((char*)&XtStrings[878])
#define XtNborderColor ((char*)&XtStrings[87])
#define XtNx ((char*)&XtStrings[885])
#define XtNy ((char*)&XtStrings[887])
#define XtCAccelerators ((char*)&XtStrings[889])
#define XtCbackground ((char*)&XtStrings[902])
#define XtCbitmap ((char*)&XtStrings[913])

```

```

#define XtCBoolean      ((char*)&XtStrings[920])
#define XtCBorderColor  ((char*)&XtStrings[928])
#define XtCBorderWidth ((char*)&XtStrings[940])
#define XtCCallback     ((char*)&XtStrings[952])
#define XtCColorMap     ((char*)&XtStrings[961])
#define XtCColor        ((char*)&XtStrings[970])
#define XtCCursor       ((char*)&XtStrings[976])
#define XtCDepth        ((char*)&XtStrings[983])
#define XtCEditType     ((char*)&XtStrings[989])
#define XtCEventBindings ((char*)&XtStrings[998])
#define XtNborder       ((char*)&XtStrings[99])
#define _XtStringDefs_h_ 1

extern char XtStrings[];

```

6.10.19 X11/Translatel.h

```

struct _TranslationData {
    unsigned char hasBindings;
    unsigned char operation;
    TMSHORTCARD numStateTrees;
    struct _TranslationData *composers[2];
    EventMask eventMask;
    TMStateTree stateTreeTbl[1];
};

typedef struct _ActionsRec *ActionPtr;
typedef struct _ActionsRec {
    int idx;
    String *params;
    Cardinal num_params;
    ActionPtr next;
} ActionRec;

typedef long unsigned int TMLongCard;
struct _LateBindings {
    unsigned int knot:1;
    unsigned int pair:1;
    short unsigned int ref_count;
    KeySym keysym;
};

typedef struct _LateBindings *LateBindingsPtr;
struct _TMTypeMatchRec {
    TMLongCard eventType;
    TMLongCard eventCode;
    TMLongCard eventCodeMask;
    MatchProc matchEvent;
};

typedef struct _TMTypeMatchRec *TMTypeMatch;
struct _TMModifierMatchRec {
    TMLongCard modifiers;
    TMLongCard modifierMask;
    LateBindingsPtr lateModifiers;
    Boolean standard;
};

typedef struct _TMModifierMatchRec *TMModifierMatch;
typedef struct _TMEventRec *TMEventPtr;
typedef Boolean(*MatchProc) (TMTypeMatch, TMModifierMatch,
TMEventPtr);
struct _TMEventRec {
    XEvent *xev;
    Event event;
};

typedef struct _EventRec {
    TMLongCard modifiers;
    TMLongCard modifierMask;
    LateBindingsPtr lateModifiers;

```

```

        TMLongCard eventType;
        TMLongCard eventCode;
        TMLongCard eventCodeMask;
        MatchProc matchEvent;
        Boolean standard;
    } Event;
typedef short unsigned int TMShortCard;
typedef struct _TMBranchHeadRec {
    unsigned int isSimple:1;
    unsigned int hasActions:1;
    unsigned int hasCycles:1;
    unsigned int more:13;
    TMShortCard typeIndex;
    TMShortCard modIndex;
} TMBranchHeadRec;
typedef struct _TMSimpleStateTreeRec {
    unsigned int isSimple:1;
    unsigned int isAccelerator:1;
    unsigned int mappingNotifyInterest:1;
    unsigned int refCount:13;
    TMShortCard numBranchHeads;
    TMShortCard numQuarks;
    TMShortCard unused;
    TMBranchHeadRec *branchHeadTbl;
    XrmQuark *quarkTbl;
} TMSimpleStateTreeRec;
struct _XtStateRec {
    unsigned int isCycleStart:1;
    unsigned int isCycleEnd:1;
    TMShortCard typeIndex;
    TMShortCard modIndex;
    ActionPtr actions;
    StatePtr nextLevel;
};
typedef struct _XtStateRec *StatePtr;
typedef struct _TMParseStateTreeRec {
    unsigned int isSimple:1;
    unsigned int isAccelerator:1;
    unsigned int mappingNotifyInterest:1;
    unsigned int isStackQuarks:1;
    unsigned int isStackBranchHeads:1;
    unsigned int isStackComplexBranchHeads:1;
    unsigned int unused:10;
    TMShortCard numBranchHeads;
    TMShortCard numQuarks;
    TMShortCard numComplexBranchHeads;
    TMBranchHeadRec *branchHeadTbl;
    XrmQuark *quarkTbl;
    StatePtr *complexBranchHeadTbl;
    TMShortCard branchHeadTblSize;
    TMShortCard quarkTblSize;
    TMShortCard complexBranchHeadTblSize;
    StatePtr head;
} TMParseStateTreeRec;
typedef struct _TMComplexStateTreeRec {
    unsigned int isSimple:1;
    unsigned int isAccelerator:1;
    unsigned int mappingNotifyInterest:1;
    unsigned int refCount:13;
    TMShortCard numBranchHeads;
    TMShortCard numQuarks;
    TMShortCard numComplexBranchHeads;
    TMBranchHeadRec *branchHeadTbl;
    XrmQuark *quarkTbl;
    StatePtr *complexBranchHeadTbl;
} TMComplexStateTreeRec;

```

```

union _TMStateTreeRec {
    TMSimpleStateTreeRec simple;
    TMParseStateTreeRec parse;
    TMComplexStateTreeRec complex;
};
typedef union _TMStateTreeRec *TMStateTree;
typedef Boolean(*_XtTraversalProc) (StatePtr, XtPointer);
typedef struct _TMSimpleBindProcsRec {
    XtActionProc *procs;
} TMSimpleBindProcsRec;
typedef struct _TMSimpleBindDataRec {
    unsigned int isComplex:1;
    TMSimpleBindProcsRec bindTbl[8];
} TMSimpleBindDataRec;
typedef struct _TMComplexBindProcsRec {
    Widget widget;
    XtTranslations aXlations;
    XtActionProc *procs;
} TMComplexBindProcsRec;
struct _ATranslationData {
    unsigned char hasBindings;
    unsigned char operation;
    struct _TranslationData *xlations;
    struct _ATranslationData *next;
    TMComplexBindProcsRec bindTbl[24];
};
typedef struct _TMComplexBindDataRec {
    unsigned int isComplex:1;
    struct _ATranslationData *accel_context;
    TMComplexBindProcsRec bindTbl[24];
} TMComplexBindDataRec;
union _TMBindDataRec {
    TMSimpleBindDataRec simple;
    TMComplexBindDataRec complex;
};
typedef union _TMBindDataRec *TMBindData;
typedef struct _TMParseStateTreeRec *TMParseStateTree;
struct _EventSeqRec {
    Event event;
    StatePtr state;
    EventSeqPtr next;
    ActionPtr actions;
};
typedef struct _EventSeqRec *EventSeqPtr;
typedef struct _TMBranchHeadRec *TMBranchHead;
typedef unsigned int _XtTranslateOp;

```

6.10.20 X11/VendorP.h

```

typedef struct {
    XtPointer extension;
} VendorShellClassPart;
typedef struct _VendorShellClassRec {
    CoreClassPart core_class;
    CompositeClassPart composite_class;
    ShellClassPart shell_class;
    WMShellClassPart wm_shell_class;
    VendorShellClassPart vendor_shell_class;
} VendorShellClassRec;
typedef struct {
    int vendor_specific;
} VendorShellPart;
typedef struct {
    CorePart core;
    CompositePart composite;
}

```

```

    ShellPart shell;
    WMShellPart wm;
    VendorShellPart vendor;
} VendorShellRec;
typedef VendorShellRec *VendorShellWidget;

```

6.10.21 Isb/Composite.h

```
typedef Cardinal(*XtOrderProc) (Widget);
```

6.10.22 Isb/Intrinsic.h

```

typedef unsigned int Cardinal;
typedef void *XtPointer;
struct _WidgetRec;
typedef struct _WidgetRec *Widget;
typedef char *String;
typedef unsigned int XtGeometryMask;
typedef short int Position;
typedef short unsigned int Dimension;
typedef struct {
    XtGeometryMask request_mode;
    Position x;
    Position y;
    Dimension width;
    Dimension height;
    Dimension border_width;
    Widget sibling;
    int stack_mode;
} XtWidgetGeometry;
struct _XtResource {
    String resource_name;
    String resource_class;
    String resource_type;
    Cardinal resource_size;
    Cardinal resource_offset;
    String default_type;
    XtPointer default_addr;
};
typedef struct _XtResource *XtResourceList;
typedef void (*XtActionProc) (Widget, XEvent *, String *, Cardinal
*);
struct _XtActionsRec {
    String string;
    XtActionProc proc;
};
typedef struct _XtActionsRec *XtActionList;
typedef char Boolean;
typedef unsigned char XtEnum;
typedef long unsigned int XtInputId;
struct _WidgetClassRec;
typedef Widget *WidgetList;
typedef enum {
    XtGeometryYes = 0,
    XtGeometryNo = 1,
    XtGeometryAlmost = 2,
    XtGeometryDone = 3
} XtGeometryResult;
typedef long int XtArgVal;
typedef struct {
    String name;
    XtArgVal value;
} Arg;

```

```

typedef Arg *ArgList;
typedef void (*XtCallbackProc) (Widget, XtPointer, XtPointer);
struct _XtCallbackRec {
    XtCallbackProc callback;
    XtPointer closure;
};
typedef struct _XtCallbackRec *XtCallbackList;
typedef long unsigned int EventMask;
typedef enum {
    XtGrabNone = 0,
    XtGrabNonexclusive = 1,
    XtGrabExclusive = 2
} XtGrabKind;
typedef long unsigned int XtValueMask;
typedef struct _XtEventRec *XtEventTable;
typedef void (*XtCreatePopupChildProc) (Widget);
typedef long unsigned int Pixel;
typedef XtActionProc *XtBoundActions;

typedef struct _WidgetClassRec *WidgetClass;

typedef struct _TranslationData *XtTranslations;

```

6.11 Interface Definitions for libXt

The interfaces defined on the following pages are included in libXt and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 6.9 shall behave as described in the referenced base document.

_XtCopyFromArg

Name

_XtCopyFromArg — Copy "size" bytes from src to dst.

Synopsis

```
void _XtCopyFromArg(XtArgVal src, char* dst, unsigned int size);
```

Description

The _XtCopyFromArg() function copies "size" bytes from src to dst. This is an internal X function call.

_XtInherit

Name

_XtInherit — inheritance operation.

Synopsis

```

#include <IntrinsicP.h>

extern void _XtInherit(
    #if NeedFunctionPrototypes
        void
    #endif

```

```
);
```

Description

`_XtInherit()` is a procedure that issues an error message if it is actually called.

`_XtInheritTranslations`

Name

`_XtInheritTranslations` — an inheritance constant

Synopsis

```
#include <CoreP.h>

externalref int _XtInheritTranslations;
```

Description

`_XtInheritTranslations` is an inheritance constant.

`_XtIsSubclassOf`

Name

`_XtIsSubclassOf` — determine if `Widget` is a subclass of `WidgetClass`.

Synopsis

```
#include <Intrinsic.h>

extern Boolean _XtIsSubclassOf(
    Widget          /* object* */,
    WidgetClass     /* widget_class */,
    WidgetClass     /* flag_class */,
    _XtXtEnum       /* type_flag */,
);
```

Description

Determine if `Widget(object)` is a subclass of `WidgetClass(widget_class)`.

6.12 Interfaces for libXext

Table 6-11 defines the library name and shared object name for the libXext library

Table 6-11 libXext Definition

Library:	libXext
SONAME:	libXext.so.6

The behavior of the interfaces in this library is specified by the following specifications:

- [LSB] This Specification
- [X-dbe] Double Buffer Extension Library
- [X-DPMS] X Display Power Management Signaling
- [X-evi] X Extended Visual Interface Extension
- [X-security] X Security Extension Specification

[X-shape] X Nonrectangular Window Shape Extension Library

[X-shm] The MIT Shared Memory Extension

[X-sync] X Synchronization Extension Library

6.12.1 libXext interfaces

6.12.1.1 Interfaces for libXext interfaces

An LSB conforming implementation shall provide the generic functions for libXext interfaces specified in Table 6-12, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-12 libXext - libXext interfaces Function Interfaces

XMissingExtension [LSB]	XSetExtensionErrorHandler [LSB]	XextAddDisplay [LSB]	XextCreateExtension [LSB]
XextDestroyExtension [LSB]	XextFindDisplay [LSB]	XextRemoveDisplay [LSB]	

6.12.2 X Shape Extension

6.12.2.1 Interfaces for X Shape Extension

An LSB conforming implementation shall provide the generic functions for X Shape Extension specified in Table 6-13, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-13 libXext - X Shape Extension Function Interfaces

XShapeCombineMask [X-shape]	XShapeCombineRectangles [X-shape]	XShapeCombineRegion [X-shape]	XShapeCombineShape [X-shape]
XShapeGetRectangles [X-shape]	XShapeInputSelected [X-shape]	XShapeOffsetShape [X-shape]	XShapeQueryExtension [X-shape]
XShapeQueryExtensions [X-shape]	XShapeQueryVersion [X-shape]	XShapeSelectInput [X-shape]	

6.12.3 X Display Power Management Signaling Extension

6.12.3.1 Interfaces for X Display Power Management Signaling Extension

An LSB conforming implementation shall provide the generic functions for X Display Power Management Signaling Extension specified in Table 6-14, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-14 libXext - X Display Power Management Signaling Extension Function Interfaces

DPMSCapable [X-DPMS]	DPMSDisable [X-DPMS]	DPMSEnable [X-DPMS]	DPMSForceLevel [X-DPMS]
DPMSGetTimeouts [X-DPMS]	DPMSGetVersion [X-DPMS]	DPMSInfo [X-DPMS]	DPMSQueryExtension [X-DPMS]

DPMSSetTimeouts [X-DPMS]			

6.12.4 X Shared Memory Extensions

6.12.4.1 Interfaces for X Shared Memory Extensions

An LSB conforming implementation shall provide the generic functions for X Shared Memory Extensions specified in Table 6-15, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-15 libXext - X Shared Memory Extensions Function Interfaces

XShmAttach [X-shm]	XShmCreateImage [X-shm]	XShmCreatePixmap [X-shm]	XShmDetach [X-shm]
XShmGetEventBase [X-shm]	XShmGetImage [X-shm]	XShmPixmapFormat [X-shm]	XShmPutImage [X-shm]
XShmQueryExtension [X-shm]	XShmQueryVersion [X-shm]		

6.12.5 X Synchronization Extension

6.12.5.1 Interfaces for X Synchronization Extension

An LSB conforming implementation shall provide the generic functions for X Synchronization Extension specified in Table 6-16, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-16 libXext - X Synchronization Extension Function Interfaces

XSyncAwait [X-sync]	XSyncChangeAlarm [X-sync]	XSyncChangeCounter [X-sync]	XSyncCreateAlarm [X-sync]
XSyncCreateCounter [X-sync]	XSyncDestroyAlarm [X-sync]	XSyncDestroyCounter [X-sync]	XSyncFreeSystemCounterList [X-sync]
XSyncGetPriority [X-sync]	XSyncInitialize [X-sync]	XSyncIntToValue [X-sync]	XSyncIntsToValue [X-sync]
XSyncListSystemCounters [X-sync]	XSyncMaxValue [X-sync]	XSyncMinValue [X-sync]	XSyncQueryAlarm [X-sync]
XSyncQueryCounter [X-sync]	XSyncQueryExtension [X-sync]	XSyncSetCounter [X-sync]	XSyncSetPriority [X-sync]
XSyncValueAdd [X-sync]	XSyncValueEqual [X-sync]	XSyncValueGreaterOrEqual [X-sync]	XSyncValueGreaterThan [X-sync]
XSyncValueHigh32 [X-sync]	XSyncValueIsNegative [X-sync]	XSyncValueIsPositive [X-sync]	XSyncValueIsZero [X-sync]
XSyncValueLessOrEqual [X-sync]	XSyncValueLessThan [X-sync]	XSyncValueLow32 [X-sync]	XSyncValueSubtract [X-sync]

6.12.6 X Security Extension

6.12.6.1 Interfaces for X Security Extension

An LSB conforming implementation shall provide the generic functions for X Security Extension specified in Table 6-17, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-17 libXext - X Security Extension Function Interfaces

XSecurityAllocXauth [X-security]	XSecurityFreeXauth [X-security]	XSecurityGenerateAuthorization [X-security]	XSecurityQueryExtension [X-security]
XSecurityRevokeAuthorization [X-security]			

6.12.7 X Double Buffer Extension

6.12.7.1 Interfaces for X Double Buffer Extension

An LSB conforming implementation shall provide the generic functions for X Double Buffer Extension specified in Table 6-18, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-18 libXext - X Double Buffer Extension Function Interfaces

XdbeAllocateBackBufferName [X-dbe]	XdbeBeginIdiom [X-dbe]	XdbeDeallocateBackBufferName [X-dbe]	XdbeEndIdiom [X-dbe]
XdbeFreeVisualInfo [X-dbe]	XdbeGetBackBufferAttributes [X-dbe]	XdbeGetVisualInfo [X-dbe]	XdbeQueryExtension [X-dbe]
XdbeSwapBuffers [X-dbe]			

6.12.8 X Extended Visual Interface Extension

6.12.8.1 Interfaces for X Extended Visual Interface Extension

An LSB conforming implementation shall provide the generic functions for X Extended Visual Interface Extension specified in Table 6-19, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-19 libXext - X Extended Visual Interface Extension Function Interfaces

XeviGetVisualInfo [X-evi]	XeviQueryExtension [X-evi]	XeviQueryVersion [X-evi]	

6.13 Data Definitions for libXext

This section defines global identifiers and their values that are associated with interfaces contained in libXext. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where

an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.13.1 X11/extensions/EVlproto.h

```
#define _EVIPROTO_H_
```

6.13.2 X11/extensions/XEVI.h

```
#define XEVI_TRANSPARENCY_NONE 0
#define X_EVIQueryVersion 0
#define XEVI_TRANSPARENCY_PIXEL 1
#define X_EVIGetVisualInfo 1
#define XEVI_TRANSPARENCY_MASK 2

typedef struct {
    VisualID core_visual_id;
    int screen;
    int level;
    unsigned int transparency_type;
    unsigned int transparency_value;
    unsigned int min_hw_colormaps;
    unsigned int max_hw_colormaps;
    unsigned int num_colormap_conflicts;
    VisualID *colormap_conflicts;
} ExtendedVisualInfo;
extern int XeviGetVisualInfo(Display *, VisualID *, int,
                             ExtendedVisualInfo *, int *);
extern int XeviQueryExtension(Display *);
extern int XeviQueryVersion(Display *, int *, int *);
```

6.13.3 X11/extensions/XEVlstr.h

```
#define XEVI_MINOR_VERSION 0
#define XEVI_MAJOR_VERSION 1
#define _EVISTR_H_ 1
#define sz_xExtendedVisualInfo 16
#define sz_xEVIGetVisualInfoReply 32
#define sz_xEVIQueryVersionReply 32
#define sz_VisualID32 4
#define sz_xEVIQueryVersionReq 4
#define sz_xEVIGetVisualInfoReq 8
#define EVINAME "Extended-Visual-Information"

typedef unsigned int VisualID32;
typedef struct _xExtendedVisualInfo {
    CARD32 core_visual_id;
    INT8 screen;
    INT8 level;
    CARD8 transparency_type;
    CARD8 pad0;
```

```

        CARD32 transparency_value;
        CARD8 min_hw_colormaps;
        CARD8 max_hw_colormaps;
        CARD16 num_colormap_conflicts;
    } xExtendedVisualInfo;
typedef struct _XEVIQueryVersion {
    CARD8 reqType;
    CARD8 xeviReqType;
    CARD16 length;
} xEVIQueryVersionReq;
typedef struct {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 majorVersion;
    CARD16 minorVersion;
    CARD32 pad0;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xEVIQueryVersionReply;
typedef struct _XEVIGetVisualInfoReq {
    CARD8 reqType;
    CARD8 xeviReqType;
    CARD16 length;
    CARD32 n_visual;
} xEVIGetVisualInfoReq;
typedef struct _XEVIGetVisualInfoReply {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 n_info;
    CARD32 n_conflicts;
    CARD32 pad0;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
} xEVIGetVisualInfoReply;

```

6.13.4 X11/extensions/XKBbells.h

```

#define _XKBBELLS_H_
#define XkbBI_Info 0
#define XkbBI_Warning 1
#define XkbBI_Failure 10
#define XkbBI_Wait 11
#define XkbBI_Proceed 12
#define XkbBI_Ignore 13
#define XkbBI_Iconify 14
#define XkbBI_Deiconify 15
#define XkbBI_Open 16
#define XkbBI_Close 17
#define XkbBI_TerminalBell 18
#define XkbBI_MarginBell 19
#define XkbBI_MinorError 2
#define XkbBI_CursorStuck 20
#define XkbBI_NewMail 21
#define XkbBI_LaunchApp 22
#define XkbBI_AppDeath 23
#define XkbBI_ImAlive 24
#define XkbBI_ClockChimeHour 25
#define XkbBI_ClockChimeHalf 26

```

```

#define XkbBI_ClockChimeQuarter 27
#define XkbBI_RepeatingLastBell 28
#define XkbBI_ComposeFail 29
#define XkbBI_MajorError 3
#define XkbBI_AX_FeatureOn 30
#define XkbBI_AX_FeatureOff 31
#define XkbBI_AX_FeatureChange 32
#define XkbBI_AX_IndicatorOn 33
#define XkbBI_AX_IndicatorOff 34
#define XkbBI_AX_IndicatorChange 35
#define XkbBI_AX_SlowKeysWarning 36
#define XkbBI_AX_SlowKeyPress 37
#define XkbBI_AX_SlowKeyAccept 38
#define XkbBI_AX_SlowKeyReject 39
#define XkbBI_BadValue 4
#define XkbBI_AX_SlowKeyRelease 40
#define XkbBI_AX_BounceKeyReject 41
#define XkbBI_AX_StickyLatch 42
#define XkbBI_AX_StickyLock 43
#define XkbBI_AX_StickyUnlock 44
#define XkbBI_NumBells 45
#define XkbBI_InvalidLocation 5
#define XkbBI_Question 6
#define XkbBI_Start 7
#define XkbBI_End 8
#define XkbBI_Success 9
#define XkbBN_AppDeath "AppDeath"
#define XkbBN_AX_BounceKeyReject "AX_BounceKeyReject"
#define XkbBN_AX_FeatureChange "AX_FeatureChange"
#define XkbBN_AX_FeatureOff "AX_FeatureOff"
#define XkbBN_AX_FeatureOn "AX_FeatureOn"
#define XkbBN_AX_IndicatorChange "AX_IndicatorChange"
#define XkbBN_AX_IndicatorOff "AX_IndicatorOff"
#define XkbBN_AX_IndicatorOn "AX_IndicatorOn"
#define XkbBN_AX_SlowKeyAccept "AX_SlowKeyAccept"
#define XkbBN_AX_SlowKeyPress "AX_SlowKeyPress"
#define XkbBN_AX_SlowKeyReject "AX_SlowKeyReject"
#define XkbBN_AX_SlowKeyRelease "AX_SlowKeyRelease"
#define XkbBN_AX_SlowKeysWarning "AX_SlowKeysWarning"
#define XkbBN_AX_StickyLatch "AX_StickyLatch"
#define XkbBN_AX_StickyLock "AX_StickyLock"
#define XkbBN_AX_StickyUnlock "AX_StickyUnlock"
#define XkbBN_BadValue "BadValue"
#define XkbBN_ClockChimeHalf "ClockChimeHalf"
#define XkbBN_ClockChimeHour "ClockChimeHour"
#define XkbBN_ClockChimeQuarter "ClockChimeQuarter"
#define XkbBN_Close "Close"
#define XkbBN_ComposeFail "ComposeFail"
#define XkbBN_CursorStuck "CursorStuck"
#define XkbBN_Deiconify "Deiconify"
#define XkbBN_End "End"
#define XkbBN_Failure "Failure"
#define XkbBN_Iconify "Iconify"
#define XkbBN_Ignore "Ignore"
#define XkbBN_ImAlive "ImAlive"
#define XkbBN_Info "Info"
#define XkbBN_InvalidLocation "InvalidLocation"
#define XkbBN_LaunchApp "LaunchApp"
#define XkbBN_MajorError "MajorError"
#define XkbBN_MarginBell "MarginBell"
#define XkbBN_MinorError "MinorError"
#define XkbBN_NewMail "NewMail"
#define XkbBN_Open "Open"
#define XkbBN_Proceed "Proceed"
#define XkbBN_Question "Question"
#define XkbBN_RepeatingLastBell "RepeatingLastBell"

```

```
#define XkbBN_Start      "Start"
#define XkbBN_Success    "Success"
#define XkbBN_TerminalBell "TerminalBell"
#define XkbBN_Wait       "Wait"
#define XkbBN_Warning    "Warning"
```

6.13.5 X11/extensions/XShm.h

```
#define X_ShmQueryVersion      0
#define X_ShmAttach           1
#define X_ShmDetach           2
#define X_ShmPutImage          3
#define X_ShmGetImage          4
#define X_ShmCreatePixmap      5
#define ShmCompletion          0
#define ShmNumberEvents (ShmCompletion + 1)
#define BadShmSeg              0
#define ShmNumberErrors (BadShmSeg + 1)

typedef unsigned long int ShmSeg;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    Drawable drawable;
    int major_code;
    int minor_code;
    ShmSeg shmseg;
    unsigned long int offset;
} XShmCompletionEvent;
typedef struct {
    ShmSeg shmseg;
    int shmids;
    char *shmaddr;
    int readOnly;
} XShmSegmentInfo;
extern int XShmAttach(Display *, XShmSegmentInfo *);
extern XImage *XShmCreateImage(Display *, Visual *, unsigned int,
int,
                                char *, XShmSegmentInfo *, unsigned int,
                                unsigned int);
extern Pixmap XShmCreatePixmap(Display *, Drawable, char *,
                                XShmSegmentInfo *, unsigned int,
                                unsigned int, unsigned int);
extern int XShmDetach(Display *, XShmSegmentInfo *);
extern int XShmGetEventBase(Display *);
extern int XShmGetImage(Display *, Drawable, XImage *, int, int,
                                unsigned long int);
extern int XShmPixmapFormat(Display *);
extern int XShmPutImage(Display *, Drawable, GC, XImage *, int, int,
int,
                                int, unsigned int, unsigned int, int);
extern int XShmQueryExtension(Display *);
extern int XShmQueryVersion(Display *, int *, int *, int *);
```

6.13.6 X11/extensions/Xdbe.h

```
#define XdbeBadBuffer 0

typedef Drawable XdbeBackBuffer;
typedef unsigned char XdbeSwapAction;
typedef struct {
```

```

    Window swap_window;
    XdbeSwapAction swap_action;
} XdbeSwapInfo;
typedef struct {
    Window window;
} XdbeBackBufferAttributes;
typedef struct {
    int type;
    Display *display;
    XdbeBackBuffer buffer;
    unsigned long int serial;
    unsigned char error_code;
    unsigned char request_code;
    unsigned char minor_code;
} XdbeBufferError;
extern XdbeBackBuffer XdbeAllocateBackBufferName(Display *, Window,
                                                XdbeSwapAction);

extern int XdbeBeginIdiom(Display *);
extern int XdbeDeallocateBackBufferName(Display *, XdbeBackBuffer);
extern int XdbeEndIdiom(Display *);
extern void XdbeFreeVisualInfo(XdbeScreenVisualInfo *);
extern XdbeBackBufferAttributes
*XdbeGetBackBufferAttributes(Display *,
                             XdbeBackBuffer);

extern XdbeScreenVisualInfo *XdbeGetVisualInfo(Display *, Drawable
*,
                                                int *);
extern int XdbeQueryExtension(Display *, int *, int *);
extern int XdbeSwapBuffers(Display *, XdbeSwapInfo *, int);

```

6.13.7 X11/extensions/Xdbeproto.h

```

#define DbeNumberErrors (DbeBadBuffer + 1)
#define DBE_MINOR_VERSION 0
#define DbeBadBuffer 0
#define DbeNumberEvents 0
#define X_DbeGetVersion 0
#define XdbeUndefined 0
#define DBE_MAJOR_VERSION 1
#define X_DbeAllocateBackBufferName 1
#define XdbeBackground 1
#define X_DbeDeallocateBackBufferName 2
#define XdbeUntouched 2
#define X_DbeSwapBuffers 3
#define XdbeCopied 3
#define X_DbeBeginIdiom 4
#define X_DbeEndIdiom 5
#define X_DbeGetVisualInfo 6
#define X_DbeGetBackBufferAttributes 7
#define DBE_PROTOCOL_NAME "DOUBLE-BUFFER"

typedef CARD8 xDbeSwapAction;
typedef CARD32 xDbeBackBuffer;

```

6.13.8 X11/extensions/Xext.h

```

#define X_EXTENSION_MISSING "missing"
#define X_EXTENSION_UNKNOWN "unknown"

extern int XMissingExtension(Display *, const char *);
extern

```



```

    int (*XSetExtensionErrorHandler(int (*)(Display *, char *, char
*))
    (Display *, char *, char *);

```

6.13.9 X11/extensions/dbeproto.h

```
#define DBE_PROTO_H
```

6.13.10 X11/extensions/dpms.h

```

#define DPMSModeOn      0
#define DPMSModeStandby 1
#define DPMSModeSuspend 2
#define DPMSModeOff     3

extern int DPMSCapable(Display *);
extern int DPMSDisable(Display *);
extern int DPMSEnable(Display *);
extern int DPMSForceLevel(Display *, CARD16);
extern int DPMSGetTimeouts(Display *, CARD16 *, CARD16 *, CARD16
*);
extern int DPMSGetVersion(Display *, int *, int *);
extern int DPMSInfo(Display *, CARD16 *, BOOL *);
extern int DPMSQueryExtension(Display *, int *, int *);
extern int DPMSSetTimeouts(Display *, CARD16, CARD16, CARD16);

```

6.13.11 X11/extensions/dpmsproto.h

```
#define _DPMSPROTO_H_
```

6.13.12 X11/extensions/dpmsstr.h

```

#define DPMSNumberErrors      0
#define DPMSNumberEvents      0
#define X_DPMSGetVersion      0
#define DPMSMajorVersion      1
#define DPMSMinorVersion      1
#define X_DPMSCapable         1
#define _DPMSSTR_H_           1
#define sz_xDPMSSetTimeoutsReq 12
#define X_DPMSGetTimeouts      2
#define X_DPMSSetTimeouts      3
#define sz_xDPMSCapableReply    32
#define sz_xDPMSGetTimeoutsReply 32
#define sz_xDPMSGetVersionReply 32
#define sz_xDPMSInfoReply       32
#define X_DPMSEnable           4
#define sz_xDPMSCapableReq      4
#define sz_xDPMSDisableReq      4
#define sz_xDPMSEnableReq       4
#define sz_xDPMSGetTimeoutsReq  4
#define sz_xDPMSInfoReq         4
#define X_DPMSDisable           5
#define X_DPMSForceLevel        6
#define X_DPMSInfo              7
#define sz_xDPMSForceLevelReq    8
#define sz_xDPMSGetVersionReq    8
#define DPMSExtensionName       "DPMS"

typedef struct {

```

```

        CARD8 reqType;
        CARD8 dpmsReqType;
        CARD16 length;
        CARD16 majorVersion;
        CARD16 minorVersion;
    } xDPMSGetVersionReq;
typedef struct {
    BYTE type;
    CARD8 pad0;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 majorVersion;
    CARD16 minorVersion;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xDPMSGetVersionReply;
typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
} xDPMSCapableReq;
typedef struct {
    BYTE type;
    CARD8 pad0;
    CARD16 sequenceNumber;
    CARD32 length;
    BOOL capable;
    CARD8 pad1;
    CARD16 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
    CARD32 pad7;
} xDPMSCapableReply;
typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
} xDPMSGetTimeoutsReq;
typedef struct {
    BYTE type;
    CARD8 pad0;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 standby;
    CARD16 suspend;
    CARD16 off;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
} xDPMSGetTimeoutsReply;
typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
    CARD16 standby;
    CARD16 suspend;
    CARD16 off;
    CARD16 pad0;
} xDPMSSetTimeoutsReq;

```

```

typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
} xDPMSEnableReq;
typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
} xDPMSDisableReq;
typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
    CARD16 level;
    CARD16 pad0;
} xDPMSForceLevelReq;
typedef struct {
    CARD8 reqType;
    CARD8 dpmsReqType;
    CARD16 length;
} xDPMSInfoReq;
typedef struct {
    BYTE type;
    CARD8 pad0;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 power_level;
    BOOL state;
    CARD8 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xDPMSInfoReply;

```

6.13.13 X11/extensions/extutil.h

```

#define XextHasExtension(i) ((i) && ((i)->codes))
#define XEXT_ERROR_STRING_PROTO(proc) \
    char *proc(Display *dpy, int code, XExtCodes *codes, char
*buf, int n)
#define XEXT_CLOSE_DISPLAY_PROTO(proc) \
    int proc(Display *dpy, XExtCodes *codes)
#define XEXT_FIND_DISPLAY_PROTO(proc) \
    XExtDisplayInfo *proc(Display *dpy)
#define XextCheckExtension(dpy,i,name,val) \
    if (!XextHasExtension(i)) { XMissingExtension (dpy, name); return
val; }
#define XextSimpleCheckExtension(dpy,i,name) \
    if (!XextHasExtension(i)) { XMissingExtension (dpy, name);
return; }
#define XEXT_GENERATE_ERROR_STRING(proc,extname,nerr,errl) \
    char *proc (Display *dpy, int code, XExtCodes *codes, char *buf,
int n) \
    { \
        code -= codes->first_error; \
        if (code >= 0 && code < nerr) { \
            char tmp[256]; \
            sprintf (tmp, "%s.%d", extname, code); \
            XGetErrorDatabaseText (dpy, "XProtoError", tmp, errl[code],
buf, n); \
            return buf; \
        } \
    }

```

```

        return (char *)0; \
    }
#define XEXT_GENERATE_CLOSE_DISPLAY(proc,extinfo) \
int proc (Display *dpy, XExtCodes *codes) \
{ \
    return XextRemoveDisplay (extinfo, dpy); \
}
#define
XEXT_GENERATE_FIND_DISPLAY(proc,extinfo,extname,hooks,nev,data) \
XExtDisplayInfo *proc (Display *dpy) \
{ \
    XExtDisplayInfo *dpyinfo; \
    if (!extinfo) { if (!(extinfo = XextCreateExtension())) return
NULL; } \
    if (!(dpyinfo = XextFindDisplay (extinfo, dpy))) \
        dpyinfo = XextAddDisplay (extinfo,dpy,extname,hooks,nev,data); \
    return dpyinfo; \
}

typedef struct _XExtDisplayInfo {
    struct _XExtDisplayInfo *next;
    Display *display;
    XExtCodes *codes;
    XPointer data;
} XExtDisplayInfo;
typedef struct _XExtensionInfo {
    XExtDisplayInfo *head;
    XExtDisplayInfo *cur;
    int ndisplays;
} XExtensionInfo;
typedef struct _XExtensionHooks {
    int (*create_gc) (Display *, GC, XExtCodes *);
    int (*copy_gc) (Display *, GC, XExtCodes *);
    int (*flush_gc) (Display *, GC, XExtCodes *);
    int (*free_gc) (Display *, GC, XExtCodes *);
    int (*create_font) (Display *, XFontStruct *, XExtCodes *);
    int (*free_font) (Display *, XFontStruct *, XExtCodes *);
    int (*close_display) (Display *, XExtCodes *);
    int (*wire_to_event) (Display *, XEvent *, xEvent *);
    int (*event_to_wire) (Display *, XEvent *, xEvent *);
    int (*error) (Display *, xError *, XExtCodes *, int *);
    char *(*error_string) (Display *, int, XExtCodes *, char *, int);
} XExtensionHooks;
extern XExtDisplayInfo *XextAddDisplay(XExtensionInfo *, Display *,
char *,
                                XExtensionHooks *, int, XPointer);
extern XExtensionInfo *XextCreateExtension(void);
extern void XextDestroyExtension(XExtensionInfo *);
extern XExtDisplayInfo *XextFindDisplay(XExtensionInfo *, Display
*);
extern int XextRemoveDisplay(XExtensionInfo *, Display *);

```

6.13.14 X11/extensions/security.h

```

#define XSecurityAuthorizationName        "XC-QUERY-SECURITY-1"
#define XSecurityAuthorizationRevokedMask    (1<<0)
#define XSecurityTimeout                (1<<0)
#define XSecurityTrustLevel              (1<<1)
#define XSecurityGroup                    (1<<2)
#define XSecurityEventMask                (1<<3)
#define XSecurityAllAuthorizationAttributes    (XSecurityTimeout
| XSecurityTrustLevel | XSecurityGroup | XSecurityEventMask)
#define XSecurityAuthorizationRevoked    0
#define XSecurityBadAuthorization        0

```

```

#define XSecurityClientTrusted 0
#define XSecurityBadAuthorizationProtocol 1
#define XSecurityClientUntrusted 1
#define XSecurityNumberEvents 1
#define XSecurityAuthorizationNameLen 19
#define XSecurityNumberErrors 2
#define XSecurityAllEventMasks XSecurityAuthorizationRevokedMask

typedef unsigned long int XSecurityAuthorization;
typedef struct {
    unsigned int timeout;
    unsigned int trust_level;
    XID group;
    long int event_mask;
} XSecurityAuthorizationAttributes;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    XSecurityAuthorization auth_id;
} XSecurityAuthorizationRevokedEvent;
extern Xauth *XSecurityAllocXauth(void);
extern void XSecurityFreeXauth(Xauth *);
extern Xauth *XSecurityGenerateAuthorization(Display *, Xauth *,
    unsigned long int,
    XSecurityAuthorizationAttributes *, XSecurityAuthorization *);
extern int XSecurityQueryExtension(Display *, int *, int *);
extern int XSecurityRevokeAuthorization(Display *,
    XSecurityAuthorization);

```

6.13.15 X11/extensions/securproto.h

```
#define _SECURPROTO_H
```

6.13.16 X11/extensions/shape.h

```

#define X_ShapeQueryVersion 0
#define X_ShapeRectangles 1
#define X_ShapeMask 2
#define X_ShapeCombine 3
#define X_ShapeOffset 4
#define X_ShapeQueryExtents 5
#define X_ShapeSelectInput 6
#define X_ShapeInputSelected 7
#define X_ShapeGetRectangles 8
#define ShapeSet 0
#define ShapeUnion 1
#define ShapeIntersect 2
#define ShapeSubtract 3
#define ShapeInvert 4
#define ShapeBounding 0
#define ShapeClip 1
#define ShapeInput 2
#define ShapeNotifyMask (1L << 0)
#define ShapeNotify 0
#define ShapeNumberEvents (ShapeNotify + 1)

typedef struct {
    int type;
    unsigned long int serial;

```

```

    int send_event;
    Display *display;
    Window window;
    int kind;
    int x;
    int y;
    unsigned int width;
    unsigned int height;
    Time time;
    int shaped;
} XShapeEvent;
extern void XShapeCombineMask(Display *, Window, int, int, int,
Pixmap,
                                int);
extern void XShapeCombineRectangles(Display *, Window, int, int,
int,
                                XRectangle *, int, int, int);
extern void XShapeCombineRegion(Display *, Window, int, int, int,
Region,
                                int);
extern void XShapeCombineShape(Display *, Window, int, int, int,
Window,
                                int, int);
extern XRectangle *XShapeGetRectangles(Display *, Window, int, int
*,
                                int *);
extern unsigned long int XShapeInputSelected(Display *, Window);
extern void XShapeOffsetShape(Display *, Window, int, int, int);
extern int XShapeQueryExtension(Display *, int *, int *);
extern int XShapeQueryExtents(Display *, Window, int *, int *, int
*,
                                unsigned int *, unsigned int *, int *, int
*,
                                int *, unsigned int *, unsigned int *);
extern int XShapeQueryVersion(Display *, int *, int *);
extern void XShapeSelectInput(Display *, Window, unsigned long int);

```

6.13.17 X11/extensions/shapeproto.h

```
#define _SHAPEPROTO_H_
```

6.13.18 X11/extensions/shmproto.h

```
#define _SHMPROTO_H_
```

6.13.19 X11/extensions/sync.h

```

typedef XID XSyncCounter;
typedef XID XSyncAlarm;
typedef struct _XSyncValue {
    int hi;
    unsigned int lo;
} XSyncValue;
typedef enum {
    XSyncAbsolute,
    XSyncRelative
} XSyncValueType;
typedef enum {
    XSyncPositiveTransition,
    XSyncNegativeTransition,
    XSyncPositiveComparison,
    XSyncNegativeComparison
}

```

```

} XSyncTestType;
typedef enum {
    XSyncAlarmActive,
    XSyncAlarmInactive,
    XSyncAlarmDestroyed
} XSyncAlarmState;

#define SYNC_NAME "SYNC"
#define _XSyncValueIsPositive(v) ((v).hi&0x80000000)?0:1)
#define _XSyncValueIsNegative(v) ((v).hi&0x80000000)?1:0)
#define _XSyncValueLessThan(a,b) ((a).hi<(b).hi ||
((a).hi==(b).hi && (a).lo<(b).lo))
#define _XSyncValueLessOrEqual(a,b) ((a).hi<(b).hi ||
((a).hi==(b).hi && (a).lo<=(b).lo))
#define _XSyncValueGreaterThan(a,b) ((a).hi>(b).hi ||
((a).hi==(b).hi && (a).lo>(b).lo))
#define _XSyncValueGreaterOrEqual(a,b) ((a).hi>(b).hi ||
((a).hi==(b).hi && (a).lo>=(b).lo))
#define _XSyncValueEqual(a,b) ((a).lo==(b).lo && (a).hi==(b).hi)
#define _XSyncValueIsZero(a) ((a).lo==0&&(a).hi==0)
#define _XSyncIntToValue(pv,i) ((pv)->hi=((i<0)?~0:0), (pv)-
>lo=(i))
#define _XSyncMaxValue(pv) ((pv)->hi=0x7fffffff, (pv)-
>lo=0xffffffff)
#define _XSyncMinValue(pv) ((pv)->hi=0x80000000, (pv)->lo=0)
#define _XSyncIntsToValue(pv,l,h) ((pv)->lo = (l), (pv)->hi =
(h))
#define _XSyncValueHigh32(v) ((v).hi)
#define _XSyncValueLow32(v) ((v).lo)
#define XSyncCACounter (1L<<0)
#define XSyncCAValueType (1L<<1)
#define XSyncCAValue (1L<<2)
#define XSyncCATestType (1L<<3)
#define XSyncCADelta (1L<<4)
#define XSyncCAEvents (1L<<5)
#define XSyncAlarmNotifyMask (1L<<XSyncAlarmNotify)
#define XSyncNumberErrors (XSyncBadAlarm+1)
#define SYNC_MINOR_VERSION 0
#define XSyncCounterNotify 0
#define X_SyncInitialize 0
#define XSyncBadCounter 0L
#define XSyncAlarmNotify 1
#define X_SyncListSystemCounters 1
#define X_SyncQueryAlarm 10
#define X_SyncDestroyAlarm 11
#define X_SyncSetPriority 12
#define X_SyncGetPriority 13
#define XSyncBadAlarm 1L
#define X_SyncCreateCounter 2
#define XSyncNumberEvents 2L
#define SYNC_MAJOR_VERSION 3
#define X_SyncSetCounter 3
#define X_SyncChangeCounter 4
#define X_SyncQueryCounter 5
#define X_SyncDestroyCounter 6
#define X_SyncAwait 7
#define X_SyncCreateAlarm 8
#define X_SyncChangeAlarm 9
#define _XSyncValueAdd(presult,a,b,poverflow) {\
int t = (a).lo;\
Bool signa = XSyncValueIsNegative(a);\
Bool signb = XSyncValueIsNegative(b);\
((presult)->lo = (a).lo + (b).lo);\
((presult)->hi = (a).hi + (b).hi);\
if (t>(presult)->lo) (presult)->hi++;

```

```

        *poverflow = ((signa == signb) && !(signa ==
XSyncValueIsNegative(*presult))); \
    }
#define _XSyncValueSubtract(presult,a,b,poverflow) {\
int t = (a).lo; \
    Bool signa = XSyncValueIsNegative(a); \
    Bool signb = XSyncValueIsNegative(b); \
    ((presult)->lo = (a).lo - (b).lo); \
    ((presult)->hi = (a).hi - (b).hi); \
    if (t > (presult)->lo) (presult)->hi--; \
    *poverflow = ((signa == signb) && !(signa ==
XSyncValueIsNegative(*presult))); \
    }

typedef struct _XSyncSystemCounter {
    char *name;
    XSyncCounter counter;
    XSyncValue resolution;
} XSyncSystemCounter;
typedef struct {
    XSyncCounter counter;
    XSyncValueType value_type;
    XSyncValue wait_value;
    XSyncTestType test_type;
} XSyncTrigger;
typedef struct {
    XSyncTrigger trigger;
    XSyncValue event_threshold;
} XSyncWaitCondition;
typedef struct {
    XSyncTrigger trigger;
    XSyncValue delta;
    int events;
    XSyncAlarmState state;
} XSyncAlarmAttributes;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    XSyncCounter counter;
    XSyncValue wait_value;
    XSyncValue counter_value;
    Time time;
    int count;
    int destroyed;
} XSyncCounterNotifyEvent;
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    XSyncAlarm alarm;
    XSyncValue counter_value;
    XSyncValue alarm_value;
    Time time;
    XSyncAlarmState state;
} XSyncAlarmNotifyEvent;
typedef struct {
    int type;
    Display *display;
    XSyncAlarm alarm;
    unsigned long int serial;
    unsigned char error_code;
    unsigned char request_code;
    unsigned char minor_code;

```



```

} XSyncAlarmError;
typedef struct {
    int type;
    Display *display;
    XSyncCounter counter;
    unsigned long int serial;
    unsigned char error_code;
    unsigned char request_code;
    unsigned char minor_code;
} XSyncCounterError;
extern int XSyncAwait(Display *, XSyncWaitCondition *, int);
extern int XSyncChangeAlarm(Display *, XSyncAlarm, unsigned long
int,
                        XSyncAlarmAttributes *);
extern int XSyncChangeCounter(Display *, XSyncCounter, XSyncValue);
extern XSyncAlarm XSyncCreateAlarm(Display *, unsigned long int,
                        XSyncAlarmAttributes *);
extern XSyncCounter XSyncCreateCounter(Display *, XSyncValue);
extern int XSyncDestroyAlarm(Display *, XSyncAlarm);
extern int XSyncDestroyCounter(Display *, XSyncCounter);
extern void XSyncFreeSystemCounterList(XSyncSystemCounter *);
extern int XSyncGetPriority(Display *, XID, int *);
extern int XSyncInitialize(Display *, int *, int *);
extern void XSyncIntToValue(XSyncValue *, int);
extern void XSyncIntsToValue(XSyncValue *, unsigned int, int);
extern XSyncSystemCounter *XSyncListSystemCounters(Display *, int
*);
extern void XSyncMaxValue(XSyncValue *);
extern void XSyncMinValue(XSyncValue *);
extern int XSyncQueryAlarm(Display *, XSyncAlarm,
XSyncAlarmAttributes *);
extern int XSyncQueryCounter(Display *, XSyncCounter, XSyncValue
*);
extern int XSyncQueryExtension(Display *, int *, int *);
extern int XSyncSetCounter(Display *, XSyncCounter, XSyncValue);
extern int XSyncSetPriority(Display *, XID, int);
extern void XSyncValueAdd(XSyncValue *, XSyncValue, XSyncValue, int
*);
extern int XSyncValueEqual(XSyncValue, XSyncValue);
extern int XSyncValueGreaterOrEqual(XSyncValue, XSyncValue);
extern int XSyncValueGreaterThan(XSyncValue, XSyncValue);
extern int XSyncValueHigh32(XSyncValue);
extern int XSyncValueIsNegative(XSyncValue);
extern int XSyncValueIsPositive(XSyncValue);
extern int XSyncValueIsZero(XSyncValue);
extern int XSyncValueLessOrEqual(XSyncValue, XSyncValue);
extern int XSyncValueLessThan(XSyncValue, XSyncValue);
extern unsigned int XSyncValueLow32(XSyncValue);
extern void XSyncValueSubtract(XSyncValue *, XSyncValue, XSyncValue,
int *);

```

6.13.20 X11/extensions/syncproto.h

```
#define _SYNCPROTO_H_
```

6.13.21 X11/extensions/syncstr.h

```

#define sz_xSyncChangeAlarmReq 12
#define sz_xSyncCreateAlarmReq 12
#define sz_xSyncSetPriorityReq 12
#define sz_xSyncSystemCounter 14
#define sz_xSyncChangeCounterReq 16
#define sz_xSyncCreateCounterReq 16

```

```

#define sz_xSyncSetCounterReq 16
#define sz_xSyncWaitCondition 28
#define sz_xSyncGetPriorityReply 32
#define sz_xSyncInitializeReply 32
#define sz_xSyncListSystemCountersReply 32
#define sz_xSyncQueryCounterReply 32
#define sz_xSyncAwaitReq 4
#define sz_xSyncListSystemCountersReq 4
#define sz_xSyncQueryAlarmReply 40
#define sz_xSyncDestroyAlarmReq 8
#define sz_xSyncDestroyCounterReq 8
#define sz_xSyncGetPriorityReq 8
#define sz_xSyncInitializeReq 8
#define sz_xSyncQueryAlarmReq 8
#define sz_xSyncQueryCounterReq 8

typedef struct _xSyncInitialize {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD8 majorVersion;
    CARD8 minorVersion;
    CARD16 pad;
} xSyncInitializeReq;
typedef struct {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 majorVersion;
    CARD8 minorVersion;
    CARD16 pad;
    CARD32 pad0;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xSyncInitializeReply;
typedef struct _xSyncListSystemCounters {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
} xSyncListSystemCountersReq;
typedef struct {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    INT32 nCounters;
    CARD32 pad0;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xSyncListSystemCountersReply;
typedef struct {
    CARD32 counter;
    INT32 resolution_hi;
    CARD32 resolution_lo;
    CARD16 name_length;
} xSyncSystemCounter;
typedef struct _xSyncCreateCounterReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 cid;

```

```

        INT32 initial_value_hi;
        CARD32 initial_value_lo;
    } xSyncCreateCounterReq;
typedef struct _xSyncChangeCounterReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 cid;
    INT32 value_hi;
    CARD32 value_lo;
} xSyncChangeCounterReq;
typedef struct _xSyncSetCounterReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 cid;
    INT32 value_hi;
    CARD32 value_lo;
} xSyncSetCounterReq;
typedef struct _xSyncDestroyCounterReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 counter;
} xSyncDestroyCounterReq;
typedef struct _xSyncQueryCounterReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 counter;
} xSyncQueryCounterReq;
typedef struct {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    INT32 value_hi;
    CARD32 value_lo;
    CARD32 pad0;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
} xSyncQueryCounterReply;
typedef struct _xSyncAwaitReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
} xSyncAwaitReq;
typedef struct _xSyncWaitCondition {
    CARD32 counter;
    CARD32 value_type;
    INT32 wait_value_hi;
    CARD32 wait_value_lo;
    CARD32 test_type;
    INT32 event_threshold_hi;
    CARD32 event_threshold_lo;
} xSyncWaitCondition;
typedef struct _xSyncCreateAlarmReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 id;
    CARD32 valueMask;
} xSyncCreateAlarmReq;
typedef struct _xSyncDestroyAlarmReq {
    CARD8 reqType;

```

```

        CARD8 syncReqType;
        CARD16 length;
        CARD32 alarm;
    } xSyncDestroyAlarmReq;
typedef struct _xSyncQueryAlarmReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 alarm;
} xSyncQueryAlarmReq;
typedef struct {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 counter;
    CARD32 value_type;
    INT32 wait_value_hi;
    CARD32 wait_value_lo;
    CARD32 test_type;
    INT32 delta_hi;
    CARD32 delta_lo;
    BOOL events;
    BYTE state;
    BYTE pad0;
    BYTE pad1;
} xSyncQueryAlarmReply;
typedef struct _xSyncChangeAlarmReq {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 alarm;
    CARD32 valueMask;
} xSyncChangeAlarmReq;
typedef struct _xSyncSetPriority {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 id;
    INT32 priority;
} xSyncSetPriorityReq;
typedef struct _xSyncGetPriority {
    CARD8 reqType;
    CARD8 syncReqType;
    CARD16 length;
    CARD32 id;
} xSyncGetPriorityReq;
typedef struct {
    BYTE type;
    CARD8 unused;
    CARD16 sequenceNumber;
    CARD32 length;
    INT32 priority;
    CARD32 pad0;
    CARD32 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
} xSyncGetPriorityReply;
typedef struct _xSyncCounterNotifyEvent {
    BYTE type;
    BYTE kind;
    CARD16 sequenceNumber;
    CARD32 counter;
    INT32 wait_value_hi;
    CARD32 wait_value_lo;

```

```

INT32 counter_value_hi;
CARD32 counter_value_lo;
CARD32 time;
CARD16 count;
BOOL destroyed;
BYTE pad0;
} xSyncCounterNotifyEvent;
typedef struct _xSyncAlarmNotifyEvent {
    BYTE type;
    BYTE kind;
    CARD16 sequenceNumber;
    CARD32 alarm;
    INT32 counter_value_hi;
    CARD32 counter_value_lo;
    INT32 alarm_value_hi;
    CARD32 alarm_value_lo;
    CARD32 time;
    CARD8 state;
    BYTE pad0;
    BYTE pad1;
    BYTE pad2;
} xSyncAlarmNotifyEvent;

```

6.13.22 X11/extensions/xtestproto.h

```
#define _XTESTPROTO_H_
```

6.14 Interface Definitions for libXext

The interfaces defined on the following pages are included in libXext and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 6.12 shall behave as described in the referenced base document.

6.15 Interfaces for libXi

Table 6-20 defines the library name and shared object name for the libXi library

Table 6-20 libXi Definition

Library:	libXi
SONAME:	libXi.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[XINPUT] X11 Input Library

6.15.1 XInput

6.15.1.1 Interfaces for XInput

An LSB conforming implementation shall provide the generic functions for XInput specified in Table 6-21, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-21 libXi - XInput Function Interfaces

XAllowDeviceEvents [XINPUT]	XChangeDeviceControl [XINPUT]	XChangeDeviceDontPropagateList [XINPUT]	XChangeDeviceKeyMapping [XINPUT]
XChangeFeedbackControl [XINPUT]	XChangeKeyboardDevice [XINPUT]	XChangePointerDevice [XINPUT]	XCloseDevice [XINPUT]
XDeviceBell [XINPUT]	XFreeDeviceControl [XINPUT]	XFreeDeviceList [XINPUT]	XFreeDeviceMotionEvents [XINPUT]
XFreeDeviceState [XINPUT]	XFreeFeedbackList [XINPUT]	XGetDeviceButtonMapping [XINPUT]	XGetDeviceControl [XINPUT]
XGetDeviceDontPropagateList [XINPUT]	XGetDeviceFocus [XINPUT]	XGetDeviceKeyMapping [XINPUT]	XGetDeviceModifierMapping [XINPUT]
XGetDeviceMotionEvents [XINPUT]	XGetExtensionVersion [XINPUT]	XGetFeedbackControl [XINPUT]	XGetSelectedExtensionEvents [XINPUT]
XGrabDevice [XINPUT]	XGrabDeviceButton [XINPUT]	XGrabDeviceKey [XINPUT]	XListInputDevices [XINPUT]
XOpenDevice [XINPUT]	XQueryDeviceState [XINPUT]	XSelectExtensionEvent [XINPUT]	XSendExtensionEvent [XINPUT]
XSetDeviceButtonMapping [XINPUT]	XSetDeviceFocus [XINPUT]	XSetDeviceMode [XINPUT]	XSetDeviceModifierMapping [XINPUT]
XSetDeviceValuators [XINPUT]	XUngrabDevice [XINPUT]	XUngrabDeviceButton [XINPUT]	XUngrabDeviceKey [XINPUT]

6.16 Data Definitions for libXi

This section defines global identifiers and their values that are associated with interfaces contained in libXi. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.16.1 X11/extensions/XI.h

```

#define InProximity      (0L << 1)
#define DeviceMode      (1L << 0)
#define DvAccelNum      (1L << 0)
#define DvInteger       (1L << 0)
#define DvString        (1L << 0)
#define DvAccelDenom    (1L << 1)
#define OutOfProximity  (1L << 1)
#define ProximityState  (1L << 1)
#define DvThreshold     (1L << 2)
#define DvKeyClickPercent (1L<<0)
#define DvPercent       (1L<<1)
#define DvPitch (1L<<2)
#define DvDuration      (1L<<3)
#define DvLed (1L<<4)
#define DvLedMode       (1L<<5)
#define DvKey (1L<<6)
#define DvAutoRepeatMode (1L<<7)
#define AddToList        0
#define AsyncThisDevice 0
#define COUNT            0
#define DeviceAdded      0
#define Dont_Check       0
#define IsXPointer       0
#define KbdFeedbackClass 0
#define KeyClass         0
#define NewPointer       0
#define Relative         0
#define XI_Absent        0
#define XI_BadDevice     0
#define XI_Initial_Release_Minor 0
#define XPOINTER         0
#define _devicePointerMotionHint 0
#define _devicePresence 0
#define UseXKeyboard     0xFF
#define Absolute         1
#define ButtonClass      1
#define CREATE           1
#define DEVICE_RESOLUTION 1
#define DeleteFromList 1
#define DeviceRemoved    1
#define IsXKeyboard      1
#define NewKeyboard      1
#define NoSuchExtension 1
#define PtrFeedbackClass 1
#define SyncThisDevice   1
#define XI_Add_DevicePresenceNotify_Major 1
#define XI_Add_XChangeDeviceControl_Major 1
#define XI_Add_XDeviceBell_Major 1
#define XI_Add_XDeviceBell_Minor 1
#define XI_Add_XSetDeviceValuators_Major 1
#define XI_BadEvent      1
#define XI_Initial_Release_Major 1
#define XI_Present       1
#define XInput_Initial_Release 1
#define XKEYBOARD        1
#define _deviceButton1Motion 1
#define sz_xAllowDeviceEventsReq 12
#define sz_xChangeDeviceDontPropagateListReq 12
#define sz_xChangeFeedbackControlReq 12
#define sz_xSelectExtensionEventReq 12
#define sz_xUngrabDeviceReq 12
#define sz_xGetDeviceMotionEventsReq 16
#define sz_xSendExtensionEventReq 16

```

```

#define sz_xSetDeviceFocusReq 16
#define sz_xUngrabDeviceButtonReq 16
#define sz_xUngrabDeviceKeyReq 16
#define DEVICE_ABS_CALIB 2
#define DeviceEnabled 2
#define IsXExtensionDevice 2
#define ReplayThisDevice 2
#define StringFeedbackClass 2
#define ValuatorClass 2
#define XI_Add_XSetDeviceValuators_Minor 2
#define XI_BadMode 2
#define XInput_Add_XDeviceBell 2
#define _deviceButton2Motion 2
#define sz_xGrabDeviceButtonReq 20
#define sz_xGrabDeviceKeyReq 20
#define sz_xGrabDeviceReq 20
#define AsyncOtherDevices 3
#define DEVICE_CORE 3
#define DeviceDisabled 3
#define FeedbackClass 3
#define FollowKeyboard 3
#define IntegerFeedbackClass 3
#define IsXExtensionKeyboard 3
#define RevertToFollowKeyboard 3
#define XI_Add_XChangeDeviceControl_Minor 3
#define XI_DeviceBusy 3
#define XInput_Add_XSetDeviceValuators 3
#define _deviceButton3Motion 3
#define sz_xChangeDeviceControlReply 32
#define sz_xChangeKeyboardDeviceReply 32
#define sz_xChangePointerDeviceReply 32
#define sz_xGetDeviceButtonMappingReply 32
#define sz_xGetDeviceControlReply 32
#define sz_xGetDeviceDontPropagateListReply 32
#define sz_xGetDeviceFocusReply 32
#define sz_xGetDeviceKeyMappingReply 32
#define sz_xGetDeviceMotionEventsReply 32
#define sz_xGetExtensionVersionReply 32
#define sz_xGetFeedbackControlReply 32
#define sz_xGetSelectedExtensionEventsReply 32
#define sz_xGrabDeviceButtonReply 32
#define sz_xGrabDeviceKeyReply 32
#define sz_xGrabDeviceReply 32
#define sz_xListInputDevicesReply 32
#define sz_xOpenDeviceReply 32
#define sz_xQueryDeviceStateReply 32
#define sz_xSetDeviceButtonMappingReply 32
#define sz_xSetDeviceModeReply 32
#define sz_xSetDeviceModifierMappingReply 32
#define sz_xSetDeviceValuatorsReply 32
#define AsyncAll 4
#define DEVICE_ENABLE 4
#define DeviceUnrecoverable 4
#define IsXExtensionPointer 4
#define LedFeedbackClass 4
#define ProximityClass 4
#define XI_Add_DevicePresenceNotify_Minor 4
#define XI_BadClass 4
#define XInput_Add_XChangeDeviceControl 4
#define _deviceButton4Motion 4
#define sz_xListInputDevicesReq 4
#define BellFeedbackClass 5
#define DEVICE_ABS_AREA 5
#define FocusClass 5
#define SyncAll 5
#define XInput_Add_DevicePresenceNotify 5

```



```

#define _deviceButton5Motion    5
#define OtherClass              6
#define _deviceButtonMotion    6
#define _deviceButtonGrab      7
#define _deviceOwnerGrabButton 8
#define sz_xChangeDeviceControlReq 8
#define sz_xChangeDeviceKeyMappingReq 8
#define sz_xChangeKeyboardDeviceReq 8
#define sz_xChangePointerDeviceReq 8
#define sz_xCloseDeviceReq      8
#define sz_xDeviceBellReq       8
#define sz_xGetDeviceButtonMappingReq 8
#define sz_xGetDeviceControlReq 8
#define sz_xGetDeviceDontPropagateListReq 8
#define sz_xGetDeviceFocusReq   8
#define sz_xGetDeviceKeyMappingReq 8
#define sz_xGetDeviceModifierMappingReq 8
#define sz_xGetExtensionVersionReq 8
#define sz_xGetFeedbackControlReq 8
#define sz_xGetSelectedExtensionEventsReq 8
#define sz_xOpenDeviceReq       8
#define sz_xQueryDeviceStateReq 8
#define sz_xSetDeviceButtonMappingReq 8
#define sz_xSetDeviceModeReq    8
#define sz_xSetDeviceModifierMappingReq 8
#define sz_xSetDeviceValuatorsReq 8
#define _noExtensionEvent       9
#define XI_BARCODE              "BARCODE"
#define XI_BUTTONBOX            "BUTTONBOX"
#define XI_CURSORKEYS           "CURSORKEYS"
#define XI_DATAGLOVE            "DATAGLOVE"
#define XI_EYETRACKER           "EYETRACKER"
#define XI_FOOTMOUSE            "FOOTMOUSE"
#define XI_ID_MODULE            "ID_MODULE"
#define XI_KEYBOARD             "KEYBOARD"
#define XI_KNOB_BOX             "KNOB_BOX"
#define XI_MOUSE                "MOUSE"
#define XI_NINE_KNOB            "NINE_KNOB"
#define XI_ONE_KNOB             "ONE_KNOB"
#define XI_QUADRATURE           "QUADRATURE"
#define XI_SPACEBALL            "SPACEBALL"
#define XI_TABLET               "TABLET"
#define XI_TOUCHPAD             "TOUCHPAD"
#define XI_TOUCHSCREEN          "TOUCHSCREEN"
#define XI_TRACKBALL            "TRACKBALL"
#define XI_NAME                  "XInputExtension"

```

```

typedef long unsigned int XEventClass;
typedef struct {
    int present;
    short int major_version;
    short int minor_version;
} XExtensionVersion;

```

6.16.2 X11/extensions/XInput.h

```

#define _XINPUT_H_
#define DeviceButtonPress(d,type,_class) \
    FindTypeAndClass(d, type, _class, ButtonClass, \
    _deviceButtonPress)
#define DeviceButtonRelease(d,type,_class) \
    FindTypeAndClass(d, type, _class, ButtonClass, \
    _deviceButtonRelease)
#define DeviceFocusIn(d,type,_class) \
    FindTypeAndClass(d, type, _class, FocusClass, _deviceFocusIn)

```

```

#define DeviceFocusOut(d,type,_class) \
    FindTypeAndClass(d, type, _class, FocusClass,
_deviceFocusOut)
#define DeviceKeyPress(d,type,_class) \
    FindTypeAndClass(d, type, _class, KeyClass, _deviceKeyPress)
#define DeviceKeyRelease(d,type,_class) \
    FindTypeAndClass(d, type, _class, KeyClass,
_deviceKeyRelease)
#define ChangeDeviceNotify(d,type,_class) \
    FindTypeAndClass(d, type, _class, OtherClass,
_changeDeviceNotify)
#define DeviceMappingNotify(d,type,_class) \
    FindTypeAndClass(d, type, _class, OtherClass,
_deviceMappingNotify)
#define DeviceStateNotify(d,type,_class) \
    FindTypeAndClass(d, type, _class, OtherClass,
_deviceStateNotify)
#define ProximityIn(d,type,_class) \
    FindTypeAndClass(d, type, _class, ProximityClass,
_proximityIn)
#define ProximityOut(d,type,_class) \
    FindTypeAndClass(d, type, _class, ProximityClass,
_proximityOut)
#define DeviceMotionNotify(d,type,_class) \
    FindTypeAndClass(d, type, _class, ValuatorClass,
_deviceMotionNotify)
#define FindTypeAndClass(d,type,_class,classid,offset) \
    { int _i; XInputClassInfo *_ip; type = 0; _class = 0; for
(_i=0, _ip= \
    ((XDevice *) d)->classes; _i< ((XDevice *) d)->num_classes;
_i++, \
    _ip++) if (_ip->input_class == classid) {type = _ip-
>event_type_base + \
    offset; _class = ((XDevice *) d)->device_id << 8 | type;}}
#define DevicePointerMotionHint(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 | \
    _devicePointerMotionHint;}
#define DeviceButton1Motion(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_deviceButton1Motion;}
#define DeviceButton2Motion(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_deviceButton2Motion;}
#define DeviceButton3Motion(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_deviceButton3Motion;}
#define DeviceButton5Motion(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_deviceButton5Motion;}
#define DeviceButtonPressGrab(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_deviceButtonGrab;}
#define DeviceOwnerGrabButton(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_deviceOwnerGrabButton;}
#define NoExtensionEvent(d,type,_class) \
    { _class = ((XDevice *) d)->device_id << 8 |
_noExtensionEvent;}
#define _deviceButtonPress 0
#define _deviceFocusIn 0
#define _deviceKeyPress 0
#define _deviceMotionNotify 0
#define _deviceStateNotify 0
#define _proximityIn 0
#define _deviceButtonRelease 1
#define _deviceFocusOut 1

```

```

#define _deviceKeyRelease      1
#define _deviceMappingNotify  1
#define _proximityOut         1
#define _changeDeviceNotify    2
#define DeviceButton4Motion(d,type, _class)      { _class =
((XDevice *) d)->device_id << 8 | _deviceButton4Motion;}
#define DeviceButtonMotion(d,type, _class)      { _class =
((XDevice *) d)->device_id << 8 | _deviceButtonMotion;}
#define DevicePresence(dpy, type, _class)      {\
extern int _XiGetDevicePresenceNotifyEvent(Display *); \
type = _XiGetDevicePresenceNotifyEvent(dpy); \
_class = (0x10000 | _devicePresence); \
}

typedef struct {
    unsigned char input_class;
    unsigned char event_type_base;
} XInputClassInfo;
typedef struct {
    XID device_id;
    int num_classes;
    XInputClassInfo *classes;
} XDevice;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    unsigned int keycode;
    int same_screen;
    unsigned int device_state;
    unsigned char axes_count;
    unsigned char first_axis;
    int axis_data[6];
} XDeviceKeyEvent;
typedef XDeviceKeyEvent XDeviceKeyPressedEvent;
typedef XDeviceKeyEvent XDeviceKeyReleasedEvent;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    unsigned int button;
    int same_screen;
    unsigned int device_state;
    unsigned char axes_count;

```

```

        unsigned char first_axis;
        int axis_data[6];
    } XDeviceButtonEvent;
typedef XDeviceButtonEvent XDeviceButtonPressedEvent;
typedef XDeviceButtonEvent XDeviceButtonReleasedEvent;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    char is_hint;
    int same_screen;
    unsigned int device_state;
    unsigned char axes_count;
    unsigned char first_axis;
    int axis_data[6];
} XDeviceMotionEvent;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    int mode;
    int detail;
    Time time;
} XDeviceFocusChangeEvent;
typedef XDeviceFocusChangeEvent XDeviceFocusInEvent;
typedef XDeviceFocusChangeEvent XDeviceFocusOutEvent;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    Window root;
    Window subwindow;
    Time time;
    int x;
    int y;
    int x_root;
    int y_root;
    unsigned int state;
    int same_screen;
    unsigned int device_state;
    unsigned char axes_count;
    unsigned char first_axis;
    int axis_data[6];
} XProximityNotifyEvent;
typedef XProximityNotifyEvent XProximityInEvent;
typedef XProximityNotifyEvent XProximityOutEvent;
typedef struct {
#ifdef __cplusplus || defined(cplusplus)
    unsigned c_class;

```

```

#else
    unsigned class;
#endif
    unsigned char length;
} XInputClass;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    Time time;
    int num_classes;
    char data[64];
} XDeviceStateNotifyEvent;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    unsigned c_class;
#else
    unsigned class;
#endif
    unsigned char length;
    unsigned char num_valuators;
    unsigned char mode;
    int valuators[6];
} XValuatorStatus;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    unsigned c_class;
#else
    unsigned class;
#endif
    unsigned char length;
    short int num_keys;
    char keys[32];
} XKeyStatus;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    unsigned c_class;
#else
    unsigned class;
#endif
    unsigned char length;
    short int num_buttons;
    char buttons[32];
} XButtonStatus;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;
    Time time;
    int request;
    int first_keycode;
    int count;
} XDeviceMappingEvent;
typedef struct {
    int type;
    long unsigned int serial;
    int send_event;
    Display *display;
    Window window;
    XID deviceid;

```

```

        Time time;
        int request;
    } XChangeDeviceNotifyEvent;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
} XFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int click;
    int percent;
    int pitch;
    int duration;
    int led_mask;
    int global_auto_repeat;
    char auto_repeats[32];
} XKbdFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int accelNum;
    int accelDenom;
    int threshold;
} XPtrFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int resolution;
    int minVal;
    int maxVal;
} XIntegerFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int max_symbols;
    int num_syms_supported;
    KeySym *syms_supported;
} XStringFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)

```

```

        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int percent;
    int pitch;
    int duration;
} XBellFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int led_values;
    int led_mask;
} XLedFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
} XFeedbackControl;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int accelNum;
    int accelDenom;
    int threshold;
} XPtrFeedbackControl;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;
    XID id;
    int click;
    int percent;
    int pitch;
    int duration;
    int led_mask;
    int led_value;
    int key;
    int auto_repeat_mode;
} XKbdFeedbackControl;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        XID c_class;
    #else
        XID class;
    #endif
    int length;

```

```

        XID id;
        int num_keysyms;
        KeySym *syms_to_display;
    } XStringFeedbackControl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
    XID id;
    int int_to_display;
} XIntegerFeedbackControl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
    XID id;
    int percent;
    int pitch;
    int duration;
} XBellFeedbackControl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
    XID id;
    int led_mask;
    int led_values;
} XLedFeedbackControl;
typedef struct {
    XID control;
    int length;
} XDeviceControl;
typedef struct {
    XID control;
    int length;
    int first_valuator;
    int num_valuators;
    int *resolutions;
} XDeviceResolutionControl;
typedef struct {
    XID control;
    int length;
    int num_valuators;
    int *resolutions;
    int *min_resolutions;
    int *max_resolutions;
} XDeviceResolutionState;
typedef struct _XAnyClassinfo *XAnyClassPtr;
typedef struct _XAnyClassinfo {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
} XAnyClassInfo;
typedef struct _XDeviceInfo *XDeviceInfoPtr;

```



```

typedef struct _XDeviceInfo {
    XID id;
    Atom type;
    char *name;
    int num_classes;
    int use;
    XAnyClassPtr inputclassinfo;
} XDeviceInfo;
typedef struct _XKeyInfo *XKeyInfoPtr;
typedef struct _XKeyInfo {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
    short unsigned int min_keycode;
    short unsigned int max_keycode;
    short unsigned int num_keys;
} XKeyInfo;
typedef struct _XButtonInfo *XButtonInfoPtr;
typedef struct _XButtonInfo {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
    short int num_buttons;
} XButtonInfo;
typedef struct _XAxisInfo *XAxisInfoPtr;
typedef struct _XAxisInfo {
    int resolution;
    int min_value;
    int max_value;
} XAxisInfo;
typedef struct _XValuatorInfo *XValuatorInfoPtr;
typedef struct _XValuatorInfo {
#if defined(__cplusplus) || defined(c_plusplus)
    XID c_class;
#else
    XID class;
#endif
    int length;
    unsigned char num_axes;
    unsigned char mode;
    long unsigned int motion_buffer;
    XAxisInfoPtr axes;
} XValuatorInfo;
typedef struct {
    XEventClass event_type;
    XID device;
} XEventList;
typedef struct {
    Time time;
    int *data;
} XDeviceTimeCoord;
typedef struct {
    XID device_id;
    int num_classes;
    XInputClass *data;
} XDeviceState;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    unsigned c_class;
#else

```

```

        unsigned class;
    #endif
        unsigned char length;
        unsigned char num_valuators;
        unsigned char mode;
        int *valuators;
    } XValuatorState;
    typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        unsigned c_class;
    #else
        unsigned class;
    #endif
        unsigned char length;
        short int num_keys;
        char keys[32];
    } XKeyState;
    typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        unsigned c_class;
    #else
        unsigned class;
    #endif
        unsigned char length;
        short int num_buttons;
        char buttons[32];
    } XButtonState;
    extern int XAllowDeviceEvents(Display *, XDevice *, int, Time);
    extern int XChangeDeviceControl(Display *, XDevice *, int,
                                   XDeviceControl *);
    extern int XChangeDeviceDontPropagateList(Display *, Window, int,
                                               XEventClass *, int);
    extern int XChangeDeviceKeyMapping(Display *, XDevice *, int, int,
                                       KeySym *, int);
    extern int XChangeFeedbackControl(Display *, XDevice *, unsigned
    long int,
                                   XFeedbackControl *);
    extern int XChangeKeyboardDevice(Display *, XDevice *);
    extern int XChangePointerDevice(Display *, XDevice *, int, int);
    extern int XCloseDevice(Display *, XDevice *);
    extern int XDeviceBell(Display *, XDevice *, XID, XID, int);
    extern void XFreeDeviceControl(XDeviceControl *);
    extern void XFreeDeviceList(XDeviceInfo *);
    extern void XFreeDeviceMotionEvents(XDeviceTimeCoord *);
    extern void XFreeDeviceState(XDeviceState *);
    extern void XFreeFeedbackList(XFeedbackState *);
    extern int XGetDeviceButtonMapping(Display *, XDevice *, unsigned
    char *,
                                   unsigned int);
    extern XDeviceControl *XGetDeviceControl(Display *, XDevice *, int);
    extern XEventClass *XGetDeviceDontPropagateList(Display *, Window,
    int *);
    extern int XGetDeviceFocus(Display *, XDevice *, Window *, int *,
    Time *);
    extern KeySym *XGetDeviceKeyMapping(Display *, XDevice *, unsigned
    int,
                                   int, int *);
    extern XModifierKeymap *XGetDeviceModifierMapping(Display *,
    XDevice *);
    extern XDeviceTimeCoord *XGetDeviceMotionEvents(Display *, XDevice
    *, Time,
                                   Time, int *, int *, int *);
    extern XExtensionVersion *XGetExtensionVersion(Display *, const
    char *);
    extern XFeedbackState *XGetFeedbackControl(Display *, XDevice *,
    int *);

```

```

extern int XGetSelectedExtensionEvents(Display *, Window, int *,
                                     XEventClass **, int *,
                                     XEventClass * *);
extern int XGrabDevice(Display *, XDevice *, Window, int, int,
                      XEventClass *, int, int, Time);
extern int XGrabDeviceButton(Display *, XDevice *, unsigned int,
                             unsigned int, XDevice *, Window, int,
                             unsigned int, XEventClass *, int, int);
extern int XGrabDeviceKey(Display *, XDevice *, unsigned int,
                          unsigned int,
                          XDevice *, Window, int, unsigned int,
                          XEventClass *, int, int);
extern XDeviceInfo *XListInputDevices(Display *, int *);
extern XDevice *XOpenDevice(Display *, XID);
extern XDeviceState *XQueryDeviceState(Display *, XDevice *);
extern int XSelectExtensionEvent(Display *, Window, XEventClass *,
                                int);
extern int XSendExtensionEvent(Display *, XDevice *, Window, int,
                               int,
                               XEventClass *, XEvent *);
extern int XSetDeviceButtonMapping(Display *, XDevice *, unsigned
char *,
                                int);
extern int XSetDeviceFocus(Display *, XDevice *, Window, int, Time);
extern int XSetDeviceMode(Display *, XDevice *, int);
extern int XSetDeviceModifierMapping(Display *, XDevice *,
                                     XModifierKeymap *);
extern int XSetDeviceValuators(Display *, XDevice *, int *, int,
                               int);
extern int XUngrabDevice(Display *, XDevice *, Time);
extern int XUngrabDeviceButton(Display *, XDevice *, unsigned int,
                               unsigned int, XDevice *, Window);
extern int XUngrabDeviceKey(Display *, XDevice *, unsigned int,
                            unsigned int, XDevice *, Window);

```

6.16.3 X11/extensions/XIproto.h

```

#define XI_DeviceValuator      0
#define InputClassBits 0x3F
#define DEVICE_BITS 0x7F
#define MORE_EVENTS 0x80
#define CLIENT_REQ 1
#define XI_DeviceKeyPress 1
#define X_GetExtensionVersion 1
#define XI_DeviceStateNotify 10
#define X_GetDeviceMotionEvents 10
#define XI_DeviceMappingNotify 11
#define X_ChangeKeyboardDevice 11
#define XI_ChangeDeviceNotify 12
#define X_ChangePointerDevice 12
#define XI_DeviceKeystateNotify 13
#define X_GrabDevice 13
#define XI_DeviceButtonstateNotify 14
#define X_UngrabDevice 14
#define XI_DevicePresenceNotify 15
#define X_GrabDeviceKey 15
#define IEVENTS 16
#define X_UngrabDeviceKey 16
#define X_GrabDeviceButton 17
#define X_UngrabDeviceButton 18
#define X_AllowDeviceEvents 19
#define XI_DeviceKeyRelease 2
#define X_ListInputDevices 2
#define X_GetDeviceFocus 20
#define X_SetDeviceFocus 21

```

```

#define X_GetFeedbackControl 22
#define X_ChangeFeedbackControl 23
#define X_GetDeviceKeyMapping 24
#define X_ChangeDeviceKeyMapping 25
#define X_GetDeviceModifierMapping 26
#define X_SetDeviceModifierMapping 27
#define X_GetDeviceButtonMapping 28
#define X_SetDeviceButtonMapping 29
#define XI_DeviceButtonPress 3
#define X_OpenDevice 3
#define X_QueryDeviceState 30
#define X_SendExtensionEvent 31
#define X_DeviceBell 32
#define X_SetDeviceValuators 33
#define X_GetDeviceControl 34
#define X_ChangeDeviceControl 35
#define XI_DeviceButtonRelease 4
#define X_CloseDevice 4
#define IERRORS 5
#define XI_DeviceMotionNotify 5
#define X_SetDeviceMode 5
#define ModeBitsShift 6
#define XI_DeviceFocusIn 6
#define X_SelectExtensionEvent 6
#define XI_DeviceFocusOut 7
#define X_GetSelectedExtensionEvents 7
#define numInputClasses 7
#define XI_ProximityIn 8
#define X_ChangeDeviceDontPropagateList 8
#define XI_ProximityOut 9
#define X_GetDeviceDontPropagateList 9

typedef struct _XExtEventInfo {
    Mask mask;
    BYTE type;
    BYTE word;
} XExtEventInfo;
typedef unsigned char *Pointer;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD16 nbytes;
    CARD8 pad1;
    CARD8 pad2;
} xGetExtensionVersionReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 major_version;
    CARD16 minor_version;
    BOOL present;
    CARD8 pad1;
    CARD8 pad2;
    CARD8 pad3;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
} xGetExtensionVersionReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;

```

```

} xListInputDevicesReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 ndevices;
    CARD8 pad1;
    CARD8 pad2;
    CARD8 pad3;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xListInputDevicesReply;
typedef struct _xDeviceInfo *xDeviceInfoPtr;
typedef struct _xAnyClassinfo *xAnyClassPtr;
typedef struct _xAnyClassinfo {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
    CARD8 length;
} xAnyClassInfo;
typedef struct _xDeviceInfo {
    CARD32 type;
    CARD8 id;
    CARD8 num_classes;
    CARD8 use;
    CARD8 attached;
} xDeviceInfo;
typedef struct _xKeyInfo *xKeyInfoPtr;
typedef struct _xKeyInfo {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
    CARD8 length;
    CARD8 min_keycode;
    CARD8 max_keycode;
    CARD16 num_keys;
    CARD8 pad1;
    CARD8 pad2;
} xKeyInfo;
typedef struct _xButtonInfo *xButtonInfoPtr;
typedef struct _xButtonInfo {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
    CARD8 length;
    CARD16 num_buttons;
} xButtonInfo;
typedef struct _xValuatorInfo *xValuatorInfoPtr;
typedef struct _xValuatorInfo {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
    CARD8 length;
    CARD8 num_axes;
}

```

```

        CARD8 mode;
        CARD32 motion_buffer_size;
    } xValuatorInfo;
typedef struct _xAxisInfo *xAxisInfoPtr;
typedef struct _xAxisInfo {
    CARD32 resolution;
    CARD32 min_value;
    CARD32 max_value;
} xAxisInfo;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xOpenDeviceReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 num_classes;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad00;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
} xOpenDeviceReply;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 event_type_base;
} xInputClassInfo;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xCloseDeviceReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    CARD8 mode;
    BYTE pad1;
    BYTE pad2;
} xSetDeviceModeReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 status;
    BYTE pad1;

```

```

        BYTE pad2;
        BYTE pad3;
        CARD32 pad01;
        CARD32 pad02;
        CARD32 pad03;
        CARD32 pad04;
        CARD32 pad05;
    } xSetDeviceModeReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 window;
    CARD16 count;
    CARD16 pad00;
} xSelectExtensionEventReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 window;
} xGetSelectedExtensionEventsReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 this_client_count;
    CARD16 all_clients_count;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xGetSelectedExtensionEventsReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 window;
    CARD16 count;
    CARD8 mode;
    BYTE pad;
} xChangeDeviceDontPropagateListReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 window;
} xGetDeviceDontPropagateListReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 count;
    CARD16 pad00;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xGetDeviceDontPropagateListReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;

```

```

        CARD16 length;
        CARD32 start;
        CARD32 stop;
        CARD8 deviceid;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } xGetDeviceMotionEventsReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD32 nEvents;
    CARD8 axes;
    CARD8 mode;
    BYTE pad1;
    BYTE pad2;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
} xGetDeviceMotionEventsReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xChangeKeyboardDeviceReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 status;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xChangeKeyboardDeviceReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 xaxis;
    CARD8 yaxis;
    CARD8 deviceid;
    BYTE pad1;
} xChangePointerDeviceReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 status;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad01;

```



```

        CARD32 pad02;
        CARD32 pad03;
        CARD32 pad04;
        CARD32 pad05;
    } xChangePointerDeviceReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 grabWindow;
    CARD32 time;
    CARD16 event_count;
    CARD8 this_device_mode;
    CARD8 other_devices_mode;
    BOOL ownerEvents;
    CARD8 deviceid;
    CARD16 pad01;
} xGrabDeviceReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 status;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xGrabDeviceReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 time;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xUngrabDeviceReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 grabWindow;
    CARD16 event_count;
    CARD16 modifiers;
    CARD8 modifier_device;
    CARD8 grabbed_device;
    CARD8 key;
    BYTE this_device_mode;
    BYTE other_devices_mode;
    BOOL ownerEvents;
    BYTE pad1;
    BYTE pad2;
} xGrabDeviceKeyReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 grabWindow;
    CARD16 modifiers;
    CARD8 modifier_device;

```

```

        CARD8 key;
        CARD8 grabbed_device;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } xUngrabDeviceKeyReq;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD32 grabWindow;
        CARD8 grabbed_device;
        CARD8 modifier_device;
        CARD16 event_count;
        CARD16 modifiers;
        BYTE this_device_mode;
        BYTE other_devices_mode;
        CARD8 button;
        BOOL ownerEvents;
        BYTE pad1;
        BYTE pad2;
    } xGrabDeviceButtonReq;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD32 grabWindow;
        CARD16 modifiers;
        CARD8 modifier_device;
        CARD8 button;
        CARD8 grabbed_device;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } xUngrabDeviceButtonReq;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD32 time;
        CARD8 mode;
        CARD8 deviceid;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } xAllowDeviceEventsReq;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD8 deviceid;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } xGetDeviceFocusReq;
    typedef struct {
        CARD8 repType;
        CARD8 RepType;
        CARD16 sequenceNumber;
        CARD32 length;
        CARD32 focus;
        CARD32 time;
        CARD8 revertTo;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
        CARD32 pad01;
    }

```

```

        CARD32 pad02;
        CARD32 pad03;
    } xGetDeviceFocusReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD32 focus;
    CARD32 time;
    CARD8 revertTo;
    CARD8 device;
    CARD16 pad01;
} xSetDeviceFocusReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xGetFeedbackControlReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD16 num_feedbacks;
    CARD16 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
    CARD32 pad06;
} xGetFeedbackControlReply;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
} xFeedbackState;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    CARD16 pitch;
    CARD16 duration;
    CARD32 led_mask;
    CARD32 led_values;
    BOOL global_auto_repeat;
    CARD8 click;
    CARD8 percent;
    BYTE pad;
    BYTE auto_repeats[32];
} xKbdFeedbackState;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else

```

```

        CARD8 class;
    #endif
        CARD8 id;
        CARD16 length;
        CARD8 pad1;
        CARD8 pad2;
        CARD16 accelNum;
        CARD16 accelDenom;
        CARD16 threshold;
    } xPtrFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
        CARD8 id;
        CARD16 length;
        CARD32 resolution;
        INT32 min_value;
        INT32 max_value;
    } xIntegerFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
        CARD8 id;
        CARD16 length;
        CARD16 max_symbols;
        CARD16 num_syms_supported;
    } xStringFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
        CARD8 id;
        CARD16 length;
        CARD8 percent;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
        CARD16 pitch;
        CARD16 duration;
    } xBellFeedbackState;
typedef struct {
    #if defined(__cplusplus) || defined(c_plusplus)
        CARD8 c_class;
    #else
        CARD8 class;
    #endif
        CARD8 id;
        CARD16 length;
        CARD32 led_mask;
        CARD32 led_values;
    } xLedFeedbackState;
typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD32 mask;
        CARD8 deviceid;
        CARD8 feedbackid;

```

```

        BYTE pad1;
        BYTE pad2;
    } xChangeFeedbackControlReq;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
} xFeedbackCtl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    CARD8 key;
    CARD8 auto_repeat_mode;
    INT8 click;
    INT8 percent;
    INT16 pitch;
    INT16 duration;
    CARD32 led_mask;
    CARD32 led_values;
} xKbdFeedbackCtl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    CARD8 pad1;
    CARD8 pad2;
    INT16 num;
    INT16 denom;
    INT16 thresh;
} xPtrFeedbackCtl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    INT32 int_to_display;
} xIntegerFeedbackCtl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    CARD8 pad1;
    CARD8 pad2;
    CARD16 num_keysyms;
} xStringFeedbackCtl;
typedef struct {

```

```

#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    INT8 percent;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    INT16 pitch;
    INT16 duration;
} xBellFeedbackCtl;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 id;
    CARD16 length;
    CARD32 led_mask;
    CARD32 led_values;
} xLedFeedbackCtl;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    CARD8 firstKeyCode;
    CARD8 count;
    BYTE pad1;
} xGetDeviceKeyMappingReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 keySymsPerKeyCode;
    CARD8 pad0;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetDeviceKeyMappingReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    CARD8 firstKeyCode;
    CARD8 keySymsPerKeyCode;
    CARD8 keyCodes;
} xChangeDeviceKeyMappingReq;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xGetDeviceModifierMappingReq;

```

```

typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 numKeyPerModifier;
    CARD8 pad0;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xGetDeviceModifierMappingReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    CARD8 numKeyPerModifier;
    CARD16 pad1;
} xSetDeviceModifierMappingReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 success;
    CARD8 pad0;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xSetDeviceModifierMappingReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xGetDeviceButtonMappingReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 nElts;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xGetDeviceButtonMappingReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    CARD8 map_length;

```

```

        BYTE pad1;
        BYTE pad2;
    } xSetDeviceButtonMappingReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 status;
    BYTE pad0;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xSetDeviceButtonMappingReply;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
} xQueryDeviceStateReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 num_classes;
    BYTE pad0;
    CARD16 pad1;
    CARD32 pad2;
    CARD32 pad3;
    CARD32 pad4;
    CARD32 pad5;
    CARD32 pad6;
} xQueryDeviceStateReply;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 length;
    CARD8 num_keys;
    BYTE pad1;
    CARD8 keys[32];
} xKeyState;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else
    CARD8 class;
#endif
    CARD8 length;
    CARD8 num_buttons;
    BYTE pad1;
    CARD8 buttons[32];
} xButtonState;
typedef struct {
#if defined(__cplusplus) || defined(c_plusplus)
    CARD8 c_class;
#else

```



```

        CARD8 class;
    #endif
        CARD8 length;
        CARD8 num_valuators;
        CARD8 mode;
    } xValuatorState;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD32 destination;
        CARD8 deviceid;
        BOOL propagate;
        CARD16 count;
        CARD8 num_events;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
    } xSendExtensionEventReq;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD8 deviceid;
        CARD8 feedbackid;
        CARD8 feedbackclass;
        INT8 percent;
    } xDeviceBellReq;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD8 deviceid;
        CARD8 first_valuator;
        CARD8 num_valuators;
        BYTE pad1;
    } xSetDeviceValuatorsReq;
    typedef struct {
        CARD8 repType;
        CARD8 RepType;
        CARD16 sequenceNumber;
        CARD32 length;
        CARD8 status;
        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
        CARD32 pad01;
        CARD32 pad02;
        CARD32 pad03;
        CARD32 pad04;
        CARD32 pad05;
    } xSetDeviceValuatorsReply;
    typedef struct {
        CARD8 reqType;
        CARD8 ReqType;
        CARD16 length;
        CARD16 control;
        CARD8 deviceid;
        BYTE pad2;
    } xGetDeviceControlReq;
    typedef struct {
        CARD8 repType;
        CARD8 RepType;
        CARD16 sequenceNumber;
        CARD32 length;
        CARD8 status;
    }

```

```

        BYTE pad1;
        BYTE pad2;
        BYTE pad3;
        CARD32 pad01;
        CARD32 pad02;
        CARD32 pad03;
        CARD32 pad04;
        CARD32 pad05;
    } xGetDeviceControlReply;
typedef struct {
    CARD16 control;
    CARD16 length;
} xDeviceState;
typedef struct {
    CARD16 control;
    CARD16 length;
    CARD32 num_valuators;
} xDeviceResolutionState;
typedef struct {
    CARD8 reqType;
    CARD8 ReqType;
    CARD16 length;
    CARD16 control;
    CARD8 deviceid;
    BYTE pad0;
} xChangeDeviceControlReq;
typedef struct {
    CARD8 repType;
    CARD8 RepType;
    CARD16 sequenceNumber;
    CARD32 length;
    CARD8 status;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
    CARD32 pad05;
} xChangeDeviceControlReply;
typedef struct {
    CARD16 control;
    CARD16 length;
} xDeviceCtl;
typedef struct {
    CARD16 control;
    CARD16 length;
    CARD8 first_valuator;
    CARD8 num_valuators;
    CARD8 pad1;
    CARD8 pad2;
} xDeviceResolutionCtl;
typedef struct {
    BYTE type;
    CARD8 deviceid;
    CARD16 sequenceNumber;
    KeyButMask device_state;
    CARD8 num_valuators;
    CARD8 first_valuator;
    INT32 valuator0;
    INT32 valuator1;
    INT32 valuator2;
    INT32 valuator3;
    INT32 valuator4;
    INT32 valuator5;

```

```

} deviceValuator;
typedef struct {
    BYTE type;
    BYTE detail;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD32 root;
    CARD32 event;
    CARD32 child;
    INT16 root_x;
    INT16 root_y;
    INT16 event_x;
    INT16 event_y;
    KeyButMask state;
    BOOL same_screen;
    CARD8 deviceid;
} deviceKeyButtonPointer;
typedef struct {
    BYTE type;
    BYTE detail;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD32 window;
    BYTE mode;
    CARD8 deviceid;
    BYTE pad1;
    BYTE pad2;
    CARD32 pad00;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
} deviceFocus;
typedef struct {
    BYTE type;
    BYTE deviceid;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 num_keys;
    CARD8 num_buttons;
    CARD8 num_valuators;
    CARD8 classes_reported;
    CARD8 buttons[4];
    CARD8 keys[4];
    INT32 valuator0;
    INT32 valuator1;
    INT32 valuator2;
} deviceStateNotify;
typedef struct {
    BYTE type;
    BYTE deviceid;
    CARD16 sequenceNumber;
    CARD8 keys[28];
} deviceKeyStateNotify;
typedef struct {
    BYTE type;
    BYTE deviceid;
    CARD16 sequenceNumber;
    CARD8 buttons[28];
} deviceButtonStateNotify;
typedef struct {
    BYTE type;
    BYTE deviceid;
    CARD16 sequenceNumber;
    CARD8 request;
    CARD8 firstKeyCode;
    CARD8 count;

```

```

    BYTE pad1;
    CARD32 time;
    CARD32 pad00;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
} deviceMappingNotify;
typedef struct {
    BYTE type;
    BYTE deviceid;
    CARD16 sequenceNumber;
    CARD32 time;
    CARD8 request;
    BYTE pad1;
    BYTE pad2;
    BYTE pad3;
    CARD32 pad00;
    CARD32 pad01;
    CARD32 pad02;
    CARD32 pad03;
    CARD32 pad04;
} changeDeviceNotify;

```

6.17 Interfaces for libXtst

Table 6-22 defines the library name and shared object name for the libXtst library

Table 6-22 libXtst Definition

Library:	libXtst
SONAME:	libXtst.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[X-test] XTEST Extension Library

6.17.1 libXtst Interfaces

6.17.1.1 Interfaces for libXtst Interfaces

An LSB conforming implementation shall provide the generic functions for libXtst Interfaces specified in Table 6-23, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-23 libXtst - libXtst Interfaces Function Interfaces

XTestCompareCurrentCursorWithWindow [X-test]	XTestCompareCursorWithWindow [X-test]	XTestDiscard [X-test]	XTestFakeButtonEvent [X-test]
XTestFakeDeviceButtonEvent [X-test]	XTestFakeDeviceKeyEvent [X-test]	XTestFakeDeviceMotionEvent [X-test]	XTestFakeKeyEvent [X-test]
XTestFakeMotionEvent [X-test]	XTestFakeProximityEvent [X-test]	XTestFakeRelativeMotionEvent [X-test]	XTestGrabControl [X-test]

XTestQueryExtension [X-test]	XTestSetGCContextOfGC [X-test]	XTestSetVisualIDOfVisual [X-test]	

6.18 Data Definitions for libXtst

This section defines global identifiers and their values that are associated with interfaces contained in libXtst. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.18.1 X11/extensions/XTest.h

```
#define XTestNumberErrors      0
#define XTestNumberEvents      0
#define X_XTestGetVersion      0
#define X_XTestCompareCursor   1
#define XTestMajorVersion      2
#define XTestMinorVersion      2
#define X_XTestFakeInput       2
#define X_XTestGrabControl     3
#define XTestExtensionName     "XTEST"

extern int XTestCompareCurrentCursorWithWindow(Display *, Window);
extern int XTestCompareCursorWithWindow(Display *, Window, Cursor);
extern int XTestDiscard(Display *);
extern int XTestFakeButtonEvent(Display *, unsigned int, int,
                                unsigned long int);
extern int XTestFakeDeviceButtonEvent(Display *, XDevice *,
                                       unsigned int,
                                       int, int *, int, unsigned long int);
extern int XTestFakeDeviceKeyEvent(Display *, XDevice *, unsigned
int, int,
                                int *, int, unsigned long int);
extern int XTestFakeDeviceMotionEvent(Display *, XDevice *, int,
int,
                                int *, int, unsigned long int);
extern int XTestFakeKeyEvent(Display *, unsigned int, int,
                             unsigned long int);
extern int XTestFakeMotionEvent(Display *, int, int, int,
                                unsigned long int);
extern int XTestFakeProximityEvent(Display *, XDevice *, int, int
*, int,
                                unsigned long int);
extern int XTestFakeRelativeMotionEvent(Display *, int, int,
                                       unsigned long int);
extern int XTestGrabControl(Display *, int);
extern int XTestQueryExtension(Display *, int *, int *, int *, int
*);
```

```
extern void XTestSetGContextOfGC(GC, GContext);
extern void XTestSetVisualIDOfVisual(Visual *, VisualID);
```

6.19 Interfaces for libxcb

Table 6-24 defines the library name and shared object name for the libxcb library

Table 6-24 libxcb Definition

Library:	libxcb
SONAME:	libxcb.so.1

The behavior of the interfaces in this library is specified by the following specifications:

[Libxcb 1.7] Libxcb API
 [LSB] This Specification

6.19.1 libxcb interfaces

6.19.1.1 Interfaces for libxcb interfaces

An LSB conforming implementation shall provide the generic functions for libxcb interfaces specified in Table 6-25, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-25 libxcb - libxcb interfaces Function Interfaces

xcb_alloc_color [LSB]	xcb_alloc_color_cells [LSB]	xcb_alloc_color_cells_masks [LSB]	xcb_alloc_color_cells_masks_end [LSB]
xcb_alloc_color_cells_masks_length [LSB]	xcb_alloc_color_cells_pixels [LSB]	xcb_alloc_color_cells_pixels_end [LSB]	xcb_alloc_color_cells_pixels_length [LSB]
xcb_alloc_color_cells_reply [LSB]	xcb_alloc_color_cells_unchecked [LSB]	xcb_alloc_color_planes [LSB]	xcb_alloc_color_planes_pixels [LSB]
xcb_alloc_color_planes_pixels_end [LSB]	xcb_alloc_color_planes_pixels_length [LSB]	xcb_alloc_color_planes_reply [LSB]	xcb_alloc_color_planes_unchecked [LSB]
xcb_alloc_color_reply [LSB]	xcb_alloc_color_unchecked [LSB]	xcb_alloc_named_color [LSB]	xcb_alloc_named_color_reply [LSB]
xcb_alloc_named_color_unchecked [LSB]	xcb_allow_events [LSB]	xcb_allow_events_checked [LSB]	xcb_arc_end [LSB]
xcb_arc_next [LSB]	xcb_atom_end [LSB]	xcb_atom_next [LSB]	xcb_bell [LSB]
xcb_bell_checked [LSB]	xcb_big_requests_enable [LSB]	xcb_big_requests_enable_reply [LSB]	xcb_big_requests_enable_unchecked [LSB]

xcb_button_end [LSB]	xcb_button_next [LSB]	xcb_change_active_pointer_grab [LSB]	xcb_change_active_pointer_grab_checked [LSB]
xcb_change_gc [LSB]	xcb_change_gc_checked [LSB]	xcb_change_hosts [LSB]	xcb_change_hosts_checked [LSB]
xcb_change_keyboard_control [LSB]	xcb_change_keyboard_control_checked [LSB]	xcb_change_keyboard_mapping [LSB]	xcb_change_keyboard_mapping_checked [LSB]
xcb_change_pointer_control [LSB]	xcb_change_pointer_control_checked [LSB]	xcb_change_property [LSB]	xcb_change_property_checked [LSB]
xcb_change_save_set [LSB]	xcb_change_save_set_checked [LSB]	xcb_change_window_attributes [LSB]	xcb_change_window_attributes_checked [LSB]
xcb_char2b_end [LSB]	xcb_char2b_next [LSB]	xcb_charinfo_end [LSB]	xcb_charinfo_next [LSB]
xcb_circulate_window [LSB]	xcb_circulate_window_checked [LSB]	xcb_clear_area [LSB]	xcb_clear_area_checked [LSB]
xcb_client_message_data_end [LSB]	xcb_client_message_data_next [LSB]	xcb_close_font [LSB]	xcb_close_font_checked [LSB]
xcb_coloritem_end [LSB]	xcb_coloritem_next [LSB]	xcb_colormap_end [LSB]	xcb_colormap_next [LSB]
xcb_configure_window [LSB]	xcb_configure_window_checked [LSB]	xcb_connect [Libxcb 1.7]	xcb_connect_to_display_with_auth_info [Libxcb 1.7]
xcb_connect_to_fd [Libxcb 1.7]	xcb_connection_has_error [Libxcb 1.7]	xcb_convert_selection [LSB]	xcb_convert_selection_checked [LSB]
xcb_copy_area [LSB]	xcb_copy_area_checked [LSB]	xcb_copy_colormap_and_free [LSB]	xcb_copy_colormap_and_free_checked [LSB]
xcb_copy_gc [LSB]	xcb_copy_gc_checked [LSB]	xcb_copy_plane [LSB]	xcb_copy_plane_checked [LSB]
xcb_create_colormap [LSB]	xcb_create_colormap_checked [LSB]	xcb_create_cursor [LSB]	xcb_create_cursor_checked [LSB]
xcb_create_gc [LSB]	xcb_create_gc_checked [LSB]	xcb_create_glyph_cursor [LSB]	xcb_create_glyph_cursor_checked [LSB]
xcb_create_pixmap [LSB]	xcb_create_pixmap_checked [LSB]	xcb_create_window [LSB]	xcb_create_window_checked [LSB]

xcb_cursor_end [LSB]	xcb_cursor_next [LSB]	xcb_delete_prop erty [LSB]	xcb_delete_prop erty_checked [LSB]
xcb_depth_end [LSB]	xcb_depth_next [LSB]	xcb_depth_visua ls [LSB]	xcb_depth_visua ls_iterator [LSB]
xcb_depth_visua ls_length [LSB]	xcb_destroy_sub windows [LSB]	xcb_destroy_sub windows_checke d [LSB]	xcb_destroy_win dow [LSB]
xcb_destroy_win dow_checked [LSB]	xcb_discard_repl y [LSB]	xcb_disconnect [Libxcb 1.7]	xcb_drawable_en d [LSB]
xcb_drawable_ne xt [LSB]	xcb_fill_poly [LSB]	xcb_fill_poly_che cked [LSB]	xcb_flush [Libxcb 1.7]
xcb_font_end [LSB]	xcb_font_next [LSB]	xcb_fontable_en d [LSB]	xcb_fontable_nex t [LSB]
xcb_fontprop_en d [LSB]	xcb_fontprop_ne xt [LSB]	xcb_force_screen _saver [LSB]	xcb_force_screen _saver_checked [LSB]
xcb_format_end [LSB]	xcb_format_next [LSB]	xcb_free_colorm ap [LSB]	xcb_free_colorm ap_checked [LSB]
xcb_free_colors [LSB]	xcb_free_colors _checked [LSB]	xcb_free_cursor [LSB]	xcb_free_cursor _checked [LSB]
xcb_free_gc [LSB]	xcb_free_gc_chec ked [LSB]	xcb_free_pixmap [LSB]	xcb_free_pixmap _checked [LSB]
xcb_gcontext_en d [LSB]	xcb_gcontext_ne xt [LSB]	xcb_generate_id [Libxcb 1.7]	xcb_get_atom_na me [LSB]
xcb_get_atom_na me_name [LSB]	xcb_get_atom_na me_name_end [LSB]	xcb_get_atom_na me_name_length [LSB]	xcb_get_atom_na me_reply [LSB]
xcb_get_atom_na me_unchecked [LSB]	xcb_get_extensio n_data [Libxcb 1.7]	xcb_get_file_desc riptor [Libxcb 1.7]	xcb_get_font_pat h [LSB]
xcb_get_font_pat h_path_iterator [LSB]	xcb_get_font_pat h_path_length [LSB]	xcb_get_font_pat h_reply [LSB]	xcb_get_font_pat h_unchecked [LSB]
xcb_get_geometr y [LSB]	xcb_get_geometr y_reply [LSB]	xcb_get_geometr y_unchecked [LSB]	xcb_get_image [LSB]
xcb_get_image_d ata [LSB]	xcb_get_image_d ata_end [LSB]	xcb_get_image_d ata_length [LSB]	xcb_get_image_r eply [LSB]
xcb_get_image_u nchecked [LSB]	xcb_get_input_fo cus [LSB]	xcb_get_input_fo cus_reply [LSB]	xcb_get_input_fo cus_unchecked [LSB]

xcb_get_keyboard_control [LSB]	xcb_get_keyboard_control_reply [LSB]	xcb_get_keyboard_control_unchecked [LSB]	xcb_get_keyboard_mapping [LSB]
xcb_get_keyboard_mapping_keysyms [LSB]	xcb_get_keyboard_mapping_keysyms_end [LSB]	xcb_get_keyboard_mapping_keysyms_length [LSB]	xcb_get_keyboard_mapping_reply [LSB]
xcb_get_keyboard_mapping_unchecked [LSB]	xcb_get_maximum_request_length [Libxcb 1.7]	xcb_get_modifier_mapping [LSB]	xcb_get_modifier_mapping_keycodes [LSB]
xcb_get_modifier_mapping_keycodes_end [LSB]	xcb_get_modifier_mapping_keycodes_length [LSB]	xcb_get_modifier_mapping_reply [LSB]	xcb_get_modifier_mapping_unchecked [LSB]
xcb_get_motion_events [LSB]	xcb_get_motion_events_events [LSB]	xcb_get_motion_events_events_iterator [LSB]	xcb_get_motion_events_events_length [LSB]
xcb_get_motion_events_reply [LSB]	xcb_get_motion_events_unchecked [LSB]	xcb_get_pointer_control [LSB]	xcb_get_pointer_control_reply [LSB]
xcb_get_pointer_control_unchecked [LSB]	xcb_get_pointer_mapping [LSB]	xcb_get_pointer_mapping_map [LSB]	xcb_get_pointer_mapping_map_end [LSB]
xcb_get_pointer_mapping_map_length [LSB]	xcb_get_pointer_mapping_reply [LSB]	xcb_get_pointer_mapping_unchecked [LSB]	xcb_get_property [LSB]
xcb_get_property_reply [LSB]	xcb_get_property_unchecked [LSB]	xcb_get_property_value [LSB]	xcb_get_property_value_end [LSB]
xcb_get_property_value_length [LSB]	xcb_get_screen_saver [LSB]	xcb_get_screen_saver_reply [LSB]	xcb_get_screen_saver_unchecked [LSB]
xcb_get_selection_owner [LSB]	xcb_get_selection_owner_reply [LSB]	xcb_get_selection_owner_unchecked [LSB]	xcb_get_setup [Libxcb 1.7]
xcb_get_window_attributes [LSB]	xcb_get_window_attributes_reply [LSB]	xcb_get_window_attributes_unchecked [LSB]	xcb_grab_button [LSB]
xcb_grab_button_checked [LSB]	xcb_grab_key [LSB]	xcb_grab_key_checked [LSB]	xcb_grab_keyboard [LSB]
xcb_grab_keyboard_reply [LSB]	xcb_grab_keyboard_unchecked [LSB]	xcb_grab_pointer [LSB]	xcb_grab_pointer_reply [LSB]
xcb_grab_pointer_unchecked [LSB]	xcb_grab_server [LSB]	xcb_grab_server_checked [LSB]	xcb_host_addresses [LSB]

xcb_host_addresses_end [LSB]	xcb_host_addresses_length [LSB]	xcb_host_end [LSB]	xcb_host_next [LSB]
xcb_image_text_16 [LSB]	xcb_image_text_16_checked [LSB]	xcb_image_text_8 [LSB]	xcb_image_text_8_checked [LSB]
xcb_install_color_map [LSB]	xcb_install_color_map_checked [LSB]	xcb_intern_atom [LSB]	xcb_intern_atom_reply [LSB]
xcb_intern_atom_unchecked [LSB]	xcb_keycode_end [LSB]	xcb_keycode_next [LSB]	xcb_keysym_end [LSB]
xcb_keysym_next [LSB]	xcb_kill_client [LSB]	xcb_kill_client_checked [LSB]	xcb_list_extensions [LSB]
xcb_list_extensions_names_iterator [LSB]	xcb_list_extensions_names_length [LSB]	xcb_list_extensions_reply [LSB]	xcb_list_extensions_unchecked [LSB]
xcb_list_fonts [LSB]	xcb_list_fonts_names_iterator [LSB]	xcb_list_fonts_names_length [LSB]	xcb_list_fonts_reply [LSB]
xcb_list_fonts_unchecked [LSB]	xcb_list_fonts_with_info [LSB]	xcb_list_fonts_with_info_name [LSB]	xcb_list_fonts_with_info_name_end [LSB]
xcb_list_fonts_with_info_name_length [LSB]	xcb_list_fonts_with_info_properties [LSB]	xcb_list_fonts_with_info_properties_iterator [LSB]	xcb_list_fonts_with_info_properties_length [LSB]
xcb_list_fonts_with_info_reply [LSB]	xcb_list_fonts_with_info_unchecked [LSB]	xcb_list_hosts [LSB]	xcb_list_hosts_hosts_iterator [LSB]
xcb_list_hosts_hosts_length [LSB]	xcb_list_hosts_reply [LSB]	xcb_list_hosts_unchecked [LSB]	xcb_list_installed_colormaps [LSB]
xcb_list_installed_colormaps_cmaps [LSB]	xcb_list_installed_colormaps_cmaps_end [LSB]	xcb_list_installed_colormaps_cmaps_length [LSB]	xcb_list_installed_colormaps_reply [LSB]
xcb_list_installed_colormaps_unchecked [LSB]	xcb_list_properties [LSB]	xcb_list_properties_atoms [LSB]	xcb_list_properties_atoms_end [LSB]
xcb_list_properties_atoms_length [LSB]	xcb_list_properties_reply [LSB]	xcb_list_properties_unchecked [LSB]	xcb_lookup_color [LSB]
xcb_lookup_color_reply [LSB]	xcb_lookup_color_unchecked [LSB]	xcb_map_subwindows [LSB]	xcb_map_subwindows_checked [LSB]
xcb_map_window [LSB]	xcb_map_window_checked [LSB]	xcb_no_operation [LSB]	xcb_no_operation_checked [LSB]

xcb_open_font [LSB]	xcb_open_font_checked [LSB]	xcb_parse_display [Libxcb 1.7]	xcb_pixmap_end [LSB]
xcb_pixmap_next [LSB]	xcb_point_end [LSB]	xcb_point_next [LSB]	xcb_poll_for_event [Libxcb 1.7]
xcb_poll_for_reply [Libxcb 1.7]	xcb_poly_arc [LSB]	xcb_poly_arc_checked [LSB]	xcb_poly_fill_arc [LSB]
xcb_poly_fill_arc_checked [LSB]	xcb_poly_fill_rectangle [LSB]	xcb_poly_fill_rectangle_checked [LSB]	xcb_poly_line [LSB]
xcb_poly_line_checked [LSB]	xcb_poly_point [LSB]	xcb_poly_point_checked [LSB]	xcb_poly_rectangle [LSB]
xcb_poly_rectangle_checked [LSB]	xcb_poly_segment [LSB]	xcb_poly_segment_checked [LSB]	xcb_poly_text_16 [LSB]
xcb_poly_text_16_checked [LSB]	xcb_poly_text_8 [LSB]	xcb_poly_text_8_checked [LSB]	xcb_popcount [Libxcb 1.7]
xcb_prefetch_extension_data [Libxcb 1.7]	xcb_prefetch_maximum_request_length [LSB]	xcb_put_image [LSB]	xcb_put_image_checked [LSB]
xcb_query_best_size [LSB]	xcb_query_best_size_reply [LSB]	xcb_query_best_size_unchecked [LSB]	xcb_query_colors [LSB]
xcb_query_colors_colors [LSB]	xcb_query_colors_colors_iterator [LSB]	xcb_query_colors_colors_length [LSB]	xcb_query_colors_reply [LSB]
xcb_query_colors_unchecked [LSB]	xcb_query_extension [LSB]	xcb_query_extension_reply [LSB]	xcb_query_extension_unchecked [LSB]
xcb_query_font [LSB]	xcb_query_font_char_infos [LSB]	xcb_query_font_char_infos_iterator [LSB]	xcb_query_font_char_infos_length [LSB]
xcb_query_font_properties [LSB]	xcb_query_font_properties_iterator [LSB]	xcb_query_font_properties_length [LSB]	xcb_query_font_reply [LSB]
xcb_query_font_unchecked [LSB]	xcb_query_keymap [LSB]	xcb_query_keymap_reply [LSB]	xcb_query_keymap_unchecked [LSB]
xcb_query_pointer [LSB]	xcb_query_pointer_reply [LSB]	xcb_query_pointer_unchecked [LSB]	xcb_query_text_extents [LSB]
xcb_query_text_extents_reply [LSB]	xcb_query_text_extents_unchecked [LSB]	xcb_query_tree [LSB]	xcb_query_tree_children [LSB]

xcb_query_tree_children_end [LSB]	xcb_query_tree_children_length [LSB]	xcb_query_tree_reply [LSB]	xcb_query_tree_unchecked [LSB]
xcb_recolor_cursor [LSB]	xcb_recolor_cursor_checked [LSB]	xcb_rectangle_end [LSB]	xcb_rectangle_next [LSB]
xcb_reparent_window [LSB]	xcb_reparent_window_checked [LSB]	xcb_request_check [Libxcb 1.7]	xcb_rgb_end [LSB]
xcb_rgb_next [LSB]	xcb_rotate_properties [LSB]	xcb_rotate_properties_checked [LSB]	xcb_screen_allowed_depths_iterator [LSB]
xcb_screen_allowed_depths_length [LSB]	xcb_screen_end [LSB]	xcb_screen_next [LSB]	xcb_segment_end [LSB]
xcb_segment_next [LSB]	xcb_send_event [LSB]	xcb_send_event_checked [LSB]	xcb_send_request [Libxcb 1.7]
xcb_set_access_control [LSB]	xcb_set_access_control_checked [LSB]	xcb_set_clip_rectangles [LSB]	xcb_set_clip_rectangles_checked [LSB]
xcb_set_close_down_mode [LSB]	xcb_set_close_down_mode_checked [LSB]	xcb_set_dashes [LSB]	xcb_set_dashes_checked [LSB]
xcb_set_font_path [LSB]	xcb_set_font_path_checked [LSB]	xcb_set_input_focus [LSB]	xcb_set_input_focus_checked [LSB]
xcb_set_modifier_mapping [LSB]	xcb_set_modifier_mapping_reply [LSB]	xcb_set_modifier_mapping_unchecked [LSB]	xcb_set_pointer_mapping [LSB]
xcb_set_pointer_mapping_reply [LSB]	xcb_set_pointer_mapping_unchecked [LSB]	xcb_set_screen_saver [LSB]	xcb_set_screen_saver_checked [LSB]
xcb_set_selection_owner [LSB]	xcb_set_selection_owner_checked [LSB]	xcb_setup_authenticate_end [LSB]	xcb_setup_authenticate_next [LSB]
xcb_setup_authenticate_reason [LSB]	xcb_setup_authenticate_reason_end [LSB]	xcb_setup_authenticate_reason_length [LSB]	xcb_setup_end [LSB]
xcb_setup_failed_end [LSB]	xcb_setup_failed_next [LSB]	xcb_setup_failed_reason [LSB]	xcb_setup_failed_reason_end [LSB]
xcb_setup_failed_reason_length [LSB]	xcb_setup_next [LSB]	xcb_setup_pixmap_formats [LSB]	xcb_setup_pixmap_formats_iterator [LSB]

xcb_setup_pixmap_formats_length [LSB]	xcb_setup_request_authorization_protocol_data [LSB]	xcb_setup_request_authorization_protocol_data_end [LSB]	xcb_setup_request_authorization_protocol_data_length [LSB]
xcb_setup_request_authorization_protocol_name [LSB]	xcb_setup_request_authorization_protocol_name_end [LSB]	xcb_setup_request_authorization_protocol_name_length [LSB]	xcb_setup_request_end [LSB]
xcb_setup_request_next [LSB]	xcb_setup_roots_iterator [LSB]	xcb_setup_roots_length [LSB]	xcb_setup_vendor [LSB]
xcb_setup_vendor_end [LSB]	xcb_setup_vendor_length [LSB]	xcb_store_colors [LSB]	xcb_store_colors_checked [LSB]
xcb_store_named_color [LSB]	xcb_store_named_color_checked [LSB]	xcb_str_end [LSB]	xcb_str_name [LSB]
xcb_str_name_end [LSB]	xcb_str_name_length [LSB]	xcb_str_next [LSB]	xcb_take_socket [LSB]
xcb_timecoord_end [LSB]	xcb_timecoord_next [LSB]	xcb_timestamp_end [LSB]	xcb_timestamp_next [LSB]
xcb_translate_coordinates [LSB]	xcb_translate_coordinates_reply [LSB]	xcb_translate_coordinates_unchecked [LSB]	xcb_ungrab_button [LSB]
xcb_ungrab_button_checked [LSB]	xcb_ungrab_key [LSB]	xcb_ungrab_key_checked [LSB]	xcb_ungrab_keyboard [LSB]
xcb_ungrab_keyboard_checked [LSB]	xcb_ungrab_pointer [LSB]	xcb_ungrab_pointer_checked [LSB]	xcb_ungrab_server [LSB]
xcb_ungrab_server_checked [LSB]	xcb_uninstall_colormap [LSB]	xcb_uninstall_colormap_checked [LSB]	xcb_unmap_subwindows [LSB]
xcb_unmap_subwindows_checked [LSB]	xcb_unmap_window [LSB]	xcb_unmap_window_checked [LSB]	xcb_visualid_end [LSB]
xcb_visualid_next [LSB]	xcb_visualtype_end [LSB]	xcb_visualtype_next [LSB]	xcb_wait_for_event [Libxcb 1.7]
xcb_wait_for_reply [Libxcb 1.7]	xcb_warp_pointer [LSB]	xcb_warp_pointer_checked [LSB]	xcb_window_end [LSB]
xcb_window_next [LSB]	xcb_writev [LSB]	xcb_xc_misc_get_version [LSB]	xcb_xc_misc_get_version_reply [LSB]
xcb_xc_misc_get_version_unchecked [LSB]	xcb_xc_misc_get_xid_list [LSB]	xcb_xc_misc_get_xid_list_ids [LSB]	xcb_xc_misc_get_xid_list_ids_end [LSB]

xcb_xc_misc_get_xid_list_ids_length [LSB]	xcb_xc_misc_get_xid_list_reply [LSB]	xcb_xc_misc_get_xid_list_unchecked [LSB]	xcb_xc_misc_get_xid_range [LSB]
xcb_xc_misc_get_xid_range_reply [LSB]	xcb_xc_misc_get_xid_range_unchecked [LSB]		

An LSB conforming implementation shall provide the generic data interfaces for libxcb interfaces specified in Table 6-26, with the full mandatory functionality as described in the referenced underlying specification.

Table 6-26 libxcb - libxcb interfaces Data Interfaces

xcb_big_requests_id [LSB]	xcb_xc_misc_id [LSB]		

6.20 Data Definitions for libxcb

This section defines global identifiers and their values that are associated with interfaces contained in libxcb. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

6.20.1 xcb/bigreq.h

```
#define XCB_BIGREQUESTS_MAJOR_VERSION 0
#define XCB_BIGREQUESTS_MINOR_VERSION 0
#define XCB_BIGREQUESTS_ENABLE 0

typedef struct {
    unsigned int sequence;
} xcb_big_requests_enable_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t minor_opcode;
    uint16_t length;
} xcb_big_requests_enable_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t maximum_request_length;
} xcb_big_requests_enable_reply_t;
extern xcb_big_requests_enable_cookie_t
xcb_big_requests_enable(xcb_connection_t * c);
```

```

extern xcb_big_requests_enable_reply_t
    *xcb_big_requests_enable_reply(xcb_connection_t * c,
                                   xcb_big_requests_enable_cookie_t
                                   cookie,
                                   xcb_generic_error_t * *e);
extern xcb_big_requests_enable_cookie_t
xcb_big_requests_enable_unchecked(xcb_connection_t * c);
extern xcb_extension_t xcb_big_requests_id;

```

6.20.2 xcb/xcb_misc.h

```

#define XCB_XCMISC_MAJOR_VERSION      1
#define XCB_XCMISC_MINOR_VERSION      1
#define XCB_XC_MISC_GET_VERSION      0
#define XCB_XC_MISC_GET_XID_RANGE     1
#define XCB_XC_MISC_GET_XID_LIST      2

typedef struct {
    unsigned int sequence;
} xcb_xc_misc_get_version_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t minor_opcode;
    uint16_t length;
    uint16_t client_major_version;
    uint16_t client_minor_version;
} xcb_xc_misc_get_version_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t server_major_version;
    uint16_t server_minor_version;
} xcb_xc_misc_get_version_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_xc_misc_get_xid_range_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t minor_opcode;
    uint16_t length;
} xcb_xc_misc_get_xid_range_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t start_id;
    uint32_t count;
} xcb_xc_misc_get_xid_range_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_xc_misc_get_xid_list_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t minor_opcode;
    uint16_t length;
    uint32_t count;
} xcb_xc_misc_get_xid_list_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
}

```

```

uint32_t ids_len;
uint8_t pad1[20];
} xcb_xc_misc_get_xid_list_reply_t;
extern xcb_xc_misc_get_version_cookie_t
xcb_xc_misc_get_version(xcb_connection_t * c,
                        uint16_t client_major_version,
                        uint16_t client_minor_version);
extern xcb_xc_misc_get_version_reply_t
*xcb_xc_misc_get_version_reply(xcb_connection_t * c,
                              xcb_xc_misc_get_version_cookie_t
                              cookie,
                              xcb_generic_error_t * *e);
extern xcb_xc_misc_get_version_cookie_t
xcb_xc_misc_get_version_unchecked(xcb_connection_t * c,
                                  uint16_t client_major_version,
                                  uint16_t client_minor_version);
extern xcb_xc_misc_get_xid_list_cookie_t
xcb_xc_misc_get_xid_list(xcb_connection_t * c, uint32_t count);
extern uint32_t *xcb_xc_misc_get_xid_list_ids(const

xcb_xc_misc_get_xid_list_reply_t
                                * R);
extern                                xcb_generic_iterator_t
xcb_xc_misc_get_xid_list_ids_end(const

xcb_xc_misc_get_xid_list_reply_t
                                * R);
extern int xcb_xc_misc_get_xid_list_ids_length(const

xcb_xc_misc_get_xid_list_reply_t
                                * R);
extern xcb_xc_misc_get_xid_list_reply_t
*xcb_xc_misc_get_xid_list_reply(xcb_connection_t * c,
                                xcb_xc_misc_get_xid_list_cookie_t
                                cookie, xcb_generic_error_t * *e);
extern xcb_xc_misc_get_xid_list_cookie_t
xcb_xc_misc_get_xid_list_unchecked(xcb_connection_t * c, uint32_t
count);
extern xcb_xc_misc_get_xid_range_cookie_t
xcb_xc_misc_get_xid_range(xcb_connection_t * c);
extern xcb_xc_misc_get_xid_range_reply_t
*xcb_xc_misc_get_xid_range_reply(xcb_connection_t * c,
                                xcb_xc_misc_get_xid_range_cookie_t
                                cookie, xcb_generic_error_t * *e);
extern xcb_xc_misc_get_xid_range_cookie_t
xcb_xc_misc_get_xid_range_unchecked(xcb_connection_t * c);
extern xcb_extension_t xcb_xc_misc_id;

```

6.20.3 xcb/xcb.h

```

#define X_PROTOCOL 11
#define X_PROTOCOL_REVISION 0
#define X_TCP_PORT 6000
#define XCB_TYPE_PAD(T,I) ((I) & (sizeof(T) > 4 ? 3 :
sizeof(T) - 1))
#define XCB_NONE 0L
#define XCB_COPY_FROM_PARENT 0L
#define XCB_CURRENT_TIME 0L
#define XCB_NO_SYMBOL 0L

typedef struct xcb_connection_t xcb_connection_t;
typedef struct {
    void *data;
    int rem;
    int index;

```



```

} xcb_generic_iterator_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t pad[7];
    uint32_t full_sequence;
} xcb_generic_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t error_code;
    uint16_t sequence;
    uint32_t resource_id;
    uint16_t minor_code;
    uint8_t major_code;
    uint8_t pad0;
    uint32_t pad[5];
    uint32_t full_sequence;
} xcb_generic_error_t;
typedef struct {
    unsigned int sequence;
} xcb_void_cookie_t;
typedef struct {
    int namelen;
    char *name;
    int datalen;
    char *data;
} xcb_auth_info_t;
typedef struct xcb_extension_t xcb_extension_t;
extern xcb_connection_t *xcb_connect(const char *displayname,
                                     int *screenp);

extern xcb_connection_t
xcb_connect_to_display_with_auth_info(const char *display,
                                     xcb_auth_info_t
                                     * auth,
                                     int
                                     *screen);

extern xcb_connection_t *xcb_connect_to_fd(int fd,
                                           xcb_auth_info_t * auth_info);
extern int xcb_connection_has_error(xcb_connection_t * c);
extern void xcb_discard_reply(xcb_connection_t * c, unsigned int
sequence);
extern void xcb_disconnect(xcb_connection_t * c);
extern int xcb_flush(xcb_connection_t * c);
extern uint32_t xcb_generate_id(xcb_connection_t * c);
extern const xcb_query_extension_reply_t
*xcb_get_extension_data(xcb_connection_t * c,
                       struct xcb_extension_t *ext);
extern int xcb_get_file_descriptor(xcb_connection_t * c);
extern uint32_t xcb_get_maximum_request_length(xcb_connection_t *
c);
extern const xcb_setup_t *xcb_get_setup(xcb_connection_t * c);
extern int xcb_parse_display(const char *name, char **host, int
*display,
                             int *screen);
extern xcb_generic_event_t *xcb_poll_for_event(xcb_connection_t *
c);
extern void xcb_prefetch_extension_data(xcb_connection_t * c,
                                       struct xcb_extension_t *ext);
extern void xcb_prefetch_maximum_request_length(xcb_connection_t *
c);
extern xcb_generic_error_t *xcb_request_check(xcb_connection_t * c,
                                             xcb_void_cookie_t cookie);

```

```
extern xcb_generic_event_t *xcb_wait_for_event(xcb_connection_t *
c);
```

6.20.4 xcb/xcbext.h

```
struct xcb_extension_t {
    const char *name;
    int global_id;
};
typedef struct {
    size_t count;
    struct xcb_extension_t *ext;
    uint8_t opcode;
    uint8_t isvoid;
} xcb_protocol_request_t;
enum xcb_send_request_flags_t {
    XCB_REQUEST_CHECKED = 1 << 0,
    XCB_REQUEST_RAW = 1 << 1,
    XCB_REQUEST_DISCARD_REPLY = 1 << 2
};
extern int xcb_poll_for_reply(xcb_connection_t * c, unsigned int
request,
                                void **reply, xcb_generic_error_t *
*error);
extern int xcb_popcount(uint32_t mask);
extern unsigned int xcb_send_request(xcb_connection_t * c, int
flags,
                                struct iovec *vector,
                                const xcb_protocol_request_t *
request);
extern int xcb_take_socket(xcb_connection_t * c,
                                void (*return_socket)(void *), void
*closure,
                                int flags, uint64_t * sent);
extern void *xcb_wait_for_reply(xcb_connection_t * c, unsigned int
request,
                                xcb_generic_error_t * *e);
extern int xcb_writev(xcb_connection_t * c, struct iovec *vector,
int count, uint64_t requests);
```

6.20.5 xcb/xproto.h

```
#define XCB_KEY_PRESS 2
#define XCB_KEY_RELEASE 3
#define XCB_BUTTON_PRESS 4
#define XCB_BUTTON_RELEASE 5
#define XCB_MOTION_NOTIFY 6
#define XCB_ENTER_NOTIFY 7
#define XCB_LEAVE_NOTIFY 8
#define XCB_FOCUS_IN 9
#define XCB_FOCUS_OUT 10
#define XCB_KEYMAP_NOTIFY 11
#define XCB_EXPOSE 12
#define XCB_GRAPHICS_EXPOSURE 13
#define XCB_NO_EXPOSURE 14
#define XCB_VISIBILITY_NOTIFY 15
#define XCB_CREATE_NOTIFY 16
#define XCB_DESTROY_NOTIFY 17
#define XCB_UNMAP_NOTIFY 18
#define XCB_MAP_NOTIFY 19
#define XCB_MAP_REQUEST 20
#define XCB_REPARENT_NOTIFY 21
#define XCB_CONFIGURE_NOTIFY 22
```

```

#define XCB_CONFIGURE_REQUEST 23
#define XCB_GRAVITY_NOTIFY 24
#define XCB_RESIZE_REQUEST 25
#define XCB_CIRCULATE_NOTIFY 26
#define XCB_CIRCULATE_REQUEST 27
#define XCB_PROPERTY_NOTIFY 28
#define XCB_SELECTION_CLEAR 29
#define XCB_SELECTION_REQUEST 30
#define XCB_SELECTION_NOTIFY 31
#define XCB_COLORMAP_NOTIFY 32
#define XCB_CLIENT_MESSAGE 33
#define XCB_MAPPING_NOTIFY 34
#define XCB_REQUEST 1
#define XCB_VALUE 2
#define XCB_WINDOW 3
#define XCB_PIXMAP 4
#define XCB_ATOM 5
#define XCB_CURSOR 6
#define XCB_FONT 7
#define XCB_MATCH 8
#define XCB_DRAWABLE 9
#define XCB_ACCESS 10
#define XCB_ALLOC 11
#define XCB_COLORMAP 12
#define XCB_G_CONTEXT 13
#define XCB_ID_CHOICE 14
#define XCB_NAME 15
#define XCB_LENGTH 16
#define XCB_IMPLEMENTATION 17
#define XCB_CREATE_WINDOW 1
#define XCB_CHANGE_WINDOW_ATTRIBUTES 2
#define XCB_GET_WINDOW_ATTRIBUTES 3
#define XCB_DESTROY_WINDOW 4
#define XCB_DESTROY_SUBWINDOWS 5
#define XCB_CHANGE_SAVE_SET 6
#define XCB_REPARENT_WINDOW 7
#define XCB_MAP_WINDOW 8
#define XCB_MAP_SUBWINDOWS 9
#define XCB_UNMAP_WINDOW 10
#define XCB_UNMAP_SUBWINDOWS 11
#define XCB_CONFIGURE_WINDOW 12
#define XCB_CIRCULATE_WINDOW 13
#define XCB_GET_GEOMETRY 14
#define XCB_QUERY_TREE 15
#define XCB_INTERN_ATOM 16
#define XCB_GET_ATOM_NAME 17
#define XCB_CHANGE_PROPERTY 18
#define XCB_DELETE_PROPERTY 19
#define XCB_GET_PROPERTY 20
#define XCB_LIST_PROPERTIES 21
#define XCB_SET_SELECTION_OWNER 22
#define XCB_GET_SELECTION_OWNER 23
#define XCB_CONVERT_SELECTION 24
#define XCB_SEND_EVENT 25
#define XCB_GRAB_POINTER 26
#define XCB_UNGRAB_POINTER 27
#define XCB_GRAB_BUTTON 28
#define XCB_UNGRAB_BUTTON 29
#define XCB_CHANGE_ACTIVE_POINTER_GRAB 30
#define XCB_GRAB_KEYBOARD 31
#define XCB_UNGRAB_KEYBOARD 32
#define XCB_GRAB_KEY 33
#define XCB_UNGRAB_KEY 34
#define XCB_ALLOW_EVENTS 35
#define XCB_GRAB_SERVER 36
#define XCB_UNGRAB_SERVER 37

```

```

#define XCB_QUERY_POINTER          38
#define XCB_GET_MOTION_EVENTS      39
#define XCB_TRANSLATE_COORDINATES  40
#define XCB_WARP_POINTER           41
#define XCB_SET_INPUT_FOCUS        42
#define XCB_GET_INPUT_FOCUS        43
#define XCB_QUERY_KEYMAP           44
#define XCB_OPEN_FONT              45
#define XCB_CLOSE_FONT             46
#define XCB_QUERY_FONT             47
#define XCB_QUERY_TEXT_EXTENTS     48
#define XCB_LIST_FONTS             49
#define XCB_LIST_FONTS_WITH_INFO   50
#define XCB_SET_FONT_PATH          51
#define XCB_GET_FONT_PATH          52
#define XCB_CREATE_PIXMAP          53
#define XCB_FREE_PIXMAP            54
#define XCB_CREATE_GC              55
#define XCB_CHANGE_GC              56
#define XCB_COPY_GC                57
#define XCB_SET_DASHES             58
#define XCB_SET_CLIP_RECTANGLES    59
#define XCB_FREE_GC                60
#define XCB_CLEAR_AREA             61
#define XCB_COPY_AREA              62
#define XCB_COPY_PLANE             63
#define XCB_POLY_POINT             64
#define XCB_POLY_LINE              65
#define XCB_POLY_SEGMENT           66
#define XCB_POLY_RECTANGLE         67
#define XCB_POLY_ARC               68
#define XCB_FILL_POLY              69
#define XCB_POLY_FILL_RECTANGLE    70
#define XCB_POLY_FILL_ARC          71
#define XCB_PUT_IMAGE              72
#define XCB_GET_IMAGE              73
#define XCB_POLY_TEXT_8            74
#define XCB_POLY_TEXT_16           75
#define XCB_IMAGE_TEXT_8           76
#define XCB_IMAGE_TEXT_16          77
#define XCB_CREATE_COLORMAP        78
#define XCB_FREE_COLORMAP          79
#define XCB_COPY_COLORMAP_AND_FREE 80
#define XCB_INSTALL_COLORMAP       81
#define XCB_UNINSTALL_COLORMAP     82
#define XCB_LIST_INSTALLED_COLORMAPS 83
#define XCB_ALLOC_COLOR            84
#define XCB_ALLOC_NAMED_COLOR      85
#define XCB_ALLOC_COLOR_CELLS      86
#define XCB_ALLOC_COLOR_PLANES     87
#define XCB_FREE_COLORS            88
#define XCB_STORE_COLORS           89
#define XCB_STORE_NAMED_COLOR      90
#define XCB_QUERY_COLORS           91
#define XCB_LOOKUP_COLOR           92
#define XCB_CREATE_CURSOR          93
#define XCB_CREATE_GLYPH_CURSOR    94
#define XCB_FREE_CURSOR            95
#define XCB_RECOLOR_CURSOR         96
#define XCB_QUERY_BEST_SIZE        97
#define XCB_QUERY_EXTENSION        98
#define XCB_LIST_EXTENSIONS        99
#define XCB_CHANGE_KEYBOARD_MAPPING 100
#define XCB_GET_KEYBOARD_MAPPING    101
#define XCB_CHANGE_KEYBOARD_CONTROL 102
#define XCB_GET_KEYBOARD_CONTROL    103

```

```

#define XCB_BELL 104
#define XCB_CHANGE_POINTER_CONTROL 105
#define XCB_GET_POINTER_CONTROL 106
#define XCB_SET_SCREEN_SAVER 107
#define XCB_GET_SCREEN_SAVER 108
#define XCB_CHANGE_HOSTS 109
#define XCB_LIST_HOSTS 110
#define XCB_SET_ACCESS_CONTROL 111
#define XCB_SET_CLOSE_DOWN_MODE 112
#define XCB_KILL_CLIENT 113
#define XCB_ROTATE_PROPERTIES 114
#define XCB_FORCE_SCREEN_SAVER 115
#define XCB_SET_POINTER_MAPPING 116
#define XCB_GET_POINTER_MAPPING 117
#define XCB_SET_MODIFIER_MAPPING 118
#define XCB_GET_MODIFIER_MAPPING 119
#define XCB_NO_OPERATION 127

typedef struct {
    uint8_t byte1;
    uint8_t byte2;
} xcb_char2b_t;
typedef struct {
    xcb_char2b_t *data;
    int rem;
    int index;
} xcb_char2b_iterator_t;
typedef uint32_t xcb_window_t;
typedef struct {
    xcb_window_t *data;
    int rem;
    int index;
} xcb_window_iterator_t;
typedef uint32_t xcb_pixmap_t;
typedef struct {
    xcb_pixmap_t *data;
    int rem;
    int index;
} xcb_pixmap_iterator_t;
typedef uint32_t xcb_cursor_t;
typedef struct {
    xcb_cursor_t *data;
    int rem;
    int index;
} xcb_cursor_iterator_t;
typedef uint32_t xcb_font_t;
typedef struct {
    xcb_font_t *data;
    int rem;
    int index;
} xcb_font_iterator_t;
typedef uint32_t xcb_gcontext_t;
typedef struct {
    xcb_gcontext_t *data;
    int rem;
    int index;
} xcb_gcontext_iterator_t;
typedef uint32_t xcb_colormap_t;
typedef struct {
    xcb_colormap_t *data;
    int rem;
    int index;
} xcb_colormap_iterator_t;
typedef uint32_t xcb_atom_t;
typedef struct {
    xcb_atom_t *data;

```

```

        int rem;
        int index;
    } xcb_atom_iterator_t;
typedef uint32_t xcb_drawable_t;
typedef struct {
    xcb_drawable_t *data;
    int rem;
    int index;
} xcb_drawable_iterator_t;
typedef uint32_t xcb_fontable_t;
typedef struct {
    xcb_fontable_t *data;
    int rem;
    int index;
} xcb_fontable_iterator_t;
typedef uint32_t xcb_visualid_t;
typedef struct {
    xcb_visualid_t *data;
    int rem;
    int index;
} xcb_visualid_iterator_t;
typedef uint32_t xcb_timestamp_t;
typedef struct {
    xcb_timestamp_t *data;
    int rem;
    int index;
} xcb_timestamp_iterator_t;
typedef uint32_t xcb_keysym_t;
typedef struct {
    xcb_keysym_t *data;
    int rem;
    int index;
} xcb_keysym_iterator_t;
typedef uint8_t xcb_keycode_t;
typedef struct {
    xcb_keycode_t *data;
    int rem;
    int index;
} xcb_keycode_iterator_t;
typedef uint8_t xcb_button_t;
typedef struct {
    xcb_button_t *data;
    int rem;
    int index;
} xcb_button_iterator_t;
typedef struct {
    int16_t x;
    int16_t y;
} xcb_point_t;
typedef struct {
    xcb_point_t *data;
    int rem;
    int index;
} xcb_point_iterator_t;
typedef struct {
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
} xcb_rectangle_t;
typedef struct {
    xcb_rectangle_t *data;
    int rem;
    int index;
} xcb_rectangle_iterator_t;
typedef struct {

```

```

    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
    int16_t angle1;
    int16_t angle2;
} xcb_arc_t;
typedef struct {
    xcb_arc_t *data;
    int rem;
    int index;
} xcb_arc_iterator_t;
typedef struct {
    uint8_t depth;
    uint8_t bits_per_pixel;
    uint8_t scanline_pad;
    uint8_t pad0[5];
} xcb_format_t;
typedef struct {
    xcb_format_t *data;
    int rem;
    int index;
} xcb_format_iterator_t;
typedef enum xcb_visual_class_t {
    XCB_VISUAL_CLASS_STATIC_GRAY = 0,
    XCB_VISUAL_CLASS_GRAY_SCALE = 1,
    XCB_VISUAL_CLASS_STATIC_COLOR = 2,
    XCB_VISUAL_CLASS_PSEUDO_COLOR = 3,
    XCB_VISUAL_CLASS_TRUE_COLOR = 4,
    XCB_VISUAL_CLASS_DIRECT_COLOR = 5
} xcb_visual_class_t;
typedef struct {
    xcb_visualid_t visual_id;
    uint8_t _class;
    uint8_t bits_per_rgb_value;
    uint16_t colormap_entries;
    uint32_t red_mask;
    uint32_t green_mask;
    uint32_t blue_mask;
    uint8_t pad0[4];
} xcb_visualtype_t;
typedef struct {
    xcb_visualtype_t *data;
    int rem;
    int index;
} xcb_visualtype_iterator_t;
typedef struct {
    uint8_t depth;
    uint8_t pad0;
    uint16_t visuals_len;
    uint8_t pad1[4];
} xcb_depth_t;
typedef struct {
    xcb_depth_t *data;
    int rem;
    int index;
} xcb_depth_iterator_t;
typedef enum xcb_event_mask_t {
    XCB_EVENT_MASK_NO_EVENT = 0,
    XCB_EVENT_MASK_KEY_PRESS = 1,
    XCB_EVENT_MASK_KEY_RELEASE = 2,
    XCB_EVENT_MASK_BUTTON_PRESS = 4,
    XCB_EVENT_MASK_BUTTON_RELEASE = 8,
    XCB_EVENT_MASK_ENTER_WINDOW = 16,
    XCB_EVENT_MASK_LEAVE_WINDOW = 32,
    XCB_EVENT_MASK_POINTER_MOTION = 64,

```

```

XCB_EVENT_MASK_POINTER_MOTION_HINT = 128,
XCB_EVENT_MASK_BUTTON_1_MOTION = 256,
XCB_EVENT_MASK_BUTTON_2_MOTION = 512,
XCB_EVENT_MASK_BUTTON_3_MOTION = 1024,
XCB_EVENT_MASK_BUTTON_4_MOTION = 2048,
XCB_EVENT_MASK_BUTTON_5_MOTION = 4096,
XCB_EVENT_MASK_BUTTON_MOTION = 8192,
XCB_EVENT_MASK_KEYMAP_STATE = 16384,
XCB_EVENT_MASK_EXPOSURE = 32768,
XCB_EVENT_MASK_VISIBILITY_CHANGE = 65536,
XCB_EVENT_MASK_STRUCTURE_NOTIFY = 131072,
XCB_EVENT_MASK_RESIZE_REDIRECT = 262144,
XCB_EVENT_MASK_SUBSTRUCTURE_NOTIFY = 524288,
XCB_EVENT_MASK_SUBSTRUCTURE_REDIRECT = 1048576,
XCB_EVENT_MASK_FOCUS_CHANGE = 2097152,
XCB_EVENT_MASK_PROPERTY_CHANGE = 4194304,
XCB_EVENT_MASK_COLOR_MAP_CHANGE = 8388608,
XCB_EVENT_MASK_OWNER_GRAB_BUTTON = 16777216
} xcb_event_mask_t;
typedef enum xcb_backing_store_t {
    XCB_BACKING_STORE_NOT_USEFUL = 0,
    XCB_BACKING_STORE_WHEN_MAPPED = 1,
    XCB_BACKING_STORE_ALWAYS = 2
} xcb_backing_store_t;
typedef struct {
    xcb_window_t root;
    xcb_colormap_t default_colormap;
    uint32_t white_pixel;
    uint32_t black_pixel;
    uint32_t current_input_masks;
    uint16_t width_in_pixels;
    uint16_t height_in_pixels;
    uint16_t width_in_millimeters;
    uint16_t height_in_millimeters;
    uint16_t min_installed_maps;
    uint16_t max_installed_maps;
    xcb_visualid_t root_visual;
    uint8_t backing_stores;
    uint8_t save_unders;
    uint8_t root_depth;
    uint8_t allowed_depths_len;
} xcb_screen_t;
typedef struct {
    xcb_screen_t *data;
    int rem;
    int index;
} xcb_screen_iterator_t;
typedef struct {
    uint8_t byte_order;
    uint8_t pad0;
    uint16_t protocol_major_version;
    uint16_t protocol_minor_version;
    uint16_t authorization_protocol_name_len;
    uint16_t authorization_protocol_data_len;
    uint8_t pad1[2];
} xcb_setup_request_t;
typedef struct {
    xcb_setup_request_t *data;
    int rem;
    int index;
} xcb_setup_request_iterator_t;
typedef struct {
    uint8_t status;
    uint8_t reason_len;
    uint16_t protocol_major_version;
    uint16_t protocol_minor_version;

```



```

    uint16_t length;
} xcb_setup_failed_t;
typedef struct {
    xcb_setup_failed_t *data;
    int rem;
    int index;
} xcb_setup_failed_iterator_t;
typedef struct {
    uint8_t status;
    uint8_t pad0[5];
    uint16_t length;
} xcb_setup_authenticate_t;
typedef struct {
    xcb_setup_authenticate_t *data;
    int rem;
    int index;
} xcb_setup_authenticate_iterator_t;
typedef enum xcb_image_order_t {
    XCB_IMAGE_ORDER_LSB_FIRST = 0,
    XCB_IMAGE_ORDER_MSB_FIRST = 1
} xcb_image_order_t;
typedef struct {
    uint8_t status;
    uint8_t pad0;
    uint16_t protocol_major_version;
    uint16_t protocol_minor_version;
    uint16_t length;
    uint32_t release_number;
    uint32_t resource_id_base;
    uint32_t resource_id_mask;
    uint32_t motion_buffer_size;
    uint16_t vendor_len;
    uint16_t maximum_request_length;
    uint8_t roots_len;
    uint8_t pixmap_formats_len;
    uint8_t image_byte_order;
    uint8_t bitmap_format_bit_order;
    uint8_t bitmap_format_scanline_unit;
    uint8_t bitmap_format_scanline_pad;
    xcb_keycode_t min_keycode;
    xcb_keycode_t max_keycode;
    uint8_t pad1[4];
} xcb_setup_t;
typedef struct {
    xcb_setup_t *data;
    int rem;
    int index;
} xcb_setup_iterator_t;
typedef enum xcb_mod_mask_t {
    XCB_MOD_MASK_SHIFT = 1,
    XCB_MOD_MASK_LOCK = 2,
    XCB_MOD_MASK_CONTROL = 4,
    XCB_MOD_MASK_1 = 8,
    XCB_MOD_MASK_2 = 16,
    XCB_MOD_MASK_3 = 32,
    XCB_MOD_MASK_4 = 64,
    XCB_MOD_MASK_5 = 128,
    XCB_MOD_MASK_ANY = 32768
} xcb_mod_mask_t;
typedef enum xcb_key_but_mask_t {
    XCB_KEY_BUT_MASK_SHIFT = 1,
    XCB_KEY_BUT_MASK_LOCK = 2,
    XCB_KEY_BUT_MASK_CONTROL = 4,
    XCB_KEY_BUT_MASK_MOD_1 = 8,
    XCB_KEY_BUT_MASK_MOD_2 = 16,
    XCB_KEY_BUT_MASK_MOD_3 = 32,

```

```

        XCB_KEY_BUT_MASK_MOD_4 = 64,
        XCB_KEY_BUT_MASK_MOD_5 = 128,
        XCB_KEY_BUT_MASK_BUTTON_1 = 256,
        XCB_KEY_BUT_MASK_BUTTON_2 = 512,
        XCB_KEY_BUT_MASK_BUTTON_3 = 1024,
        XCB_KEY_BUT_MASK_BUTTON_4 = 2048,
        XCB_KEY_BUT_MASK_BUTTON_5 = 4096
    } xcb_key_but_mask_t;
typedef enum xcb_window_enum_t {
    XCB_WINDOW_NONE = 0
} xcb_window_enum_t;
typedef struct {
    uint8_t response_type;
    xcb_keycode_t detail;
    uint16_t sequence;
    xcb_timestamp_t time;
    xcb_window_t root;
    xcb_window_t event;
    xcb_window_t child;
    int16_t root_x;
    int16_t root_y;
    int16_t event_x;
    int16_t event_y;
    uint16_t state;
    uint8_t same_screen;
    uint8_t pad0;
} xcb_key_press_event_t;
typedef xcb_key_press_event_t xcb_key_release_event_t;
typedef enum xcb_button_mask_t {
    XCB_BUTTON_MASK_1 = 256,
    XCB_BUTTON_MASK_2 = 512,
    XCB_BUTTON_MASK_3 = 1024,
    XCB_BUTTON_MASK_4 = 2048,
    XCB_BUTTON_MASK_5 = 4096,
    XCB_BUTTON_MASK_ANY = 32768
} xcb_button_mask_t;
typedef struct {
    uint8_t response_type;
    xcb_button_t detail;
    uint16_t sequence;
    xcb_timestamp_t time;
    xcb_window_t root;
    xcb_window_t event;
    xcb_window_t child;
    int16_t root_x;
    int16_t root_y;
    int16_t event_x;
    int16_t event_y;
    uint16_t state;
    uint8_t same_screen;
    uint8_t pad0;
} xcb_button_press_event_t;
typedef xcb_button_press_event_t xcb_button_release_event_t;
typedef enum xcb_motion_t {
    XCB_MOTION_NORMAL = 0,
    XCB_MOTION_HINT = 1
} xcb_motion_t;
typedef struct {
    uint8_t response_type;
    uint8_t detail;
    uint16_t sequence;
    xcb_timestamp_t time;
    xcb_window_t root;
    xcb_window_t event;
    xcb_window_t child;
    int16_t root_x;

```

```

    int16_t root_y;
    int16_t event_x;
    int16_t event_y;
    uint16_t state;
    uint8_t same_screen;
    uint8_t pad0;
} xcb_motion_notify_event_t;
typedef enum xcb_notify_detail_t {
    XCB_NOTIFY_DETAIL_ANCESTOR = 0,
    XCB_NOTIFY_DETAIL_VIRTUAL = 1,
    XCB_NOTIFY_DETAIL_INFERIOR = 2,
    XCB_NOTIFY_DETAIL_NONLINEAR = 3,
    XCB_NOTIFY_DETAIL_NONLINEAR_VIRTUAL = 4,
    XCB_NOTIFY_DETAIL_POINTER = 5,
    XCB_NOTIFY_DETAIL_POINTER_ROOT = 6,
    XCB_NOTIFY_DETAIL_NONE = 7
} xcb_notify_detail_t;
typedef enum xcb_notify_mode_t {
    XCB_NOTIFY_MODE_NORMAL = 0,
    XCB_NOTIFY_MODE_GRAB = 1,
    XCB_NOTIFY_MODE_UNGRAB = 2,
    XCB_NOTIFY_MODE_WHILE_GRABBED = 3
} xcb_notify_mode_t;
typedef struct {
    uint8_t response_type;
    uint8_t detail;
    uint16_t sequence;
    xcb_timestamp_t time;
    xcb_window_t root;
    xcb_window_t event;
    xcb_window_t child;
    int16_t root_x;
    int16_t root_y;
    int16_t event_x;
    int16_t event_y;
    uint16_t state;
    uint8_t mode;
    uint8_t same_screen;
    uint8_t focus;
} xcb_enter_notify_event_t;
typedef xcb_enter_notify_event_t xcb_leave_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t detail;
    uint16_t sequence;
    xcb_window_t event;
    uint8_t mode;
    uint8_t pad0[3];
} xcb_focus_in_event_t;
typedef xcb_focus_in_event_t xcb_focus_out_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t keys[31];
} xcb_keymap_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t window;
    uint16_t x;
    uint16_t y;
    uint16_t width;
    uint16_t height;
    uint16_t count;
    uint8_t pad1[2];
} xcb_expose_event_t;
typedef struct {

```

```

uint8_t response_type;
uint8_t pad0;
uint16_t sequence;
xcb_drawable_t drawable;
uint16_t x;
uint16_t y;
uint16_t width;
uint16_t height;
uint16_t minor_opcode;
uint16_t count;
uint8_t major_opcode;
uint8_t pad1[3];
} xcb_graphics_exposure_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_drawable_t drawable;
    uint16_t minor_opcode;
    uint8_t major_opcode;
    uint8_t pad1;
} xcb_no_exposure_event_t;
typedef enum xcb_visibility_t {
    XCB_VISIBILITY_UNOBSURED = 0,
    XCB_VISIBILITY_PARTIALLY_OBSCURED = 1,
    XCB_VISIBILITY_FULLY_OBSCURED = 2
} xcb_visibility_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t window;
    uint8_t state;
    uint8_t pad1[3];
} xcb_visibility_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t parent;
    xcb_window_t window;
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
    uint16_t border_width;
    uint8_t override_redirect;
    uint8_t pad1;
} xcb_create_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t event;
    xcb_window_t window;
} xcb_destroy_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t event;
    xcb_window_t window;
    uint8_t from_configure;
    uint8_t pad1[3];
} xcb_unmap_notify_event_t;
typedef struct {

```

```

uint8_t response_type;
uint8_t pad0;
uint16_t sequence;
xcb_window_t event;
xcb_window_t window;
uint8_t override_redirect;
uint8_t pad1[3];
} xcb_map_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t parent;
    xcb_window_t window;
} xcb_map_request_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t event;
    xcb_window_t window;
    xcb_window_t parent;
    int16_t x;
    int16_t y;
    uint8_t override_redirect;
    uint8_t pad1[3];
} xcb_reparent_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t event;
    xcb_window_t window;
    xcb_window_t above_sibling;
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
    uint16_t border_width;
    uint8_t override_redirect;
    uint8_t pad1;
} xcb_configure_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t stack_mode;
    uint16_t sequence;
    xcb_window_t parent;
    xcb_window_t window;
    xcb_window_t sibling;
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
    uint16_t border_width;
    uint16_t value_mask;
} xcb_configure_request_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t event;
    xcb_window_t window;
    int16_t x;
    int16_t y;
} xcb_gravity_notify_event_t;
typedef struct {

```

```

uint8_t response_type;
uint8_t pad0;
uint16_t sequence;
xcb_window_t window;
uint16_t width;
uint16_t height;
} xcb_resize_request_event_t;
typedef enum xcb_place_t {
    XCB_PLACE_ON_TOP = 0,
    XCB_PLACE_ON_BOTTOM = 1
} xcb_place_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t event;
    xcb_window_t window;
    uint8_t pad1[4];
    uint8_t place;
    uint8_t pad2[3];
} xcb_circulate_notify_event_t;
typedef xcb_circulate_notify_event_t xcb_circulate_request_event_t;
typedef enum xcb_property_t {
    XCB_PROPERTY_NEW_VALUE = 0,
    XCB_PROPERTY_DELETE = 1
} xcb_property_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t window;
    xcb_atom_t atom;
    xcb_timestamp_t time;
    uint8_t state;
    uint8_t pad1[3];
} xcb_property_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_timestamp_t time;
    xcb_window_t owner;
    xcb_atom_t selection;
} xcb_selection_clear_event_t;
typedef enum xcb_time_t {
    XCB_TIME_CURRENT_TIME = 0
} xcb_time_t;
typedef enum xcb_atom_enum_t {
    XCB_ATOM_NONE = 0,
    XCB_ATOM_ANY = 0,
    XCB_ATOM_PRIMARY,
    XCB_ATOM_SECONDARY,
    XCB_ATOM_ARC,
    XCB_ATOM_ATOM,
    XCB_ATOM_BITMAP,
    XCB_ATOM_CARDINAL,
    XCB_ATOM_COLORMAP,
    XCB_ATOM_CURSOR,
    XCB_ATOM_CUT_BUFFER0,
    XCB_ATOM_CUT_BUFFER1,
    XCB_ATOM_CUT_BUFFER2,
    XCB_ATOM_CUT_BUFFER3,
    XCB_ATOM_CUT_BUFFER4,
    XCB_ATOM_CUT_BUFFER5,
    XCB_ATOM_CUT_BUFFER6,
    XCB_ATOM_CUT_BUFFER7,

```

```

XCB_ATOM_DRAWABLE,
XCB_ATOM_FONT,
XCB_ATOM_INTEGER,
XCB_ATOM_PIXMAP,
XCB_ATOM_POINT,
XCB_ATOM_RECTANGLE,
XCB_ATOM_RESOURCE_MANAGER,
XCB_ATOM_RGB_COLOR_MAP,
XCB_ATOM_RGB_BEST_MAP,
XCB_ATOM_RGB_BLUE_MAP,
XCB_ATOM_RGB_DEFAULT_MAP,
XCB_ATOM_RGB_GRAY_MAP,
XCB_ATOM_RGB_GREEN_MAP,
XCB_ATOM_RGB_RED_MAP,
XCB_ATOM_STRING,
XCB_ATOM_VISUALID,
XCB_ATOM_WINDOW,
XCB_ATOM_WM_COMMAND,
XCB_ATOM_WM_HINTS,
XCB_ATOM_WM_CLIENT_MACHINE,
XCB_ATOM_WM_ICON_NAME,
XCB_ATOM_WM_ICON_SIZE,
XCB_ATOM_WM_NAME,
XCB_ATOM_WM_NORMAL_HINTS,
XCB_ATOM_WM_SIZE_HINTS,
XCB_ATOM_WM_ZOOM_HINTS,
XCB_ATOM_MIN_SPACE,
XCB_ATOM_NORM_SPACE,
XCB_ATOM_MAX_SPACE,
XCB_ATOM_END_SPACE,
XCB_ATOM_SUPERSCRIPT_X,
XCB_ATOM_SUPERSCRIPT_Y,
XCB_ATOM_SUBSCRIPT_X,
XCB_ATOM_SUBSCRIPT_Y,
XCB_ATOM_UNDERLINE_POSITION,
XCB_ATOM_UNDERLINE_THICKNESS,
XCB_ATOM_STRIKEOUT_ASCENT,
XCB_ATOM_STRIKEOUT_DESCENT,
XCB_ATOM_ITALIC_ANGLE,
XCB_ATOM_X_HEIGHT,
XCB_ATOM_QUAD_WIDTH,
XCB_ATOM_WEIGHT,
XCB_ATOM_POINT_SIZE,
XCB_ATOM_RESOLUTION,
XCB_ATOM_COPYRIGHT,
XCB_ATOM_NOTICE,
XCB_ATOM_FONT_NAME,
XCB_ATOM_FAMILY_NAME,
XCB_ATOM_FULL_NAME,
XCB_ATOM_CAP_HEIGHT,
XCB_ATOM_WM_CLASS,
XCB_ATOM_WM_TRANSIENT_FOR
} xcb_atom_enum_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_timestamp_t time;
    xcb_window_t owner;
    xcb_window_t requestor;
    xcb_atom_t selection;
    xcb_atom_t target;
    xcb_atom_t property;
} xcb_selection_request_event_t;
typedef struct {
    uint8_t response_type;

```

```

uint8_t pad0;
uint16_t sequence;
xcb_timestamp_t time;
xcb_window_t requestor;
xcb_atom_t selection;
xcb_atom_t target;
xcb_atom_t property;
} xcb_selection_notify_event_t;
typedef enum xcb_colormap_state_t {
    XCB_COLORMAP_STATE_UNINSTALLED = 0,
    XCB_COLORMAP_STATE_INSTALLED = 1
} xcb_colormap_state_t;
typedef enum xcb_colormap_enum_t {
    XCB_COLORMAP_NONE = 0
} xcb_colormap_enum_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    xcb_window_t window;
    xcb_colormap_t colormap;
    uint8_t _new;
    uint8_t state;
    uint8_t pad1[2];
} xcb_colormap_notify_event_t;
typedef union {
    uint8_t data8[20];
    uint16_t data16[10];
    uint32_t data32[5];
} xcb_client_message_data_t;
typedef struct {
    xcb_client_message_data_t *data;
    int rem;
    int index;
} xcb_client_message_data_iterator_t;
typedef struct {
    uint8_t response_type;
    uint8_t format;
    uint16_t sequence;
    xcb_window_t window;
    xcb_atom_t type;
    xcb_client_message_data_t data;
} xcb_client_message_event_t;
typedef enum xcb_mapping_t {
    XCB_MAPPING_MODIFIER = 0,
    XCB_MAPPING_KEYBOARD = 1,
    XCB_MAPPING_POINTER = 2
} xcb_mapping_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint8_t request;
    xcb_keycode_t first_keycode;
    uint8_t count;
    uint8_t pad1;
} xcb_mapping_notify_event_t;
typedef struct {
    uint8_t response_type;
    uint8_t error_code;
    uint16_t sequence;
    uint32_t bad_value;
    uint16_t minor_opcode;
    uint8_t major_opcode;
    uint8_t pad0;
} xcb_request_error_t;

```



```

typedef struct {
    uint8_t response_type;
    uint8_t error_code;
    uint16_t sequence;
    uint32_t bad_value;
    uint16_t minor_opcode;
    uint8_t major_opcode;
    uint8_t pad0;
} xcb_value_error_t;
typedef xcb_value_error_t xcb_window_error_t;
typedef xcb_value_error_t xcb_pixmap_error_t;
typedef xcb_value_error_t xcb_atom_error_t;
typedef xcb_value_error_t xcb_cursor_error_t;
typedef xcb_value_error_t xcb_font_error_t;
typedef xcb_request_error_t xcb_match_error_t;
typedef xcb_value_error_t xcb_drawable_error_t;
typedef xcb_request_error_t xcb_access_error_t;
typedef xcb_request_error_t xcb_alloc_error_t;
typedef xcb_value_error_t xcb_colormap_error_t;
typedef xcb_value_error_t xcb_g_context_error_t;
typedef xcb_value_error_t xcb_id_choice_error_t;
typedef xcb_request_error_t xcb_name_error_t;
typedef xcb_request_error_t xcb_length_error_t;
typedef xcb_request_error_t xcb_implementation_error_t;
typedef enum xcb_window_class_t {
    XCB_WINDOW_CLASS_COPY_FROM_PARENT = 0,
    XCB_WINDOW_CLASS_INPUT_OUTPUT = 1,
    XCB_WINDOW_CLASS_INPUT_ONLY = 2
} xcb_window_class_t;
typedef enum xcb_cw_t {
    XCB_CW_BACK_PIXMAP = 1,
    XCB_CW_BACK_PIXEL = 2,
    XCB_CW_BORDER_PIXMAP = 4,
    XCB_CW_BORDER_PIXEL = 8,
    XCB_CW_BIT_GRAVITY = 16,
    XCB_CW_WIN_GRAVITY = 32,
    XCB_CW_BACKING_STORE = 64,
    XCB_CW_BACKING_PLANES = 128,
    XCB_CW_BACKING_PIXEL = 256,
    XCB_CW_OVERRIDE_REDIRECT = 512,
    XCB_CW_SAVE_UNDER = 1024,
    XCB_CW_EVENT_MASK = 2048,
    XCB_CW_DONT_PROPAGATE = 4096,
    XCB_CW_COLORMAP = 8192,
    XCB_CW_CURSOR = 16384
} xcb_cw_t;
typedef enum xcb_back_pixmap_t {
    XCB_BACK_PIXMAP_NONE = 0,
    XCB_BACK_PIXMAP_PARENT_RELATIVE = 1
} xcb_back_pixmap_t;
typedef enum xcb_gravity_t {
    XCB_GRAVITY_BIT_FORGET = 0,
    XCB_GRAVITY_WIN_UNMAP = 0,
    XCB_GRAVITY_NORTH_WEST = 1,
    XCB_GRAVITY_NORTH = 2,
    XCB_GRAVITY_NORTH_EAST = 3,
    XCB_GRAVITY_WEST = 4,
    XCB_GRAVITY_CENTER = 5,
    XCB_GRAVITY_EAST = 6,
    XCB_GRAVITY_SOUTH_WEST = 7,
    XCB_GRAVITY_SOUTH = 8,
    XCB_GRAVITY_SOUTH_EAST = 9,
    XCB_GRAVITY_STATIC = 10
} xcb_gravity_t;
typedef struct {
    uint8_t major_opcode;

```

```

uint8_t depth;
uint16_t length;
xcb_window_t wid;
xcb_window_t parent;
int16_t x;
int16_t y;
uint16_t width;
uint16_t height;
uint16_t border_width;
uint16_t _class;
xcb_visualid_t visual;
uint32_t value_mask;
} xcb_create_window_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
    uint32_t value_mask;
} xcb_change_window_attributes_request_t;
typedef enum xcb_map_state_t {
    XCB_MAP_STATE_UNMAPPED = 0,
    XCB_MAP_STATE_UNVIEWABLE = 1,
    XCB_MAP_STATE_VIEWABLE = 2
} xcb_map_state_t;
typedef struct {
    unsigned int sequence;
} xcb_get_window_attributes_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_get_window_attributes_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t backing_store;
    uint16_t sequence;
    uint32_t length;
    xcb_visualid_t visual;
    uint16_t _class;
    uint8_t bit_gravity;
    uint8_t win_gravity;
    uint32_t backing_planes;
    uint32_t backing_pixel;
    uint8_t save_under;
    uint8_t map_is_installed;
    uint8_t map_state;
    uint8_t override_redirect;
    xcb_colormap_t colormap;
    uint32_t all_event_masks;
    uint32_t your_event_mask;
    uint16_t do_not_propagate_mask;
    uint8_t pad0[2];
} xcb_get_window_attributes_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_destroy_window_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;

```

```

} xcb_destroy_subwindows_request_t;
typedef enum xcb_set_mode_t {
    XCB_SET_MODE_INSERT = 0,
    XCB_SET_MODE_DELETE = 1
} xcb_set_mode_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
    xcb_window_t window;
} xcb_change_save_set_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
    xcb_window_t parent;
    int16_t x;
    int16_t y;
} xcb_reparent_window_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_map_window_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_map_subwindows_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_unmap_window_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_unmap_subwindows_request_t;
typedef enum xcb_config_window_t {
    XCB_CONFIG_WINDOW_X = 1,
    XCB_CONFIG_WINDOW_Y = 2,
    XCB_CONFIG_WINDOW_WIDTH = 4,
    XCB_CONFIG_WINDOW_HEIGHT = 8,
    XCB_CONFIG_WINDOW_BORDER_WIDTH = 16,
    XCB_CONFIG_WINDOW_SIBLING = 32,
    XCB_CONFIG_WINDOW_STACK_MODE = 64
} xcb_config_window_t;
typedef enum xcb_stack_mode_t {
    XCB_STACK_MODE_ABOVE = 0,
    XCB_STACK_MODE_BELOW = 1,
    XCB_STACK_MODE_TOP_IF = 2,
    XCB_STACK_MODE_BOTTOM_IF = 3,
    XCB_STACK_MODE_OPPOSITE = 4
} xcb_stack_mode_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
    uint16_t value_mask;

```

```

    uint8_t pad1[2];
} xcb_configure_window_request_t;
typedef enum xcb_circulate_t {
    XCB_CIRCULATE_RAISE_LOWEST = 0,
    XCB_CIRCULATE_LOWER_HIGHEST = 1
} xcb_circulate_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t direction;
    uint16_t length;
    xcb_window_t window;
} xcb_circulate_window_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_geometry_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
} xcb_get_geometry_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t depth;
    uint16_t sequence;
    uint32_t length;
    xcb_window_t root;
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
    uint16_t border_width;
    uint8_t pad0[2];
} xcb_get_geometry_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_query_tree_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_query_tree_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    xcb_window_t root;
    xcb_window_t parent;
    uint16_t children_len;
    uint8_t pad1[14];
} xcb_query_tree_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_intern_atom_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t only_if_exists;
    uint16_t length;
    uint16_t name_len;
    uint8_t pad0[2];
} xcb_intern_atom_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;

```

```

        uint16_t sequence;
        uint32_t length;
        xcb_atom_t atom;
    } xcb_intern_atom_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_get_atom_name_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_atom_t atom;
} xcb_get_atom_name_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t name_len;
    uint8_t pad1[22];
} xcb_get_atom_name_reply_t;
typedef enum xcb_prop_mode_t {
    XCB_PROP_MODE_REPLACE = 0,
    XCB_PROP_MODE_PREPEND = 1,
    XCB_PROP_MODE_APPEND = 2
} xcb_prop_mode_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
    xcb_window_t window;
    xcb_atom_t property;
    xcb_atom_t type;
    uint8_t format;
    uint8_t pad0[3];
    uint32_t data_len;
} xcb_change_property_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
    xcb_atom_t property;
} xcb_delete_property_request_t;
typedef enum xcb_get_property_type_t {
    XCB_GET_PROPERTY_TYPE_ANY = 0
} xcb_get_property_type_t;
typedef struct {
    unsigned int sequence;
} xcb_get_property_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t _delete;
    uint16_t length;
    xcb_window_t window;
    xcb_atom_t property;
    xcb_atom_t type;
    uint32_t long_offset;
    uint32_t long_length;
} xcb_get_property_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t format;
    uint16_t sequence;
    uint32_t length;
    xcb_atom_t type;

```

```

        uint32_t bytes_after;
        uint32_t value_len;
        uint8_t pad0[12];
    } xcb_get_property_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_list_properties_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_list_properties_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t atoms_len;
    uint8_t pad1[22];
} xcb_list_properties_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t owner;
    xcb_atom_t selection;
    xcb_timestamp_t time;
} xcb_set_selection_owner_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_selection_owner_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_atom_t selection;
} xcb_get_selection_owner_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    xcb_window_t owner;
} xcb_get_selection_owner_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t requestor;
    xcb_atom_t selection;
    xcb_atom_t target;
    xcb_atom_t property;
    xcb_timestamp_t time;
} xcb_convert_selection_request_t;
typedef enum xcb_send_event_dest_t {
    XCB_SEND_EVENT_DEST_POINTER_WINDOW = 0,
    XCB_SEND_EVENT_DEST_ITEM_FOCUS = 1
} xcb_send_event_dest_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t propagate;
    uint16_t length;
    xcb_window_t destination;
    uint32_t event_mask;
    char event[32];

```

```

} xcb_send_event_request_t;
typedef enum xcb_grab_mode_t {
    XCB_GRAB_MODE_SYNC = 0,
    XCB_GRAB_MODE_ASYNC = 1
} xcb_grab_mode_t;
typedef enum xcb_grab_status_t {
    XCB_GRAB_STATUS_SUCCESS = 0,
    XCB_GRAB_STATUS_ALREADY_GRABBED = 1,
    XCB_GRAB_STATUS_INVALID_TIME = 2,
    XCB_GRAB_STATUS_NOT_VIEWABLE = 3,
    XCB_GRAB_STATUS_FROZEN = 4
} xcb_grab_status_t;
typedef enum xcb_cursor_enum_t {
    XCB_CURSOR_NONE = 0
} xcb_cursor_enum_t;
typedef struct {
    unsigned int sequence;
} xcb_grab_pointer_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t owner_events;
    uint16_t length;
    xcb_window_t grab_window;
    uint16_t event_mask;
    uint8_t pointer_mode;
    uint8_t keyboard_mode;
    xcb_window_t confine_to;
    xcb_cursor_t cursor;
    xcb_timestamp_t time;
} xcb_grab_pointer_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t status;
    uint16_t sequence;
    uint32_t length;
} xcb_grab_pointer_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_timestamp_t time;
} xcb_ungrab_pointer_request_t;
typedef enum xcb_button_index_t {
    XCB_BUTTON_INDEX_ANY = 0,
    XCB_BUTTON_INDEX_1 = 1,
    XCB_BUTTON_INDEX_2 = 2,
    XCB_BUTTON_INDEX_3 = 3,
    XCB_BUTTON_INDEX_4 = 4,
    XCB_BUTTON_INDEX_5 = 5
} xcb_button_index_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t owner_events;
    uint16_t length;
    xcb_window_t grab_window;
    uint16_t event_mask;
    uint8_t pointer_mode;
    uint8_t keyboard_mode;
    xcb_window_t confine_to;
    xcb_cursor_t cursor;
    uint8_t button;
    uint8_t pad0;
    uint16_t modifiers;
} xcb_grab_button_request_t;
typedef struct {
    uint8_t major_opcode;

```

```

        uint8_t button;
        uint16_t length;
        xcb_window_t grab_window;
        uint16_t modifiers;
        uint8_t pad0[2];
    } xcb_ungrab_button_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_cursor_t cursor;
    xcb_timestamp_t time;
    uint16_t event_mask;
    uint8_t pad1[2];
} xcb_change_active_pointer_grab_request_t;
typedef struct {
    unsigned int sequence;
} xcb_grab_keyboard_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t owner_events;
    uint16_t length;
    xcb_window_t grab_window;
    xcb_timestamp_t time;
    uint8_t pointer_mode;
    uint8_t keyboard_mode;
    uint8_t pad0[2];
} xcb_grab_keyboard_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t status;
    uint16_t sequence;
    uint32_t length;
} xcb_grab_keyboard_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_timestamp_t time;
} xcb_ungrab_keyboard_request_t;
typedef enum xcb_grab_t {
    XCB_GRAB_ANY = 0
} xcb_grab_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t owner_events;
    uint16_t length;
    xcb_window_t grab_window;
    uint16_t modifiers;
    xcb_keycode_t key;
    uint8_t pointer_mode;
    uint8_t keyboard_mode;
    uint8_t pad0[3];
} xcb_grab_key_request_t;
typedef struct {
    uint8_t major_opcode;
    xcb_keycode_t key;
    uint16_t length;
    xcb_window_t grab_window;
    uint16_t modifiers;
    uint8_t pad0[2];
} xcb_ungrab_key_request_t;
typedef enum xcb_allow_t {
    XCB_ALLOW_ASYNC_POINTER = 0,
    XCB_ALLOW_SYNC_POINTER = 1,
    XCB_ALLOW_REPLAY_POINTER = 2,

```



```

XCB_ALLOW_ASYNC_KEYBOARD = 3,
XCB_ALLOW_SYNC_KEYBOARD = 4,
XCB_ALLOW_REPLAY_KEYBOARD = 5,
XCB_ALLOW_ASYNC_BOTH = 6,
XCB_ALLOW_SYNC_BOTH = 7
} xcb_allow_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
    xcb_timestamp_t time;
} xcb_allow_events_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_grab_server_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_ungrab_server_request_t;
typedef struct {
    unsigned int sequence;
} xcb_query_pointer_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_query_pointer_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t same_screen;
    uint16_t sequence;
    uint32_t length;
    xcb_window_t root;
    xcb_window_t child;
    int16_t root_x;
    int16_t root_y;
    int16_t win_x;
    int16_t win_y;
    uint16_t mask;
    uint8_t pad0[2];
} xcb_query_pointer_reply_t;
typedef struct {
    xcb_timestamp_t time;
    int16_t x;
    int16_t y;
} xcb_timecoord_t;
typedef struct {
    xcb_timecoord_t *data;
    int rem;
    int index;
} xcb_timecoord_iterator_t;
typedef struct {
    unsigned int sequence;
} xcb_get_motion_events_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
    xcb_timestamp_t start;
    xcb_timestamp_t stop;
} xcb_get_motion_events_request_t;

```

```

typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t events_len;
    uint8_t pad1[20];
} xcb_get_motion_events_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_translate_coordinates_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t src_window;
    xcb_window_t dst_window;
    int16_t src_x;
    int16_t src_y;
} xcb_translate_coordinates_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t same_screen;
    uint16_t sequence;
    uint32_t length;
    xcb_window_t child;
    uint16_t dst_x;
    uint16_t dst_y;
} xcb_translate_coordinates_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t src_window;
    xcb_window_t dst_window;
    int16_t src_x;
    int16_t src_y;
    uint16_t src_width;
    uint16_t src_height;
    int16_t dst_x;
    int16_t dst_y;
} xcb_warp_pointer_request_t;
typedef enum xcb_input_focus_t {
    XCB_INPUT_FOCUS_NONE = 0,
    XCB_INPUT_FOCUS_POINTER_ROOT = 1,
    XCB_INPUT_FOCUS_PARENT = 2,
    XCB_INPUT_FOCUS_FOLLOW_KEYBOARD = 3
} xcb_input_focus_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t revert_to;
    uint16_t length;
    xcb_window_t focus;
    xcb_timestamp_t time;
} xcb_set_input_focus_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_input_focus_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_input_focus_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t revert_to;

```

```

        uint16_t sequence;
        uint32_t length;
        xcb_window_t focus;
    } xcb_get_input_focus_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_query_keymap_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_query_keymap_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint8_t keys[32];
} xcb_query_keymap_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_font_t fid;
    uint16_t name_len;
    uint8_t pad1[2];
} xcb_open_font_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_font_t font;
} xcb_close_font_request_t;
typedef enum xcb_font_draw_t {
    XCB_FONT_DRAW_LEFT_TO_RIGHT = 0,
    XCB_FONT_DRAW_RIGHT_TO_LEFT = 1
} xcb_font_draw_t;
typedef struct {
    xcb_atom_t name;
    uint32_t value;
} xcb_fontprop_t;
typedef struct {
    xcb_fontprop_t *data;
    int rem;
    int index;
} xcb_fontprop_iterator_t;
typedef struct {
    int16_t left_side_bearing;
    int16_t right_side_bearing;
    int16_t character_width;
    int16_t ascent;
    int16_t descent;
    uint16_t attributes;
} xcb_charinfo_t;
typedef struct {
    xcb_charinfo_t *data;
    int rem;
    int index;
} xcb_charinfo_iterator_t;
typedef struct {
    unsigned int sequence;
} xcb_query_font_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;

```

```

        xcb_fontable_t font;
    } xcb_query_font_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    xcb_charinfo_t min_bounds;
    uint8_t pad1[4];
    xcb_charinfo_t max_bounds;
    uint8_t pad2[4];
    uint16_t min_char_or_byte2;
    uint16_t max_char_or_byte2;
    uint16_t default_char;
    uint16_t properties_len;
    uint8_t draw_direction;
    uint8_t min_byte1;
    uint8_t max_byte1;
    uint8_t all_chars_exist;
    int16_t font_ascent;
    int16_t font_descent;
    uint32_t char_infos_len;
} xcb_query_font_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_query_text_extents_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t odd_length;
    uint16_t length;
    xcb_fontable_t font;
} xcb_query_text_extents_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t draw_direction;
    uint16_t sequence;
    uint32_t length;
    int16_t font_ascent;
    int16_t font_descent;
    int16_t overall_ascent;
    int16_t overall_descent;
    int32_t overall_width;
    int32_t overall_left;
    int32_t overall_right;
} xcb_query_text_extents_reply_t;
typedef struct {
    uint8_t name_len;
} xcb_str_t;
typedef struct {
    xcb_str_t *data;
    int rem;
    int index;
} xcb_str_iterator_t;
typedef struct {
    unsigned int sequence;
} xcb_list_fonts_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    uint16_t max_names;
    uint16_t pattern_len;
} xcb_list_fonts_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;

```

```

        uint16_t sequence;
        uint32_t length;
        uint16_t names_len;
        uint8_t pad1[22];
    } xcb_list_fonts_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_list_fonts_with_info_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    uint16_t max_names;
    uint16_t pattern_len;
} xcb_list_fonts_with_info_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t name_len;
    uint16_t sequence;
    uint32_t length;
    xcb_charinfo_t min_bounds;
    uint8_t pad0[4];
    xcb_charinfo_t max_bounds;
    uint8_t pad1[4];
    uint16_t min_char_or_byte2;
    uint16_t max_char_or_byte2;
    uint16_t default_char;
    uint16_t properties_len;
    uint8_t draw_direction;
    uint8_t min_bytel;
    uint8_t max_bytel;
    uint8_t all_chars_exist;
    int16_t font_ascent;
    int16_t font_descent;
    uint32_t replies_hint;
} xcb_list_fonts_with_info_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    uint16_t font_id;
} xcb_set_font_path_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_font_path_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_font_path_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t path_len;
    uint8_t pad1[22];
} xcb_get_font_path_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t depth;
    uint16_t length;
    xcb_pixmap_t pid;
    xcb_drawable_t drawable;
    uint16_t width;
    uint16_t height;

```

```

} xcb_create_pixmap_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_pixmap_t pixmap;
} xcb_free_pixmap_request_t;
typedef enum xcb_gc_t {
    XCB_GC_FUNCTION = 1,
    XCB_GC_PLANE_MASK = 2,
    XCB_GC_FOREGROUND = 4,
    XCB_GC_BACKGROUND = 8,
    XCB_GC_LINE_WIDTH = 16,
    XCB_GC_LINE_STYLE = 32,
    XCB_GC_CAP_STYLE = 64,
    XCB_GC_JOIN_STYLE = 128,
    XCB_GC_FILL_STYLE = 256,
    XCB_GC_FILL_RULE = 512,
    XCB_GC_TILE = 1024,
    XCB_GC_STIPPLE = 2048,
    XCB_GC_TILE_STIPPLE_ORIGIN_X = 4096,
    XCB_GC_TILE_STIPPLE_ORIGIN_Y = 8192,
    XCB_GC_FONT = 16384,
    XCB_GC_SUBWINDOW_MODE = 32768,
    XCB_GC_GRAPHICS_EXPOSURES = 65536,
    XCB_GC_CLIP_ORIGIN_X = 131072,
    XCB_GC_CLIP_ORIGIN_Y = 262144,
    XCB_GC_CLIP_MASK = 524288,
    XCB_GC_DASH_OFFSET = 1048576,
    XCB_GC_DASH_LIST = 2097152,
    XCB_GC_ARC_MODE = 4194304
} xcb_gc_t;
typedef enum xcb_gx_t {
    XCB_GX_CLEAR = 0,
    XCB_GX_AND = 1,
    XCB_GX_AND_REVERSE = 2,
    XCB_GX_COPY = 3,
    XCB_GX_AND_INVERTED = 4,
    XCB_GX_NOOP = 5,
    XCB_GX_XOR = 6,
    XCB_GX_OR = 7,
    XCB_GX_NOR = 8,
    XCB_GX_EQUIV = 9,
    XCB_GX_INVERT = 10,
    XCB_GX_OR_REVERSE = 11,
    XCB_GX_COPY_INVERTED = 12,
    XCB_GX_OR_INVERTED = 13,
    XCB_GX_NAND = 14,
    XCB_GX_SET = 15
} xcb_gx_t;
typedef enum xcb_line_style_t {
    XCB_LINE_STYLE_SOLID = 0,
    XCB_LINE_STYLE_ON_OFF_DASH = 1,
    XCB_LINE_STYLE_DOUBLE_DASH = 2
} xcb_line_style_t;
typedef enum xcb_cap_style_t {
    XCB_CAP_STYLE_NOT_LAST = 0,
    XCB_CAP_STYLE_BUTT = 1,
    XCB_CAP_STYLE_ROUND = 2,
    XCB_CAP_STYLE_PROJECTING = 3
} xcb_cap_style_t;
typedef enum xcb_join_style_t {
    XCB_JOIN_STYLE_MITER = 0,
    XCB_JOIN_STYLE_ROUND = 1,
    XCB_JOIN_STYLE_BEVEL = 2
} xcb_join_style_t;

```

```

typedef enum xcb_fill_style_t {
    XCB_FILL_STYLE_SOLID = 0,
    XCB_FILL_STYLE_TILED = 1,
    XCB_FILL_STYLE_STIPPLED = 2,
    XCB_FILL_STYLE_OPAQUE_STIPPLED = 3
} xcb_fill_style_t;
typedef enum xcb_fill_rule_t {
    XCB_FILL_RULE_EVEN_ODD = 0,
    XCB_FILL_RULE_WINDING = 1
} xcb_fill_rule_t;
typedef enum xcb_subwindow_mode_t {
    XCB_SUBWINDOW_MODE_CLIP_BY_CHILDREN = 0,
    XCB_SUBWINDOW_MODE_INCLUDE_INFERIORS = 1
} xcb_subwindow_mode_t;
typedef enum xcb_arc_mode_t {
    XCB_ARC_MODE_CHORD = 0,
    XCB_ARC_MODE_PIE_SLICE = 1
} xcb_arc_mode_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_gcontext_t cid;
    xcb_drawable_t drawable;
    uint32_t value_mask;
} xcb_create_gc_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_gcontext_t gc;
    uint32_t value_mask;
} xcb_change_gc_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_gcontext_t src_gc;
    xcb_gcontext_t dst_gc;
    uint32_t value_mask;
} xcb_copy_gc_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_gcontext_t gc;
    uint16_t dash_offset;
    uint16_t dashes_len;
} xcb_set_dashes_request_t;
typedef enum xcb_clip_ordering_t {
    XCB_CLIP_ORDERING_UNSORTED = 0,
    XCB_CLIP_ORDERING_Y_SORTED = 1,
    XCB_CLIP_ORDERING_YX_SORTED = 2,
    XCB_CLIP_ORDERING_YX_BANDED = 3
} xcb_clip_ordering_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t ordering;
    uint16_t length;
    xcb_gcontext_t gc;
    int16_t clip_x_origin;
    int16_t clip_y_origin;
} xcb_set_clip_rectangles_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;

```

```

        uint16_t length;
        xcb_gcontext_t gc;
    } xcb_free_gc_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t exposures;
    uint16_t length;
    xcb_window_t window;
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
} xcb_clear_area_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t src_drawable;
    xcb_drawable_t dst_drawable;
    xcb_gcontext_t gc;
    int16_t src_x;
    int16_t src_y;
    int16_t dst_x;
    int16_t dst_y;
    uint16_t width;
    uint16_t height;
} xcb_copy_area_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t src_drawable;
    xcb_drawable_t dst_drawable;
    xcb_gcontext_t gc;
    int16_t src_x;
    int16_t src_y;
    int16_t dst_x;
    int16_t dst_y;
    uint16_t width;
    uint16_t height;
    uint32_t bit_plane;
} xcb_copy_plane_request_t;
typedef enum xcb_coord_mode_t {
    XCB_COORD_MODE_ORIGIN = 0,
    XCB_COORD_MODE_PREVIOUS = 1
} xcb_coord_mode_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t coordinate_mode;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_point_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t coordinate_mode;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_line_request_t;
typedef struct {
    int16_t x1;
    int16_t y1;
    int16_t x2;
    int16_t y2;
} xcb_segment_t;

```



```

typedef struct {
    xcb_segment_t *data;
    int rem;
    int index;
} xcb_segment_iterator_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_segment_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_rectangle_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_arc_request_t;
typedef enum xcb_poly_shape_t {
    XCB_POLY_SHAPE_COMPLEX = 0,
    XCB_POLY_SHAPE_NONCONVEX = 1,
    XCB_POLY_SHAPE_CONVEX = 2
} xcb_poly_shape_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
    uint8_t shape;
    uint8_t coordinate_mode;
    uint8_t pad1[2];
} xcb_fill_poly_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_fill_rectangle_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
} xcb_poly_fill_arc_request_t;
typedef enum xcb_image_format_t {
    XCB_IMAGE_FORMAT_XY_BITMAP = 0,
    XCB_IMAGE_FORMAT_XY_PIXMAP = 1,
    XCB_IMAGE_FORMAT_Z_PIXMAP = 2
} xcb_image_format_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t format;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;

```

```

        uint16_t width;
        uint16_t height;
        int16_t dst_x;
        int16_t dst_y;
        uint8_t left_pad;
        uint8_t depth;
        uint8_t pad0[2];
    } xcb_put_image_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_image_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t format;
    uint16_t length;
    xcb_drawable_t drawable;
    int16_t x;
    int16_t y;
    uint16_t width;
    uint16_t height;
    uint32_t plane_mask;
} xcb_get_image_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t depth;
    uint16_t sequence;
    uint32_t length;
    xcb_visualid_t visual;
    uint8_t pad0[20];
} xcb_get_image_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
    int16_t x;
    int16_t y;
} xcb_poly_text_8_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
    int16_t x;
    int16_t y;
} xcb_poly_text_16_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t string_len;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
    int16_t x;
    int16_t y;
} xcb_image_text_8_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t string_len;
    uint16_t length;
    xcb_drawable_t drawable;
    xcb_gcontext_t gc;
    int16_t x;
    int16_t y;
} xcb_image_text_16_request_t;

```

```

typedef enum xcb_colormap_alloc_t {
    XCB_COLORMAP_ALLOC_NONE = 0,
    XCB_COLORMAP_ALLOC_ALL = 1
} xcb_colormap_alloc_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t alloc;
    uint16_t length;
    xcb_colormap_t mid;
    xcb_window_t window;
    xcb_visualid_t visual;
} xcb_create_colormap_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
} xcb_free_colormap_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t mid;
    xcb_colormap_t src_cmap;
} xcb_copy_colormap_and_free_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
} xcb_install_colormap_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
} xcb_uninstall_colormap_request_t;
typedef struct {
    unsigned int sequence;
} xcb_list_installed_colormaps_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
} xcb_list_installed_colormaps_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t cmaps_len;
    uint8_t pad1[22];
} xcb_list_installed_colormaps_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_alloc_color_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
    uint16_t red;
    uint16_t green;
    uint16_t blue;
    uint8_t pad1[2];

```

```

} xcb_alloc_color_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t red;
    uint16_t green;
    uint16_t blue;
    uint8_t pad1[2];
    uint32_t pixel;
} xcb_alloc_color_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_alloc_named_color_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
    uint16_t name_len;
    uint8_t pad1[2];
} xcb_alloc_named_color_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint32_t pixel;
    uint16_t exact_red;
    uint16_t exact_green;
    uint16_t exact_blue;
    uint16_t visual_red;
    uint16_t visual_green;
    uint16_t visual_blue;
} xcb_alloc_named_color_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_alloc_color_cells_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t contiguous;
    uint16_t length;
    xcb_colormap_t cmap;
    uint16_t colors;
    uint16_t planes;
} xcb_alloc_color_cells_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t pixels_len;
    uint16_t masks_len;
    uint8_t pad1[20];
} xcb_alloc_color_cells_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_alloc_color_planes_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t contiguous;
    uint16_t length;
    xcb_colormap_t cmap;
    uint16_t colors;
    uint16_t reds;

```

```

        uint16_t greens;
        uint16_t blues;
    } xcb_alloc_color_planes_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t pixels_len;
    uint8_t pad1[2];
    uint32_t red_mask;
    uint32_t green_mask;
    uint32_t blue_mask;
    uint8_t pad2[8];
} xcb_alloc_color_planes_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
    uint32_t plane_mask;
} xcb_free_colors_request_t;
typedef enum xcb_color_flag_t {
    XCB_COLOR_FLAG_RED = 1,
    XCB_COLOR_FLAG_GREEN = 2,
    XCB_COLOR_FLAG_BLUE = 4
} xcb_color_flag_t;
typedef struct {
    uint32_t pixel;
    uint16_t red;
    uint16_t green;
    uint16_t blue;
    uint8_t flags;
    uint8_t pad0;
} xcb_coloritem_t;
typedef struct {
    xcb_coloritem_t *data;
    int rem;
    int index;
} xcb_coloritem_iterator_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
} xcb_store_colors_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t flags;
    uint16_t length;
    xcb_colormap_t cmap;
    uint32_t pixel;
    uint16_t name_len;
    uint8_t pad0[2];
} xcb_store_named_color_request_t;
typedef struct {
    uint16_t red;
    uint16_t green;
    uint16_t blue;
    uint8_t pad0[2];
} xcb_rgb_t;
typedef struct {
    xcb_rgb_t *data;
    int rem;
    int index;
} xcb_rgb_iterator_t;

```

```

typedef struct {
    unsigned int sequence;
} xcb_query_colors_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
} xcb_query_colors_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t colors_len;
    uint8_t pad1[22];
} xcb_query_colors_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_lookup_color_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_colormap_t cmap;
    uint16_t name_len;
    uint8_t pad1[2];
} xcb_lookup_color_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t exact_red;
    uint16_t exact_green;
    uint16_t exact_blue;
    uint16_t visual_red;
    uint16_t visual_green;
    uint16_t visual_blue;
} xcb_lookup_color_reply_t;
typedef enum xcb_pixmap_enum_t {
    XCB_PIXMAP_NONE = 0
} xcb_pixmap_enum_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_cursor_t cid;
    xcb_pixmap_t source;
    xcb_pixmap_t mask;
    uint16_t fore_red;
    uint16_t fore_green;
    uint16_t fore_blue;
    uint16_t back_red;
    uint16_t back_green;
    uint16_t back_blue;
    uint16_t x;
    uint16_t y;
} xcb_create_cursor_request_t;
typedef enum xcb_font_enum_t {
    XCB_FONT_NONE = 0
} xcb_font_enum_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;

```

```

xcb_cursor_t cid;
xcb_font_t source_font;
xcb_font_t mask_font;
uint16_t source_char;
uint16_t mask_char;
uint16_t fore_red;
uint16_t fore_green;
uint16_t fore_blue;
uint16_t back_red;
uint16_t back_green;
uint16_t back_blue;
} xcb_create_glyph_cursor_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_cursor_t cursor;
} xcb_free_cursor_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_cursor_t cursor;
    uint16_t fore_red;
    uint16_t fore_green;
    uint16_t fore_blue;
    uint16_t back_red;
    uint16_t back_green;
    uint16_t back_blue;
} xcb_recolor_cursor_request_t;
typedef enum xcb_query_shape_of_t {
    XCB_QUERY_SHAPE_OF_LARGEST_CURSOR = 0,
    XCB_QUERY_SHAPE_OF_FASTEST_TILE = 1,
    XCB_QUERY_SHAPE_OF_FASTEST_STIPPLE = 2
} xcb_query_shape_of_t;
typedef struct {
    unsigned int sequence;
} xcb_query_best_size_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t _class;
    uint16_t length;
    xcb_drawable_t drawable;
    uint16_t width;
    uint16_t height;
} xcb_query_best_size_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t width;
    uint16_t height;
} xcb_query_best_size_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_query_extension_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    uint16_t name_len;
    uint8_t pad1[2];
} xcb_query_extension_request_t;
typedef struct {
    uint8_t response_type;

```

```

        uint8_t pad0;
        uint16_t sequence;
        uint32_t length;
        uint8_t present;
        uint8_t major_opcode;
        uint8_t first_event;
        uint8_t first_error;
    } xcb_query_extension_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_list_extensions_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_list_extensions_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t names_len;
    uint16_t sequence;
    uint32_t length;
    uint8_t pad0[24];
} xcb_list_extensions_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t keycode_count;
    uint16_t length;
    xcb_keycode_t first_keycode;
    uint8_t keysyms_per_keycode;
} xcb_change_keyboard_mapping_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_keyboard_mapping_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_keycode_t first_keycode;
    uint8_t count;
} xcb_get_keyboard_mapping_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t keysyms_per_keycode;
    uint16_t sequence;
    uint32_t length;
    uint8_t pad0[24];
} xcb_get_keyboard_mapping_reply_t;
typedef enum xcb_kb_t {
    XCB_KB_KEY_CLICK_PERCENT = 1,
    XCB_KB_BELL_PERCENT = 2,
    XCB_KB_BELL_PITCH = 4,
    XCB_KB_BELL_DURATION = 8,
    XCB_KB_LED = 16,
    XCB_KB_LED_MODE = 32,
    XCB_KB_KEY = 64,
    XCB_KB_AUTO_REPEAT_MODE = 128
} xcb_kb_t;
typedef enum xcb_led_mode_t {
    XCB_LED_MODE_OFF = 0,
    XCB_LED_MODE_ON = 1
} xcb_led_mode_t;
typedef enum xcb_auto_repeat_mode_t {
    XCB_AUTO_REPEAT_MODE_OFF = 0,
    XCB_AUTO_REPEAT_MODE_ON = 1,
    XCB_AUTO_REPEAT_MODE_DEFAULT = 2
} xcb_auto_repeat_mode_t;

```



```

typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    uint32_t value_mask;
} xcb_change_keyboard_control_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_keyboard_control_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_keyboard_control_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t global_auto_repeat;
    uint16_t sequence;
    uint32_t length;
    uint32_t led_mask;
    uint8_t key_click_percent;
    uint8_t bell_percent;
    uint16_t bell_pitch;
    uint16_t bell_duration;
    uint8_t pad0[2];
    uint8_t auto_repeats[32];
} xcb_get_keyboard_control_reply_t;
typedef struct {
    uint8_t major_opcode;
    int8_t percent;
    uint16_t length;
} xcb_bell_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    int16_t acceleration_numerator;
    int16_t acceleration_denominator;
    int16_t threshold;
    uint8_t do_acceleration;
    uint8_t do_threshold;
} xcb_change_pointer_control_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_pointer_control_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_pointer_control_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t acceleration_numerator;
    uint16_t acceleration_denominator;
    uint16_t threshold;
    uint8_t pad1[18];
} xcb_get_pointer_control_reply_t;
typedef enum xcb_blanking_t {
    XCB_BLANKING_NOT_PREFERRED = 0,
    XCB_BLANKING_PREFERRED = 1,
    XCB_BLANKING_DEFAULT = 2
} xcb_blanking_t;
typedef enum xcb_exposures_t {

```

```

        XCB_EXPOSURES_NOT_ALLOWED = 0,
        XCB_EXPOSURES_ALLOWED = 1,
        XCB_EXPOSURES_DEFAULT = 2
    } xcb_exposures_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    int16_t timeout;
    int16_t interval;
    uint8_t prefer_blanking;
    uint8_t allow_exposures;
} xcb_set_screen_saver_request_t;
typedef struct {
    unsigned int sequence;
} xcb_get_screen_saver_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_screen_saver_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t pad0;
    uint16_t sequence;
    uint32_t length;
    uint16_t timeout;
    uint16_t interval;
    uint8_t prefer_blanking;
    uint8_t allow_exposures;
    uint8_t pad1[18];
} xcb_get_screen_saver_reply_t;
typedef enum xcb_host_mode_t {
    XCB_HOST_MODE_INSERT = 0,
    XCB_HOST_MODE_DELETE = 1
} xcb_host_mode_t;
typedef enum xcb_family_t {
    XCB_FAMILY_INET = 0,
    XCB_FAMILY_DECNET = 1,
    XCB_FAMILY_CHAOS = 2,
    XCB_FAMILY_SERVER_INTERPRETED = 5,
    XCB_FAMILY_INET_6 = 6
} xcb_family_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
    uint8_t family;
    uint8_t pad0;
    uint16_t address_len;
} xcb_change_hosts_request_t;
typedef struct {
    uint8_t family;
    uint8_t pad0;
    uint16_t address_len;
} xcb_host_t;
typedef struct {
    xcb_host_t *data;
    int rem;
    int index;
} xcb_host_iterator_t;
typedef struct {
    unsigned int sequence;
} xcb_list_hosts_cookie_t;
typedef struct {
    uint8_t major_opcode;

```

```

    uint8_t pad0;
    uint16_t length;
} xcb_list_hosts_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t mode;
    uint16_t sequence;
    uint32_t length;
    uint16_t hosts_len;
    uint8_t pad0[22];
} xcb_list_hosts_reply_t;
typedef enum xcb_access_control_t {
    XCB_ACCESS_CONTROL_DISABLE = 0,
    XCB_ACCESS_CONTROL_ENABLE = 1
} xcb_access_control_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
} xcb_set_access_control_request_t;
typedef enum xcb_close_down_t {
    XCB_CLOSE_DOWN_DESTROY_ALL = 0,
    XCB_CLOSE_DOWN_RETAIN_PERMANENT = 1,
    XCB_CLOSE_DOWN_RETAIN_TEMPORARY = 2
} xcb_close_down_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
} xcb_set_close_down_mode_request_t;
typedef enum xcb_kill_t {
    XCB_KILL_ALL_TEMPORARY = 0
} xcb_kill_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    uint32_t resource;
} xcb_kill_client_request_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
    xcb_window_t window;
    uint16_t atoms_len;
    int16_t delta;
} xcb_rotate_properties_request_t;
typedef enum xcb_screen_saver_t {
    XCB_SCREEN_SAVER_RESET = 0,
    XCB_SCREEN_SAVER_ACTIVE = 1
} xcb_screen_saver_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t mode;
    uint16_t length;
} xcb_force_screen_saver_request_t;
typedef enum xcb_mapping_status_t {
    XCB_MAPPING_STATUS_SUCCESS = 0,
    XCB_MAPPING_STATUS_BUSY = 1,
    XCB_MAPPING_STATUS_FAILURE = 2
} xcb_mapping_status_t;
typedef struct {
    unsigned int sequence;
} xcb_set_pointer_mapping_cookie_t;
typedef struct {
    uint8_t major_opcode;

```

```

        uint8_t map_len;
        uint16_t length;
    } xcb_set_pointer_mapping_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t status;
    uint16_t sequence;
    uint32_t length;
} xcb_set_pointer_mapping_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_get_pointer_mapping_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_pointer_mapping_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t map_len;
    uint16_t sequence;
    uint32_t length;
    uint8_t pad0[24];
} xcb_get_pointer_mapping_reply_t;
typedef enum xcb_map_index_t {
    XCB_MAP_INDEX_SHIFT = 0,
    XCB_MAP_INDEX_LOCK = 1,
    XCB_MAP_INDEX_CONTROL = 2,
    XCB_MAP_INDEX_1 = 3,
    XCB_MAP_INDEX_2 = 4,
    XCB_MAP_INDEX_3 = 5,
    XCB_MAP_INDEX_4 = 6,
    XCB_MAP_INDEX_5 = 7
} xcb_map_index_t;
typedef struct {
    unsigned int sequence;
} xcb_set_modifier_mapping_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t keycodes_per_modifier;
    uint16_t length;
} xcb_set_modifier_mapping_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t status;
    uint16_t sequence;
    uint32_t length;
} xcb_set_modifier_mapping_reply_t;
typedef struct {
    unsigned int sequence;
} xcb_get_modifier_mapping_cookie_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;
    uint16_t length;
} xcb_get_modifier_mapping_request_t;
typedef struct {
    uint8_t response_type;
    uint8_t keycodes_per_modifier;
    uint16_t sequence;
    uint32_t length;
    uint8_t pad0[24];
} xcb_get_modifier_mapping_reply_t;
typedef struct {
    uint8_t major_opcode;
    uint8_t pad0;

```

```

    uint16_t length;
} xcb_no_operation_request_t;
extern xcb_alloc_color_cookie_t xcb_alloc_color(xcb_connection_t *
c,
                                xcb_colormap_t cmap,
                                uint16_t red,
                                uint16_t green,
                                uint16_t blue);

extern xcb_alloc_color_cells_cookie_t
xcb_alloc_color_cells(xcb_connection_t * c, uint8_t contiguous,
                      xcb_colormap_t cmap, uint16_t colors,
                      uint16_t planes);
extern uint32_t *xcb_alloc_color_cells_masks(const
xcb_alloc_color_cells_reply_t
                                * R);
extern
                                xcb_generic_iterator_t
xcb_alloc_color_cells_masks_end(const
xcb_alloc_color_cells_reply_t
                                * R);
extern int xcb_alloc_color_cells_masks_length(const
xcb_alloc_color_cells_reply_t
                                * R);
extern uint32_t *xcb_alloc_color_cells_pixels(const
xcb_alloc_color_cells_reply_t
                                * R);
extern
                                xcb_generic_iterator_t
xcb_alloc_color_cells_pixels_end(const
xcb_alloc_color_cells_reply_t
                                * R);
extern int xcb_alloc_color_cells_pixels_length(const
xcb_alloc_color_cells_reply_t
                                * R);
extern xcb_alloc_color_cells_reply_t
    *xcb_alloc_color_cells_reply(xcb_connection_t * c,
                                xcb_alloc_color_cells_cookie_t cookie,
                                xcb_generic_error_t **e);
extern xcb_alloc_color_cells_cookie_t
xcb_alloc_color_cells_unchecked(xcb_connection_t * c, uint8_t
contiguous,
                                xcb_colormap_t cmap, uint16_t colors,
                                uint16_t planes);
extern xcb_alloc_color_planes_cookie_t
xcb_alloc_color_planes(xcb_connection_t * c, uint8_t contiguous,
                      xcb_colormap_t cmap, uint16_t colors, uint16_t
reds,
                      uint16_t greens, uint16_t blues);
extern uint32_t *xcb_alloc_color_planes_pixels(const
xcb_alloc_color_planes_reply_t
                                * R);
extern
                                xcb_generic_iterator_t
xcb_alloc_color_planes_pixels_end(const
xcb_alloc_color_planes_reply_t
                                * R);
extern int xcb_alloc_color_planes_pixels_length(const
xcb_alloc_color_planes_reply_t
                                * R);
extern xcb_alloc_color_planes_reply_t
                                * R);
extern xcb_alloc_color_planes_reply_t

```

```

        *xcb_alloc_color_planes_reply(xcb_connection_t * c,
                                     xcb_alloc_color_planes_cookie_t
cookie,
                                     xcb_generic_error_t * *e);
extern xcb_alloc_color_planes_cookie_t
xcb_alloc_color_planes_unchecked(xcb_connection_t * c, uint8_t
contiguous,
                                xcb_colormap_t cmap, uint16_t colors,
                                uint16_t reds, uint16_t greens,
                                uint16_t blues);
extern
        xcb_alloc_color_reply_t
*xcb_alloc_color_reply(xcb_connection_t * c,

xcb_alloc_color_cookie_t
                                cookie,
                                xcb_generic_error_t *
*e);
extern
        xcb_alloc_color_cookie_t
xcb_alloc_color_unchecked(xcb_connection_t
                                * c,
                                xcb_colormap_t
cmap,
                                uint16_t red,
                                uint16_t green,
                                uint16_t blue);
extern xcb_alloc_named_color_cookie_t
xcb_alloc_named_color(xcb_connection_t * c, xcb_colormap_t cmap,
uint16_t name_len, const char *name);
extern xcb_alloc_named_color_reply_t
        *xcb_alloc_named_color_reply(xcb_connection_t * c,
        xcb_alloc_named_color_cookie_t cookie,
        xcb_generic_error_t * *e);
extern xcb_alloc_named_color_cookie_t
xcb_alloc_named_color_unchecked(xcb_connection_t * c,
xcb_colormap_t cmap,
                                uint16_t name_len, const char *name);
extern xcb_void_cookie_t xcb_allow_events(xcb_connection_t * c,
uint8_t mode,
        xcb_timestamp_t time);
extern xcb_void_cookie_t xcb_allow_events_checked(xcb_connection_t
* c,
        uint8_t mode,
        xcb_timestamp_t time);
extern xcb_generic_iterator_t xcb_arc_end(xcb_arc_iterator_t i);
extern void xcb_arc_next(xcb_arc_iterator_t * i);
extern xcb_generic_iterator_t xcb_atom_end(xcb_atom_iterator_t i);
extern void xcb_atom_next(xcb_atom_iterator_t * i);
extern xcb_void_cookie_t xcb_bell(xcb_connection_t * c, int8_t
percent);
extern xcb_void_cookie_t xcb_bell_checked(xcb_connection_t * c,
int8_t percent);
extern xcb_generic_iterator_t xcb_button_end(xcb_button_iterator_t
i);
extern void xcb_button_next(xcb_button_iterator_t * i);
extern
        xcb_void_cookie_t
xcb_change_active_pointer_grab(xcb_connection_t *
        c,
        xcb_cursor_t
cursor,
        xcb_timestamp_t
time,
        uint16_t
event_mask);
extern xcb_void_cookie_t
xcb_change_active_pointer_grab_checked(xcb_connection_t * c,
        xcb_cursor_t cursor,

```

```

        xcb_timestamp_t time,
        uint16_t event_mask);
extern xcb_void_cookie_t xcb_change_gc(xcb_connection_t * c,
        xcb_gcontext_t gc,
        uint32_t value_mask,
        const uint32_t * value_list);
extern xcb_void_cookie_t xcb_change_gc_checked(xcb_connection_t *
c,
        xcb_gcontext_t gc,
        uint32_t value_mask,
        const uint32_t *
        value_list);
extern xcb_void_cookie_t xcb_change_hosts(xcb_connection_t * c,
        uint8_t mode, uint8_t family,
        uint16_t address_len,
        const char *address);
extern xcb_void_cookie_t xcb_change_hosts_checked(xcb_connection_t
* c,
        uint8_t mode,
        uint8_t family,
        uint16_t address_len,
        const char *address);
extern
xcb_change_keyboard_control(xcb_connection_t * c,
        xcb_void_cookie_t
        uint32_t value_mask,
        const uint32_t *
        value_list);
extern xcb_void_cookie_t
xcb_change_keyboard_control_checked(xcb_connection_t * c,
        uint32_t value_mask,
        const uint32_t * value_list);
extern
xcb_change_keyboard_mapping(xcb_connection_t * c,
        xcb_void_cookie_t
        uint8_t keycode_count,
        xcb_keycode_t
        first_keycode,
        uint8_t
        keysyms_per_keycode,
        const xcb_keysym_t *
        keysyms);
extern xcb_void_cookie_t
xcb_change_keyboard_mapping_checked(xcb_connection_t * c,
        uint8_t keycode_count,
        xcb_keycode_t first_keycode,
        uint8_t keysyms_per_keycode,
        const xcb_keysym_t * keysyms);
extern
xcb_change_pointer_control(xcb_connection_t * c,
        xcb_void_cookie_t
        int16_t
        acceleration_numerator,
        int16_t
        acceleration_denominator,
        int16_t threshold,
        uint8_t
        do_acceleration,
        uint8_t do_threshold);
extern xcb_void_cookie_t
xcb_change_pointer_control_checked(xcb_connection_t * c,
        int16_t acceleration_numerator,
        int16_t acceleration_denominator,
        int16_t threshold,
        uint8_t do_acceleration,
        uint8_t do_threshold);
extern xcb_void_cookie_t xcb_change_property(xcb_connection_t * c,
        uint8_t mode,

```

```

xcb_window_t window,
xcb_atom_t property,
xcb_atom_t type,
uint8_t format,
uint32_t data_len,
const void *data);
extern xcb_void_cookie_t
xcb_change_property_checked(xcb_connection_t * c,
uint8_t mode,
xcb_window_t window,
xcb_atom_t property,
xcb_atom_t type,
uint8_t format,
uint32_t data_len,
const void *data);
extern xcb_void_cookie_t xcb_change_save_set(xcb_connection_t * c,
uint8_t mode,
xcb_window_t window);
extern xcb_void_cookie_t
xcb_change_save_set_checked(xcb_connection_t * c,
uint8_t mode,
xcb_window_t window);
extern xcb_void_cookie_t
xcb_change_window_attributes(xcb_connection_t * c,
xcb_window_t window,
uint32_t value_mask,
const uint32_t *
value_list);
extern xcb_void_cookie_t
xcb_change_window_attributes_checked(xcb_connection_t * c,
xcb_window_t window,
uint32_t value_mask,
const uint32_t * value_list);
extern xcb_generic_iterator_t xcb_char2b_end(xcb_char2b_iterator_t
i);
extern void xcb_char2b_next(xcb_char2b_iterator_t * i);
extern xcb_generic_iterator_t
xcb_charinfo_end(xcb_charinfo_iterator_t i);
extern void xcb_charinfo_next(xcb_charinfo_iterator_t * i);
extern xcb_void_cookie_t xcb_circulate_window(xcb_connection_t * c,
uint8_t direction,
xcb_window_t window);
extern xcb_void_cookie_t
xcb_circulate_window_checked(xcb_connection_t * c,
uint8_t direction,
xcb_window_t window);
extern xcb_void_cookie_t xcb_clear_area(xcb_connection_t * c,
uint8_t exposures,
xcb_window_t window, int16_t x,
int16_t y, uint16_t width,
uint16_t height);
extern xcb_void_cookie_t xcb_clear_area_checked(xcb_connection_t *
c,
uint8_t exposures,
xcb_window_t window,
int16_t x, int16_t y,
uint16_t width,
uint16_t height);
extern xcb_generic_iterator_t
xcb_client_message_data_end(xcb_client_message_data_iterator_t i);
extern void
xcb_client_message_data_next(xcb_client_message_data_iterator_t
* i);
extern xcb_void_cookie_t xcb_close_font(xcb_connection_t * c,
xcb_font_t font);

```



```

extern xcb_void_cookie_t xcb_close_font_checked(xcb_connection_t *
c,
xcb_font_t font);
extern
xcb_generic_iterator_t
xcb_coloritem_end(xcb_coloritem_iterator_t
i);
extern void xcb_coloritem_next(xcb_coloritem_iterator_t * i);
extern
xcb_generic_iterator_t
xcb_colormap_end(xcb_colormap_iterator_t i);
extern void xcb_colormap_next(xcb_colormap_iterator_t * i);
extern xcb_void_cookie_t xcb_configure_window(xcb_connection_t * c,
xcb_window_t window,
uint16_t value_mask,
const uint32_t * value_list);
extern
xcb_void_cookie_t
xcb_configure_window_checked(xcb_connection_t * c,
xcb_window_t window,
uint16_t value_mask,
const uint32_t *
value_list);
extern xcb_void_cookie_t xcb_convert_selection(xcb_connection_t *
c,
xcb_window_t requestor,
xcb_atom_t selection,
xcb_atom_t target,
xcb_atom_t property,
xcb_timestamp_t time);
extern
xcb_void_cookie_t
xcb_convert_selection_checked(xcb_connection_t *
c,
xcb_window_t
requestor,
xcb_atom_t
selection,
xcb_atom_t target,
xcb_atom_t property,
xcb_timestamp_t
time);
extern xcb_void_cookie_t xcb_copy_area(xcb_connection_t * c,
xcb_drawable_t src_drawable,
xcb_drawable_t dst_drawable,
xcb_gcontext_t gc, int16_t src_x,
int16_t src_y, int16_t dst_x,
int16_t dst_y, uint16_t width,
uint16_t height);
extern xcb_void_cookie_t xcb_copy_area_checked(xcb_connection_t *
c,
xcb_drawable_t
src_drawable,
xcb_drawable_t
dst_drawable,
xcb_gcontext_t gc,
int16_t src_x,
int16_t src_y,
int16_t dst_x,
int16_t dst_y,
uint16_t width,
uint16_t height);
extern
xcb_void_cookie_t
xcb_copy_colormap_and_free(xcb_connection_t * c,
xcb_colormap_t mid,
xcb_colormap_t
src_cmap);
extern xcb_void_cookie_t
xcb_copy_colormap_and_free_checked(xcb_connection_t * c,
xcb_colormap_t mid,

```

```

        xcb_colormap_t src_cmap);
extern xcb_void_cookie_t xcb_copy_gc(xcb_connection_t * c,
        xcb_gcontext_t src_gc,
        xcb_gcontext_t dst_gc,
        uint32_t value_mask);
extern xcb_void_cookie_t xcb_copy_gc_checked(xcb_connection_t * c,
        xcb_gcontext_t src_gc,
        xcb_gcontext_t dst_gc,
        uint32_t value_mask);
extern xcb_void_cookie_t xcb_copy_plane(xcb_connection_t * c,
        xcb_drawable_t src_drawable,
        xcb_drawable_t dst_drawable,
        xcb_gcontext_t gc, int16_t src_x,
        int16_t src_y, int16_t dst_x,
        int16_t dst_y, uint16_t width,
        uint16_t height,
        uint32_t bit_plane);
extern xcb_void_cookie_t xcb_copy_plane_checked(xcb_connection_t *
c,
        xcb_drawable_t
        src_drawable,
        xcb_drawable_t
        dst_drawable,
        xcb_gcontext_t gc,
        int16_t src_x,
        int16_t src_y,
        int16_t dst_x,
        int16_t dst_y,
        uint16_t width,
        uint16_t height,
        uint32_t bit_plane);
extern xcb_void_cookie_t xcb_create_colormap(xcb_connection_t * c,
        uint8_t alloc,
        xcb_colormap_t mid,
        xcb_window_t window,
        xcb_visualid_t visual);
extern
        xcb_void_cookie_t
xcb_create_colormap_checked(xcb_connection_t * c,
        uint8_t alloc,
        xcb_colormap_t mid,
        xcb_window_t window,
        xcb_visualid_t
        visual);
extern xcb_void_cookie_t xcb_create_cursor(xcb_connection_t * c,
        xcb_cursor_t cid,
        xcb_pixmap_t source,
        xcb_pixmap_t mask,
        uint16_t fore_red,
        uint16_t fore_green,
        uint16_t fore_blue,
        uint16_t back_red,
        uint16_t back_green,
        uint16_t back_blue, uint16_t x,
        uint16_t y);
extern
        xcb_void_cookie_t
xcb_create_cursor_checked(xcb_connection_t * c,
        xcb_cursor_t cid,
        xcb_pixmap_t source,
        xcb_pixmap_t mask,
        uint16_t fore_red,
        uint16_t fore_green,
        uint16_t fore_blue,
        uint16_t back_red,
        uint16_t back_green,
        uint16_t back_blue,
        uint16_t x, uint16_t y);

```

```

extern xcb_void_cookie_t xcb_create_gc(xcb_connection_t * c,
                                       xcb_gcontext_t cid,
                                       xcb_drawable_t drawable,
                                       uint32_t value_mask,
                                       const uint32_t * value_list);
extern xcb_void_cookie_t xcb_create_gc_checked(xcb_connection_t *
c,
                                             xcb_gcontext_t cid,
                                             xcb_drawable_t drawable,
                                             uint32_t value_mask,
                                             const uint32_t *
                                             value_list);
extern xcb_void_cookie_t xcb_create_glyph_cursor(xcb_connection_t
* c,
                                             xcb_cursor_t cid,
                                             xcb_font_t source_font,
                                             xcb_font_t mask_font,
                                             uint16_t source_char,
                                             uint16_t mask_char,
                                             uint16_t fore_red,
                                             uint16_t fore_green,
                                             uint16_t fore_blue,
                                             uint16_t back_red,
                                             uint16_t back_green,
                                             uint16_t back_blue);
extern
xcb_create_glyph_cursor_checked(xcb_connection_t *
c,
                               xcb_cursor_t cid,
                               xcb_font_t
                               source_font,
                               xcb_font_t
                               mask_font,
                               uint16_t
                               source_char,
                               uint16_t
                               mask_char,
                               uint16_t fore_red,
                               uint16_t
                               fore_green,
                               uint16_t
                               fore_blue,
                               uint16_t back_red,
                               uint16_t
                               back_green,
                               uint16_t
                               back_blue);
extern xcb_void_cookie_t xcb_create_pixmap(xcb_connection_t * c,
uint8_t depth, xcb_pixmap_t
pid,
xcb_drawable_t drawable,
uint16_t width,
uint16_t height);
extern
xcb_create_pixmap_checked(xcb_connection_t * c,
                          uint8_t depth,
                          xcb_pixmap_t pid,
                          xcb_drawable_t drawable,
                          uint16_t width,
                          uint16_t height);
extern xcb_void_cookie_t xcb_create_window(xcb_connection_t * c,
uint8_t depth, xcb_window_t
wid,
xcb_window_t parent, int16_t x,
int16_t y, uint16_t width,
uint16_t height,

```

```

uint16_t border_width,
uint16_t _class,
xcb_visualid_t visual,
uint32_t value_mask,
const uint32_t * value_list);
extern
xcb_void_cookie_t xcb_create_window_checked(xcb_connection_t * c,
uint8_t depth,
xcb_window_t wid,
xcb_window_t parent,
int16_t x, int16_t y,
uint16_t width,
uint16_t height,
uint16_t border_width,
uint16_t _class,
xcb_visualid_t visual,
uint32_t value_mask,
const uint32_t *
value_list);
extern xcb_generic_iterator_t xcb_cursor_end(xcb_cursor_iterator_t
i);
extern void xcb_cursor_next(xcb_cursor_iterator_t * i);
extern xcb_void_cookie_t xcb_delete_property(xcb_connection_t * c,
xcb_window_t window,
xcb_atom_t property);
extern
xcb_void_cookie_t xcb_delete_property_checked(xcb_connection_t * c,
xcb_window_t window,
xcb_atom_t property);
extern xcb_generic_iterator_t xcb_depth_end(xcb_depth_iterator_t
i);
extern void xcb_depth_next(xcb_depth_iterator_t * i);
extern xcb_visualtype_t *xcb_depth_visuals(const xcb_depth_t * R);
extern xcb_visualtype_iterator_t xcb_depth_visuals_iterator(const
xcb_depth_t *
R);
extern int xcb_depth_visuals_length(const xcb_depth_t * R);
extern xcb_void_cookie_t xcb_destroy_subwindows(xcb_connection_t *
c,
xcb_window_t window);
extern
xcb_void_cookie_t xcb_destroy_subwindows_checked(xcb_connection_t *
c,
xcb_window_t
window);
extern xcb_void_cookie_t xcb_destroy_window(xcb_connection_t * c,
xcb_window_t window);
extern
xcb_void_cookie_t xcb_destroy_window_checked(xcb_connection_t * c,
xcb_window_t window);
extern
xcb_generic_iterator_t
xcb_drawable_end(xcb_drawable_iterator_t i);
extern void xcb_drawable_next(xcb_drawable_iterator_t * i);
extern xcb_void_cookie_t xcb_fill_poly(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc, uint8_t shape,
uint8_t coordinate_mode,
uint32_t points_len,
const xcb_point_t * points);
extern xcb_void_cookie_t xcb_fill_poly_checked(xcb_connection_t *
c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint8_t shape,
uint8_t coordinate_mode,
uint32_t points_len,

```

```

const xcb_point_t * points);
extern xcb_generic_iterator_t xcb_font_end(xcb_font_iterator_t i);
extern void xcb_font_next(xcb_font_iterator_t * i);
extern xcb_generic_iterator_t
xcb_fontable_end(xcb_fontable_iterator_t i);
extern void xcb_fontable_next(xcb_fontable_iterator_t * i);
extern xcb_generic_iterator_t
xcb_fontprop_end(xcb_fontprop_iterator_t i);
extern void xcb_fontprop_next(xcb_fontprop_iterator_t * i);
extern xcb_void_cookie_t xcb_force_screen_saver(xcb_connection_t *
c,
uint8_t mode);
extern xcb_void_cookie_t
xcb_force_screen_saver_checked(xcb_connection_t *
c, uint8_t mode);
extern xcb_generic_iterator_t xcb_format_end(xcb_format_iterator_t
i);
extern void xcb_format_next(xcb_format_iterator_t * i);
extern xcb_void_cookie_t xcb_free_colormap(xcb_connection_t * c,
xcb_colormap_t cmap);
extern xcb_void_cookie_t
xcb_free_colormap_checked(xcb_connection_t * c,
xcb_colormap_t cmap);
extern xcb_void_cookie_t xcb_free_colors(xcb_connection_t * c,
xcb_colormap_t cmap,
uint32_t plane_mask,
uint32_t pixels_len,
const uint32_t * pixels);
extern xcb_void_cookie_t xcb_free_colors_checked(xcb_connection_t
* c,
xcb_colormap_t cmap,
uint32_t plane_mask,
uint32_t pixels_len,
const uint32_t * pixels);
extern xcb_void_cookie_t xcb_free_cursor(xcb_connection_t * c,
xcb_cursor_t cursor);
extern xcb_void_cookie_t xcb_free_cursor_checked(xcb_connection_t
* c,
xcb_cursor_t cursor);
extern xcb_void_cookie_t xcb_free_gc(xcb_connection_t * c,
xcb_gcontext_t gc);
extern xcb_void_cookie_t xcb_free_gc_checked(xcb_connection_t * c,
xcb_gcontext_t gc);
extern xcb_void_cookie_t xcb_free_pixmap(xcb_connection_t * c,
xcb_pixmap_t pixmap);
extern xcb_void_cookie_t xcb_free_pixmap_checked(xcb_connection_t
* c,
xcb_pixmap_t pixmap);
extern xcb_generic_iterator_t
xcb_gcontext_end(xcb_gcontext_iterator_t i);
extern void xcb_gcontext_next(xcb_gcontext_iterator_t * i);
extern xcb_get_atom_name_cookie_t
xcb_get_atom_name(xcb_connection_t * c,
xcb_atom_t atom);
extern char *xcb_get_atom_name_name(const
xcb_get_atom_name_reply_t * R);
extern xcb_generic_iterator_t xcb_get_atom_name_name_end(const
xcb_get_atom_name_reply_t
* R);
extern int xcb_get_atom_name_name_length(const
xcb_get_atom_name_reply_t *
R);
extern xcb_get_atom_name_reply_t
*xcb_get_atom_name_reply(xcb_connection_t
* c,

```

```

xcb_get_atom_name_cookie_t
                                cookie,

xcb_generic_error_t
                                * *e);

extern xcb_get_atom_name_cookie_t
xcb_get_atom_name_unchecked(xcb_connection_t * c, xcb_atom_t atom);
extern                                xcb_get_font_path_cookie_t
xcb_get_font_path(xcb_connection_t * c);
extern xcb_str_iterator_t xcb_get_font_path_path_iterator(const

xcb_get_font_path_reply_t
                                * R);
extern          int          xcb_get_font_path_path_length(const
xcb_get_font_path_reply_t *
                                R);
extern                                xcb_get_font_path_reply_t
*xcb_get_font_path_reply(xcb_connection_t
                                * c,

xcb_get_font_path_cookie_t
                                cookie,

xcb_generic_error_t
                                * *e);

extern xcb_get_font_path_cookie_t
xcb_get_font_path_unchecked(xcb_connection_t * c);
extern xcb_get_geometry_cookie_t xcb_get_geometry(xcb_connection_t
* c,
                                xcb_drawable_t drawable);
extern                                xcb_get_geometry_reply_t
*xcb_get_geometry_reply(xcb_connection_t *
                                c,

xcb_get_geometry_cookie_t
                                cookie,
                                xcb_generic_error_t
                                * *e);

extern xcb_get_geometry_cookie_t
xcb_get_geometry_unchecked(xcb_connection_t * c, xcb_drawable_t
drawable);
extern xcb_get_image_cookie_t xcb_get_image(xcb_connection_t * c,
                                uint8_t format,
                                xcb_drawable_t drawable,
                                int16_t x, int16_t y,
                                uint16_t width,
                                uint16_t height,
                                uint32_t plane_mask);
extern uint8_t *xcb_get_image_data(const xcb_get_image_reply_t *
R);
extern xcb_generic_iterator_t xcb_get_image_data_end(const
                                xcb_get_image_reply_t
                                * R);
extern int xcb_get_image_data_length(const xcb_get_image_reply_t *
R);
extern xcb_get_image_reply_t *xcb_get_image_reply(xcb_connection_t
* c,
                                xcb_get_image_cookie_t
                                cookie,
                                xcb_generic_error_t *
                                *e);
extern                                xcb_get_image_cookie_t
xcb_get_image_unchecked(xcb_connection_t * c,
                                uint8_t format,
                                xcb_drawable_t

```

```

        drawable, int16_t x,
        int16_t y,
        uint16_t width,
        uint16_t height,
        uint32_t plane_mask);
extern xcb_get_input_focus_cookie_t
xcb_get_input_focus(xcb_connection_t *
                    c);
extern xcb_get_input_focus_reply_t
xcb_get_input_focus_reply(xcb_connection_t * c,
                          xcb_get_input_focus_cookie_t cookie,
                          xcb_generic_error_t * *e);
extern xcb_get_input_focus_cookie_t
xcb_get_input_focus_unchecked(xcb_connection_t * c);
extern xcb_get_keyboard_control_cookie_t
xcb_get_keyboard_control(xcb_connection_t * c);
extern xcb_get_keyboard_control_reply_t
xcb_get_keyboard_control_reply(xcb_connection_t * c,
                              xcb_get_keyboard_control_cookie_t
                              cookie, xcb_generic_error_t * *e);
extern xcb_get_keyboard_control_cookie_t
xcb_get_keyboard_control_unchecked(xcb_connection_t * c);
extern xcb_get_keyboard_mapping_cookie_t
xcb_get_keyboard_mapping(xcb_connection_t * c, xcb_keycode_t
first_keycode,
                        uint8_t count);
extern xcb_keysym_t *xcb_get_keyboard_mapping_keysyms(const
xcb_get_keyboard_mapping_reply_t
                                * R);
extern xcb_get_keyboard_mapping_keysyms_end(const
xcb_get_keyboard_mapping_reply_t
                                * R);
extern int xcb_get_keyboard_mapping_keysyms_length(const
xcb_get_keyboard_mapping_reply_t
                                * R);
extern xcb_get_keyboard_mapping_reply_t
xcb_get_keyboard_mapping_reply(xcb_connection_t * c,
                              xcb_get_keyboard_mapping_cookie_t
                              cookie, xcb_generic_error_t * *e);
extern xcb_get_keyboard_mapping_cookie_t
xcb_get_keyboard_mapping_unchecked(xcb_connection_t * c,
                                   xcb_keycode_t first_keycode,
                                   uint8_t count);
extern xcb_get_modifier_mapping_cookie_t
xcb_get_modifier_mapping(xcb_connection_t * c);
extern xcb_keycode_t *xcb_get_modifier_mapping_keycodes(const
xcb_get_modifier_mapping_reply_t
                                * R);
extern xcb_get_modifier_mapping_keycodes_end(const
xcb_get_modifier_mapping_reply_t
                                * R);
extern int xcb_get_modifier_mapping_keycodes_length(const
xcb_get_modifier_mapping_reply_t
                                * R);
extern xcb_get_modifier_mapping_reply_t
xcb_get_modifier_mapping_reply(xcb_connection_t * c,
                              xcb_get_modifier_mapping_cookie_t
                              cookie, xcb_generic_error_t * *e);

```

```

extern xcb_get_modifier_mapping_cookie_t
xcb_get_modifier_mapping_unchecked(xcb_connection_t * c);
extern xcb_get_motion_events_cookie_t
xcb_get_motion_events(xcb_connection_t * c, xcb_window_t window,
                      xcb_timestamp_t start, xcb_timestamp_t stop);
extern xcb_timecoord_t *xcb_get_motion_events_events(const

xcb_get_motion_events_reply_t
                                * R);
extern
                                xcb_timecoord_iterator_t
xcb_get_motion_events_events_iterator(const

xcb_get_motion_events_reply_t
                                * R);
extern int xcb_get_motion_events_events_length(const

xcb_get_motion_events_reply_t
                                * R);
extern xcb_get_motion_events_reply_t
    *xcb_get_motion_events_reply(xcb_connection_t * c,
                                xcb_get_motion_events_cookie_t cookie,
                                xcb_generic_error_t * *e);
extern xcb_get_motion_events_cookie_t
xcb_get_motion_events_unchecked(xcb_connection_t * c, xcb_window_t
window,
                                xcb_timestamp_t start,
                                xcb_timestamp_t stop);
extern xcb_get_pointer_control_cookie_t
xcb_get_pointer_control(xcb_connection_t * c);
extern xcb_get_pointer_control_reply_t
    *xcb_get_pointer_control_reply(xcb_connection_t * c,
                                xcb_get_pointer_control_cookie_t
cookie,
                                xcb_generic_error_t * *e);
extern xcb_get_pointer_control_cookie_t
xcb_get_pointer_control_unchecked(xcb_connection_t * c);
extern xcb_get_pointer_mapping_cookie_t
xcb_get_pointer_mapping(xcb_connection_t * c);
extern uint8_t *xcb_get_pointer_mapping_map(const

xcb_get_pointer_mapping_reply_t
                                * R);
extern
                                xcb_generic_iterator_t
xcb_get_pointer_mapping_map_end(const

xcb_get_pointer_mapping_reply_t
                                * R);
extern int xcb_get_pointer_mapping_map_length(const

xcb_get_pointer_mapping_reply_t
                                * R);
extern xcb_get_pointer_mapping_reply_t
    *xcb_get_pointer_mapping_reply(xcb_connection_t * c,
                                xcb_get_pointer_mapping_cookie_t
cookie,
                                xcb_generic_error_t * *e);
extern xcb_get_pointer_mapping_cookie_t
xcb_get_pointer_mapping_unchecked(xcb_connection_t * c);
extern xcb_get_property_cookie_t xcb_get_property(xcb_connection_t
* c,
                                uint8_t _delete,
                                xcb_window_t window,
                                xcb_atom_t property,
                                xcb_atom_t type,
                                uint32_t long_offset,
                                uint32_t long_length);

```



```

extern xcb_get_property_reply_t
*xcb_get_property_reply(xcb_connection_t *
                        c,

xcb_get_property_cookie_t

                        cookie,
                        xcb_generic_error_t
                        * *e);

extern xcb_get_property_cookie_t
xcb_get_property_unchecked(xcb_connection_t * c, uint8_t delete,
                           xcb_window_t window, xcb_atom_t property,
                           xcb_atom_t type, uint32_t long_offset,
                           uint32_t long_length);
extern void *xcb_get_property_value(const xcb_get_property_reply_t
* R);
extern xcb_generic_iterator_t xcb_get_property_value_end(const

xcb_get_property_reply_t

                        * R);
extern int xcb_get_property_value_length(const
xcb_get_property_reply_t *

                        R);
extern xcb_get_screen_saver_cookie_t
xcb_get_screen_saver(xcb_connection_t

                        * c);
extern xcb_get_screen_saver_reply_t
*xcb_get_screen_saver_reply(xcb_connection_t * c,
                            xcb_get_screen_saver_cookie_t cookie,
                            xcb_generic_error_t * *e);
extern xcb_get_screen_saver_cookie_t
xcb_get_screen_saver_unchecked(xcb_connection_t * c);
extern xcb_get_selection_owner_cookie_t
xcb_get_selection_owner(xcb_connection_t * c, xcb_atom_t
selection);
extern xcb_get_selection_owner_reply_t
*xcb_get_selection_owner_reply(xcb_connection_t * c,
                               xcb_get_selection_owner_cookie_t
cookie,
                               xcb_generic_error_t * *e);
extern xcb_get_selection_owner_cookie_t
xcb_get_selection_owner_unchecked(xcb_connection_t * c,
                                  xcb_atom_t selection);
extern xcb_get_window_attributes_cookie_t
xcb_get_window_attributes(xcb_connection_t * c, xcb_window_t
window);
extern xcb_get_window_attributes_reply_t
*xcb_get_window_attributes_reply(xcb_connection_t * c,
                                 xcb_get_window_attributes_cookie_t
                                 cookie, xcb_generic_error_t * *e);
extern xcb_get_window_attributes_cookie_t
xcb_get_window_attributes_unchecked(xcb_connection_t * c,
                                    xcb_window_t window);
extern xcb_void_cookie_t xcb_grab_button(xcb_connection_t * c,
                                         uint8_t owner_events,
                                         xcb_window_t grab_window,
                                         uint16_t event_mask,
                                         uint8_t pointer_mode,
                                         uint8_t keyboard_mode,
                                         xcb_window_t confine_to,
                                         xcb_cursor_t cursor,
                                         uint8_t button,
                                         uint16_t modifiers);
extern xcb_void_cookie_t xcb_grab_button_checked(xcb_connection_t
* c,
                                                  uint8_t owner_events,
                                                  xcb_window_t grab_window,

```

```

uint16_t event_mask,
uint8_t pointer_mode,
uint8_t keyboard_mode,
xcb_window_t confine_to,
xcb_cursor_t cursor,
uint8_t button,
uint16_t modifiers);
extern xcb_void_cookie_t xcb_grab_key(xcb_connection_t * c,
uint8_t owner_events,
xcb_window_t grab_window,
uint16_t modifiers,
xcb_keycode_t key,
uint8_t pointer_mode,
uint8_t keyboard_mode);
extern xcb_void_cookie_t xcb_grab_key_checked(xcb_connection_t * c,
uint8_t owner_events,
xcb_window_t grab_window,
uint16_t modifiers,
xcb_keycode_t key,
uint8_t pointer_mode,
uint8_t keyboard_mode);
extern
xcb_grab_keyboard(xcb_connection_t * c,
uint8_t owner_events,
xcb_window_t
grab_window,
xcb_timestamp_t time,
uint8_t pointer_mode,
uint8_t keyboard_mode);
extern
*xcb_grab_keyboard_reply(xcb_connection_t
* c,

xcb_grab_keyboard_cookie_t

cookie,

xcb_generic_error_t

* *e);
extern xcb_grab_keyboard_cookie_t
xcb_grab_keyboard_unchecked(xcb_connection_t * c, uint8_t
owner_events,

xcb_window_t grab_window, xcb_timestamp_t
time,

uint8_t pointer_mode, uint8_t
keyboard_mode);
extern xcb_grab_pointer_cookie_t xcb_grab_pointer(xcb_connection_t
* c,

uint8_t owner_events,
xcb_window_t grab_window,
uint16_t event_mask,
uint8_t pointer_mode,
uint8_t keyboard_mode,
xcb_window_t confine_to,
xcb_cursor_t cursor,
xcb_timestamp_t time);
extern
*xcb_grab_pointer_reply(xcb_connection_t *
c,

xcb_grab_pointer_cookie_t

cookie,
xcb_generic_error_t
* *e);
extern xcb_grab_pointer_cookie_t
xcb_grab_pointer_unchecked(xcb_connection_t * c, uint8_t
owner_events,

```

```

xcb_window_t      grab_window,      uint16_t
event_mask,
uint8_t pointer_mode, uint8_t keyboard_mode,
xcb_window_t confine_to, xcb_cursor_t cursor,
xcb_timestamp_t time);
extern xcb_void_cookie_t xcb_grab_server(xcb_connection_t * c);
extern xcb_void_cookie_t xcb_grab_server_checked(xcb_connection_t
* c);
extern uint8_t *xcb_host_address(const xcb_host_t * R);
extern xcb_generic_iterator_t xcb_host_address_end(const
xcb_host_t * R);
extern int xcb_host_address_length(const xcb_host_t * R);
extern xcb_generic_iterator_t xcb_host_end(xcb_host_iterator_t i);
extern void xcb_host_next(xcb_host_iterator_t * i);
extern xcb_void_cookie_t xcb_image_text_16(xcb_connection_t * c,
uint8_t string_len,
xcb_drawable_t drawable,
xcb_gcontext_t gc, int16_t x,
int16_t y,
const xcb_char2b_t * string);
extern xcb_void_cookie_t xcb_image_text_16_checked(xcb_connection_t * c,
uint8_t string_len,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
int16_t x, int16_t y,
const xcb_char2b_t *
string);
extern xcb_void_cookie_t xcb_image_text_8(xcb_connection_t * c,
uint8_t string_len,
xcb_drawable_t drawable,
xcb_gcontext_t gc, int16_t x,
int16_t y, const char *string);
extern xcb_void_cookie_t xcb_image_text_8_checked(xcb_connection_t
* c,
uint8_t string_len,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
int16_t x, int16_t y,
const char *string);
extern xcb_void_cookie_t xcb_install_colormap(xcb_connection_t * c,
xcb_colormap_t cmap);
extern xcb_void_cookie_t
xcb_install_colormap_checked(xcb_connection_t * c,
xcb_colormap_t cmap);
extern xcb_intern_atom_cookie_t xcb_intern_atom(xcb_connection_t *
c,
uint8_t only_if_exists,
uint16_t name_len,
const char *name);
extern xcb_intern_atom_reply_t
*xcb_intern_atom_reply(xcb_connection_t * c,
xcb_intern_atom_cookie_t
cookie,
xcb_generic_error_t *
*e);
extern xcb_intern_atom_unchecked(xcb_connection_t
* c,
uint8_t
only_if_exists,
uint16_t
name_len,
const char
*name);

```

```

extern xcb_generic_iterator_t
xcb_keycode_end(xcb_keycode_iterator_t i);
extern void xcb_keycode_next(xcb_keycode_iterator_t * i);
extern xcb_generic_iterator_t xcb_keysym_end(xcb_keysym_iterator_t
i);
extern void xcb_keysym_next(xcb_keysym_iterator_t * i);
extern xcb_void_cookie_t xcb_kill_client(xcb_connection_t * c,
uint32_t resource);
extern xcb_void_cookie_t xcb_kill_client_checked(xcb_connection_t
* c,
uint32_t resource);
extern xcb_list_extensions_cookie_t
xcb_list_extensions(xcb_connection_t *
c);
extern xcb_str_iterator_t xcb_list_extensions_names_iterator(const
xcb_list_extensions_reply_t
* R);
extern int xcb_list_extensions_names_length(const
xcb_list_extensions_reply_t *
R);
extern xcb_list_extensions_reply_t
*xcb_list_extensions_reply(xcb_connection_t * c,
xcb_list_extensions_cookie_t cookie,
xcb_generic_error_t * *e);
extern xcb_list_extensions_cookie_t
xcb_list_extensions_unchecked(xcb_connection_t * c);
extern xcb_list_fonts_cookie_t xcb_list_fonts(xcb_connection_t * c,
uint16_t max_names,
uint16_t pattern_len,
const char *pattern);
extern xcb_str_iterator_t xcb_list_fonts_names_iterator(const
xcb_list_fonts_reply_t
* R);
extern int xcb_list_fonts_names_length(const
xcb_list_fonts_reply_t * R);
extern xcb_list_fonts_reply_t
*xcb_list_fonts_reply(xcb_connection_t * c,
xcb_list_fonts_cookie_t
cookie,
xcb_generic_error_t *
*e);
extern xcb_list_fonts_cookie_t
xcb_list_fonts_unchecked(xcb_connection_t *
c,
uint16_t max_names,
uint16_t
pattern_len,
const char
*pattern);
extern xcb_list_fonts_with_info_cookie_t
xcb_list_fonts_with_info(xcb_connection_t * c, uint16_t max_names,
uint16_t pattern_len, const char *pattern);
extern char *xcb_list_fonts_with_info_name(const
xcb_list_fonts_with_info_reply_t
* R);
extern xcb_generic_iterator_t
xcb_list_fonts_with_info_name_end(const
xcb_list_fonts_with_info_reply_t
* R);
extern int xcb_list_fonts_with_info_name_length(const

```

```

xcb_list_fonts_with_info_reply_t
                                * R);
extern xcb_fontprop_t *xcb_list_fonts_with_info_properties(const
xcb_list_fonts_with_info_reply_t
                                * R);
extern xcb_fontprop_iterator_t
xcb_list_fonts_with_info_properties_iterator(const
xcb_list_fonts_with_info_reply_t
                                * R);
extern int xcb_list_fonts_with_info_properties_length(const
xcb_list_fonts_with_info_reply_t
                                * R);
extern xcb_list_fonts_with_info_reply_t
    *xcb_list_fonts_with_info_reply(xcb_connection_t * c,
                                    xcb_list_fonts_with_info_cookie_t
                                    cookie, xcb_generic_error_t **e);
extern xcb_list_fonts_with_info_cookie_t
xcb_list_fonts_with_info_unchecked(xcb_connection_t * c,
                                   uint16_t max_names,
                                   uint16_t pattern_len,
                                   const char *pattern);
extern xcb_list_hosts_cookie_t xcb_list_hosts(xcb_connection_t *
c);
extern xcb_host_iterator_t xcb_list_hosts_hosts_iterator(const
xcb_list_hosts_reply_t
                                * R);
extern int xcb_list_hosts_hosts_length(const
xcb_list_hosts_reply_t * R);
extern xcb_list_hosts_reply_t
    *xcb_list_hosts_reply(xcb_connection_t * c,
                           xcb_list_hosts_cookie_t
                           cookie,
                           xcb_generic_error_t *
                           *e);
extern xcb_list_hosts_cookie_t
xcb_list_hosts_unchecked(xcb_connection_t *
                           c);
extern xcb_list_installed_colormaps_cookie_t
xcb_list_installed_colormaps(xcb_connection_t * c, xcb_window_t
window);
extern xcb_colormap_t *xcb_list_installed_colormaps_cmaps(const
xcb_list_installed_colormaps_reply_t
                                * R);
extern xcb_generic_iterator_t
xcb_list_installed_colormaps_cmaps_end(const
xcb_list_installed_colormaps_reply_t
                                * R);
extern int xcb_list_installed_colormaps_cmaps_length(const
xcb_list_installed_colormaps_reply_t
                                * R);
extern xcb_list_installed_colormaps_reply_t
    *xcb_list_installed_colormaps_reply(xcb_connection_t * c,
                                         xcb_list_installed_colormaps_cookie_t
                                         cookie, xcb_generic_error_t **e);
extern xcb_list_installed_colormaps_cookie_t
xcb_list_installed_colormaps_unchecked(xcb_connection_t * c,
                                         xcb_list_installed_colormaps_cookie_t
                                         cookie, xcb_generic_error_t **e);

```

```

                                xcb_window_t window);
extern                                xcb_list_properties_cookie_t
xcb_list_properties(xcb_connection_t *
                                c,
                                xcb_window_t
                                window);
extern xcb_atom_t *xcb_list_properties_atoms(const
                                xcb_list_properties_reply_t
*
                                R);
extern xcb_generic_iterator_t xcb_list_properties_atoms_end(const
xcb_list_properties_reply_t
                                * R);
extern int xcb_list_properties_atoms_length(const
                                xcb_list_properties_reply_t *
                                R);
extern xcb_list_properties_reply_t
    *xcb_list_properties_reply(xcb_connection_t * c,
                                xcb_list_properties_cookie_t cookie,
                                xcb_generic_error_t * *e);
extern xcb_list_properties_cookie_t
xcb_list_properties_unchecked(xcb_connection_t * c, xcb_window_t
window);
extern xcb_lookup_color_cookie_t xcb_lookup_color(xcb_connection_t
* c,
                                xcb_colormap_t cmap,
                                uint16_t name_len,
                                const char *name);
extern                                xcb_lookup_color_reply_t
*xcb_lookup_color_reply(xcb_connection_t *
                                c,
                                cookie,
                                xcb_generic_error_t
                                * *e);
extern xcb_lookup_color_cookie_t
xcb_lookup_color_unchecked(xcb_connection_t * c, xcb_colormap_t
cmap,
                                uint16_t name_len, const char *name);
extern xcb_void_cookie_t xcb_map_subwindows(xcb_connection_t * c,
                                xcb_window_t window);
extern                                xcb_void_cookie_t
xcb_map_subwindows_checked(xcb_connection_t * c,
                                xcb_window_t window);
extern xcb_void_cookie_t xcb_map_window(xcb_connection_t * c,
                                xcb_window_t window);
extern xcb_void_cookie_t xcb_map_window_checked(xcb_connection_t *
c,
                                xcb_window_t window);
extern xcb_void_cookie_t xcb_no_operation(xcb_connection_t * c);
extern xcb_void_cookie_t xcb_no_operation_checked(xcb_connection_t
* c);
extern xcb_void_cookie_t xcb_open_font(xcb_connection_t * c,
                                xcb_font_t fid, uint16_t name_len,
                                const char *name);
extern xcb_void_cookie_t xcb_open_font_checked(xcb_connection_t *
c,
                                xcb_font_t fid,
                                uint16_t name_len,
                                const char *name);
extern xcb_generic_iterator_t xcb_pixmap_end(xcb_pixmap_iterator_t
i);
extern void xcb_pixmap_next(xcb_pixmap_iterator_t * i);

```

```

extern xcb_generic_iterator_t xcb_point_end(xcb_point_iterator_t
i);
extern void xcb_point_next(xcb_point_iterator_t * i);
extern xcb_void_cookie_t xcb_poly_arc(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc, uint32_t
arcs_len,
const xcb_arc_t * arcs);
extern xcb_void_cookie_t xcb_poly_arc_checked(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t arcs_len,
const xcb_arc_t * arcs);
extern xcb_void_cookie_t xcb_poly_fill_arc(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t arcs_len,
const xcb_arc_t * arcs);
extern xcb_void_cookie_t xcb_poly_fill_arc_checked(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t arcs_len,
const xcb_arc_t * arcs);
extern xcb_void_cookie_t xcb_poly_fill_rectangle(xcb_connection_t
* c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t rectangles_len,
const xcb_rectangle_t *
rectangles);
extern xcb_void_cookie_t xcb_poly_fill_rectangle_checked(xcb_connection_t *
c,
xcb_drawable_t
drawable,
xcb_gcontext_t gc,
uint32_t
rectangles_len,
const
xcb_rectangle_t *
rectangles);
extern xcb_void_cookie_t xcb_poly_line(xcb_connection_t * c,
uint8_t coordinate_mode,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t points_len,
const xcb_point_t * points);
extern xcb_void_cookie_t xcb_poly_line_checked(xcb_connection_t *
c,
uint8_t coordinate_mode,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t points_len,
const xcb_point_t * points);
extern xcb_void_cookie_t xcb_poly_point(xcb_connection_t * c,
uint8_t coordinate_mode,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t points_len,
const xcb_point_t * points);
extern xcb_void_cookie_t xcb_poly_point_checked(xcb_connection_t *
c,
uint8_t coordinate_mode,
xcb_drawable_t drawable,
xcb_gcontext_t gc,

```

```

uint32_t points_len,
const xcb_point_t *
points);
extern xcb_void_cookie_t xcb_poly_rectangle(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t rectangles_len,
const xcb_rectangle_t *
rectangles);
extern xcb_void_cookie_t xcb_poly_rectangle_checked(xcb_connection_t * c,
xcb_drawable_t
drawable,
xcb_gcontext_t gc,
uint32_t
rectangles_len,
const xcb_rectangle_t
*
rectangles);
extern xcb_void_cookie_t xcb_poly_segment(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t segments_len,
const xcb_segment_t * segments);
extern xcb_void_cookie_t xcb_poly_segment_checked(xcb_connection_t
* c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint32_t segments_len,
const xcb_segment_t *
segments);
extern xcb_void_cookie_t xcb_poly_text_16(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc, int16_t x,
int16_t y, uint32_t items_len,
const unsigned char *items);
extern xcb_void_cookie_t xcb_poly_text_16_checked(xcb_connection_t
* c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
int16_t x, int16_t y,
uint32_t items_len,
const unsigned char
*items);
extern xcb_void_cookie_t xcb_poly_text_8(xcb_connection_t * c,
xcb_drawable_t drawable,
xcb_gcontext_t gc, int16_t x,
int16_t y, uint32_t items_len,
const unsigned char *items);
extern xcb_void_cookie_t xcb_poly_text_8_checked(xcb_connection_t
* c,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
int16_t x, int16_t y,
uint32_t items_len,
const unsigned char
*items);
extern xcb_void_cookie_t xcb_put_image(xcb_connection_t * c,
uint8_t format,
xcb_drawable_t drawable,
xcb_gcontext_t gc, uint16_t width,
uint16_t height, int16_t dst_x,
int16_t dst_y, uint8_t left_pad,
uint8_t depth, uint32_t data_len,
const unsigned char *data);

```



```

extern xcb_void_cookie_t xcb_put_image_checked(xcb_connection_t *
c,
uint8_t format,
xcb_drawable_t drawable,
xcb_gcontext_t gc,
uint16_t width,
uint16_t height,
int16_t dst_x,
int16_t dst_y,
uint8_t left_pad,
uint8_t depth,
uint32_t data_len,
const unsigned char *data);
extern xcb_query_best_size_cookie_t
xcb_query_best_size(xcb_connection_t *
c, uint8_t class,
xcb_drawable_t
drawable,
uint16_t width,
uint16_t height);
extern xcb_query_best_size_reply_t
*xcb_query_best_size_reply(xcb_connection_t * c,
xcb_query_best_size_cookie_t cookie,
xcb_generic_error_t * *e);
extern xcb_query_best_size_cookie_t
xcb_query_best_size_unchecked(xcb_connection_t * c, uint8_t class,
xcb_drawable_t drawable, uint16_t width,
uint16_t height);
extern xcb_query_colors_cookie_t xcb_query_colors(xcb_connection_t
* c,
xcb_colormap_t cmap,
uint32_t pixels_len,
const uint32_t * pixels);
extern xcb_rgb_t *xcb_query_colors_colors(const
xcb_query_colors_reply_t *
R);
extern xcb_rgb_iterator_t xcb_query_colors_colors_iterator(const
xcb_query_colors_reply_t
* R);
extern int xcb_query_colors_colors_length(const
xcb_query_colors_reply_t *
R);
extern xcb_query_colors_reply_t
*xcb_query_colors_reply(xcb_connection_t *
c,
xcb_query_colors_cookie_t
cookie,
xcb_generic_error_t
* *e);
extern xcb_query_colors_cookie_t
xcb_query_colors_unchecked(xcb_connection_t * c, xcb_colormap_t
cmap,
uint32_t pixels_len, const uint32_t *
pixels);
extern xcb_query_extension_cookie_t
xcb_query_extension(xcb_connection_t *
c,
uint16_t name_len,
const char *name);
extern xcb_query_extension_reply_t
*xcb_query_extension_reply(xcb_connection_t * c,
xcb_query_extension_cookie_t cookie,
xcb_generic_error_t * *e);
extern xcb_query_extension_cookie_t

```

```

xcb_query_extension_unchecked(xcb_connection_t * c, uint16_t
name_len,
                                const char *name);
extern xcb_query_font_cookie_t xcb_query_font(xcb_connection_t * c,
                                              xcb_fontable_t font);
extern xcb_charinfo_t *xcb_query_font_char_infos(const
                                              xcb_query_font_reply_t *
                                              R);
extern
xcb_query_font_char_infos_iterator(const
xcb_query_font_reply_t
                                * R);
extern int xcb_query_font_char_infos_length(const
xcb_query_font_reply_t *
                                R);
extern xcb_fontprop_t *xcb_query_font_properties(const
                                              xcb_query_font_reply_t *
                                              R);
extern
xcb_query_font_properties_iterator(const
xcb_query_font_reply_t
                                * R);
extern int xcb_query_font_properties_length(const
xcb_query_font_reply_t *
                                R);
extern
*xcb_query_font_reply(xcb_connection_t * c,
xcb_query_font_cookie_t
                                cookie,
                                xcb_generic_error_t *
                                *e);
extern
xcb_query_font_unchecked(xcb_connection_t *
                                c,
                                xcb_fontable_t
                                font);
extern xcb_query_keymap_cookie_t xcb_query_keymap(xcb_connection_t
* c);
extern
*xcb_query_keymap_reply(xcb_connection_t *
                                c,
                                xcb_query_keymap_cookie_t
                                cookie,
                                xcb_generic_error_t
                                * *e);
extern xcb_query_keymap_unchecked(xcb_connection_t * c);
extern
xcb_query_pointer(xcb_connection_t * c,
                                xcb_window_t window);
extern
*xcb_query_pointer_reply(xcb_connection_t
                                * c,
                                xcb_query_pointer_cookie_t
                                cookie,
                                xcb_generic_error_t
                                * *e);
extern xcb_query_pointer_unchecked(xcb_connection_t * c, xcb_window_t
window);

```

```

extern xcb_query_text_extents_cookie_t
xcb_query_text_extents(xcb_connection_t * c, xcb_fontable_t font,
                      uint32_t string_len, const xcb_char2b_t *
string);
extern xcb_query_text_extents_reply_t
*xcb_query_text_extents_reply(xcb_connection_t * c,
                             xcb_query_text_extents_cookie_t
cookie,
                             xcb_generic_error_t * *e);
extern xcb_query_text_extents_cookie_t
xcb_query_text_extents_unchecked(xcb_connection_t * c,
xcb_fontable_t font,
                                uint32_t string_len,
                                const xcb_char2b_t * string);
extern xcb_query_tree_cookie_t xcb_query_tree(xcb_connection_t * c,
                                              xcb_window_t window);
extern xcb_window_t *xcb_query_tree_children(const
xcb_query_tree_reply_t *
R);
extern xcb_generic_iterator_t xcb_query_tree_children_end(const
xcb_query_tree_reply_t
* R);
extern int xcb_query_tree_children_length(const
xcb_query_tree_reply_t *
R);
extern xcb_query_tree_reply_t
*xcb_query_tree_reply(xcb_connection_t * c,
xcb_query_tree_cookie_t
cookie,
xcb_generic_error_t *
*e);
extern xcb_query_tree_cookie_t
xcb_query_tree_unchecked(xcb_connection_t *
c,
xcb_window_t
window);
extern xcb_void_cookie_t xcb_recolor_cursor(xcb_connection_t * c,
                                             xcb_cursor_t cursor,
                                             uint16_t fore_red,
                                             uint16_t fore_green,
                                             uint16_t fore_blue,
                                             uint16_t back_red,
                                             uint16_t back_green,
                                             uint16_t back_blue);
extern xcb_void_cookie_t
xcb_recolor_cursor_checked(xcb_connection_t * c,
                           xcb_cursor_t cursor,
                           uint16_t fore_red,
                           uint16_t fore_green,
                           uint16_t fore_blue,
                           uint16_t back_red,
                           uint16_t back_green,
                           uint16_t back_blue);
extern xcb_generic_iterator_t
xcb_rectangle_end(xcb_rectangle_iterator_t
i);
extern void xcb_rectangle_next(xcb_rectangle_iterator_t * i);
extern xcb_void_cookie_t xcb_reparent_window(xcb_connection_t * c,
                                              xcb_window_t window,
                                              xcb_window_t parent,
                                              int16_t x, int16_t y);
extern xcb_void_cookie_t
xcb_reparent_window_checked(xcb_connection_t * c,
                             xcb_window_t window,

```

```

xcb_window_t parent,
    int16_t x, int16_t y);
extern xcb_generic_iterator_t xcb_rgb_end(xcb_rgb_iterator_t i);
extern void xcb_rgb_next(xcb_rgb_iterator_t * i);
extern xcb_void_cookie_t xcb_rotate_properties(xcb_connection_t *
c,
    xcb_window_t window,
    uint16_t atoms_len,
    int16_t delta,
    const xcb_atom_t * atoms);
extern xcb_void_cookie_t
xcb_rotate_properties_checked(xcb_connection_t *
c,
    xcb_window_t window,
    uint16_t atoms_len,
    int16_t delta,
    const xcb_atom_t *
atoms);
extern xcb_depth_iterator_t
xcb_screen_allowed_depths_iterator(const
    xcb_screen_t
    * R);
extern int xcb_screen_allowed_depths_length(const xcb_screen_t *
R);
extern xcb_generic_iterator_t xcb_screen_end(xcb_screen_iterator_t
i);
extern void xcb_screen_next(xcb_screen_iterator_t * i);
extern xcb_generic_iterator_t
xcb_segment_end(xcb_segment_iterator_t i);
extern void xcb_segment_next(xcb_segment_iterator_t * i);
extern xcb_void_cookie_t xcb_send_event(xcb_connection_t * c,
    uint8_t propagate,
    xcb_window_t destination,
    uint32_t event_mask,
    const char *event);
extern xcb_void_cookie_t xcb_send_event_checked(xcb_connection_t *
c,
    uint8_t propagate,
    xcb_window_t destination,
    uint32_t event_mask,
    const char *event);
extern xcb_void_cookie_t xcb_set_access_control(xcb_connection_t *
c,
    uint8_t mode);
extern xcb_void_cookie_t
xcb_set_access_control_checked(xcb_connection_t *
c, uint8_t mode);
extern xcb_void_cookie_t xcb_set_clip_rectangles(xcb_connection_t
* c,
    uint8_t ordering,
    xcb_gcontext_t gc,
    int16_t clip_x_origin,
    int16_t clip_y_origin,
    uint32_t rectangles_len,
    const xcb_rectangle_t *
rectangles);
extern xcb_void_cookie_t
xcb_set_clip_rectangles_checked(xcb_connection_t *
c,
    uint8_t ordering,
    xcb_gcontext_t gc,
    int16_t
clip_x_origin,
    int16_t
clip_y_origin,
    uint32_t

```

```

rectangles_len,
const
xcb_rectangle_t *
rectangles);
extern xcb_void_cookie_t xcb_set_close_down_mode(xcb_connection_t
* c,
uint8_t mode);
extern xcb_void_cookie_t
xcb_set_close_down_mode_checked(xcb_connection_t *
c, uint8_t mode);
extern xcb_void_cookie_t xcb_set_dashes(xcb_connection_t * c,
xcb_gcontext_t gc,
uint16_t dash_offset,
uint16_t dashes_len,
const unsigned char *dashes);
extern xcb_void_cookie_t xcb_set_dashes_checked(xcb_connection_t *
c,
xcb_gcontext_t gc,
uint16_t dash_offset,
uint16_t dashes_len,
const unsigned char
*dashes);
extern xcb_void_cookie_t xcb_set_font_path(xcb_connection_t * c,
uint16_t font_qty,
uint32_t path_len,
const char *path);
extern xcb_void_cookie_t
xcb_set_font_path_checked(xcb_connection_t * c,
uint16_t font_qty,
uint32_t path_len,
const char *path);
extern xcb_void_cookie_t xcb_set_input_focus(xcb_connection_t * c,
uint8_t revert_to,
xcb_window_t focus,
xcb_timestamp_t time);
extern xcb_void_cookie_t
xcb_set_input_focus_checked(xcb_connection_t * c,
uint8_t revert_to,
xcb_window_t focus,
xcb_timestamp_t time);
extern xcb_set_modifier_mapping_cookie_t
xcb_set_modifier_mapping(xcb_connection_t * c,
uint8_t keycodes_per_modifier,
const xcb_keycode_t * keycodes);
extern xcb_set_modifier_mapping_reply_t
*xcb_set_modifier_mapping_reply(xcb_connection_t * c,
xcb_set_modifier_mapping_cookie_t
cookie, xcb_generic_error_t * *e);
extern xcb_set_modifier_mapping_cookie_t
xcb_set_modifier_mapping_unchecked(xcb_connection_t * c,
uint8_t keycodes_per_modifier,
const xcb_keycode_t * keycodes);
extern xcb_set_pointer_mapping_cookie_t
xcb_set_pointer_mapping(xcb_connection_t * c, uint8_t map_len,
const unsigned char *map);
extern xcb_set_pointer_mapping_reply_t
*xcb_set_pointer_mapping_reply(xcb_connection_t * c,
xcb_set_pointer_mapping_cookie_t
cookie,
xcb_generic_error_t * *e);
extern xcb_set_pointer_mapping_cookie_t
xcb_set_pointer_mapping_unchecked(xcb_connection_t * c, uint8_t
map_len,
const unsigned char *map);
extern xcb_void_cookie_t xcb_set_screen_saver(xcb_connection_t * c,
int16_t timeout,

```

```

        int16_t interval,
        uint8_t prefer_blanking,
        uint8_t allow_exposures);
extern
        xcb_void_cookie_t
xcb_set_screen_saver_checked(xcb_connection_t * c,
        int16_t timeout,
        int16_t interval,
        uint8_t
        prefer_blanking,
        uint8_t
        allow_exposures);
extern xcb_void_cookie_t xcb_set_selection_owner(xcb_connection_t
* c,
        xcb_window_t owner,
        xcb_atom_t selection,
        xcb_timestamp_t time);
extern
        xcb_void_cookie_t
xcb_set_selection_owner_checked(xcb_connection_t *
        c,
        xcb_window_t
        owner,
        xcb_atom_t
        selection,
        xcb_timestamp_t
        time);

extern xcb_generic_iterator_t
xcb_setup_authenticate_end(xcb_setup_authenticate_iterator_t i);
extern
        void
xcb_setup_authenticate_next(xcb_setup_authenticate_iterator_t *
        i);
extern
        char
        *xcb_setup_authenticate_reason(const
xcb_setup_authenticate_t *
        R);
extern
        xcb_generic_iterator_t
xcb_setup_authenticate_reason_end(const
        xcb_setup_authenticate_t
        * R);
extern int xcb_setup_authenticate_reason_length(const
        xcb_setup_authenticate_t *
        R);
extern xcb_generic_iterator_t xcb_setup_end(xcb_setup_iterator_t
i);
extern xcb_generic_iterator_t
xcb_setup_failed_end(xcb_setup_failed_iterator_t i);
extern void xcb_setup_failed_next(xcb_setup_failed_iterator_t * i);
extern char *xcb_setup_failed_reason(const xcb_setup_failed_t * R);
extern xcb_generic_iterator_t xcb_setup_failed_reason_end(const
        xcb_setup_failed_t
        * R);
extern int xcb_setup_failed_reason_length(const xcb_setup_failed_t
* R);
extern void xcb_setup_next(xcb_setup_iterator_t * i);
extern xcb_format_t *xcb_setup_pixmap_formats(const xcb_setup_t *
R);
extern
        xcb_format_iterator_t
xcb_setup_pixmap_formats_iterator(const
        xcb_setup_t
        * R);
extern int xcb_setup_pixmap_formats_length(const xcb_setup_t * R);
extern char *xcb_setup_request_authorization_protocol_data(const
        xcb_setup_request_t
        * R);
extern xcb_generic_iterator_t

```

```

xcb_setup_request_authorization_protocol_data_end(const
xcb_setup_request_t
                                                    * R);

extern
xcb_setup_request_authorization_protocol_data_length(const
xcb_setup_request_t
                                                    * R);

extern char *xcb_setup_request_authorization_protocol_name(const
xcb_setup_request_t
                                                    * R);

extern xcb_generic_iterator_t
xcb_setup_request_authorization_protocol_name_end(const
xcb_setup_request_t
                                                    * R);

extern
xcb_setup_request_authorization_protocol_name_length(const
xcb_setup_request_t
                                                    * R);

extern xcb_generic_iterator_t
xcb_setup_request_end(xcb_setup_request_iterator_t i);
extern void xcb_setup_request_next(xcb_setup_request_iterator_t *
i);
extern xcb_screen_iterator_t xcb_setup_roots_iterator(const
xcb_setup_t *
                                                    R);

extern int xcb_setup_roots_length(const xcb_setup_t * R);
extern char *xcb_setup_vendor(const xcb_setup_t * R);
extern xcb_generic_iterator_t xcb_setup_vendor_end(const
xcb_setup_t * R);
extern int xcb_setup_vendor_length(const xcb_setup_t * R);
extern xcb_void_cookie_t xcb_store_colors(xcb_connection_t * c,
xcb_colormap_t cmap,
uint32_t items_len,
const xcb_coloritem_t * items);
extern xcb_void_cookie_t xcb_store_colors_checked(xcb_connection_t
* c,
xcb_colormap_t cmap,
uint32_t items_len,
const xcb_coloritem_t *
items);
extern xcb_void_cookie_t xcb_store_named_color(xcb_connection_t *
c,
uint8_t flags,
xcb_colormap_t cmap,
uint32_t pixel,
uint16_t name_len,
const char *name);
extern xcb_void_cookie_t xcb_store_named_color_checked(xcb_connection_t *
c, uint8_t flags,
xcb_colormap_t cmap,
uint32_t pixel,
uint16_t name_len,
const char *name);

extern xcb_generic_iterator_t xcb_str_end(xcb_str_iterator_t i);
extern char *xcb_str_name(const xcb_str_t * R);
extern xcb_generic_iterator_t xcb_str_name_end(const xcb_str_t *
R);
extern int xcb_str_name_length(const xcb_str_t * R);
extern void xcb_str_next(xcb_str_iterator_t * i);
extern xcb_generic_iterator_t
xcb_timecoord_end(xcb_timecoord_iterator_t
i);

```

```

extern void xcb_timecoord_next(xcb_timecoord_iterator_t * i);
extern                                xcb_generic_iterator_t
xcb_timestamp_end(xcb_timestamp_iterator_t
                  i);
extern void xcb_timestamp_next(xcb_timestamp_iterator_t * i);
extern xcb_translate_coordinates_cookie_t
xcb_translate_coordinates(xcb_connection_t * c, xcb_window_t
src_window,
                        xcb_window_t dst_window, int16_t src_x,
                        int16_t src_y);
extern xcb_translate_coordinates_reply_t
*xcb_translate_coordinates_reply(xcb_connection_t * c,
                                xcb_translate_coordinates_cookie_t
                                cookie, xcb_generic_error_t * *e);
extern xcb_translate_coordinates_cookie_t
xcb_translate_coordinates_unchecked(xcb_connection_t * c,
                                xcb_window_t src_window,
                                xcb_window_t dst_window, int16_t
src_x,
                                int16_t src_y);
extern xcb_void_cookie_t xcb_ungrab_button(xcb_connection_t * c,
                                uint8_t button,
                                xcb_window_t grab_window,
                                uint16_t modifiers);
extern                                xcb_void_cookie_t
xcb_ungrab_button_checked(xcb_connection_t * c,
                                uint8_t button,
                                xcb_window_t grab_window,
                                uint16_t modifiers);
extern xcb_void_cookie_t xcb_ungrab_key(xcb_connection_t * c,
                                xcb_keycode_t key,
                                xcb_window_t grab_window,
                                uint16_t modifiers);
extern xcb_void_cookie_t xcb_ungrab_key_checked(xcb_connection_t *
c,
                                xcb_keycode_t key,
                                xcb_window_t grab_window,
                                uint16_t modifiers);
extern xcb_void_cookie_t xcb_ungrab_keyboard(xcb_connection_t * c,
                                xcb_timestamp_t time);
extern                                xcb_void_cookie_t
xcb_ungrab_keyboard_checked(xcb_connection_t * c,
                                xcb_timestamp_t time);
extern xcb_void_cookie_t xcb_ungrab_pointer(xcb_connection_t * c,
                                xcb_timestamp_t time);
extern                                xcb_void_cookie_t
xcb_ungrab_pointer_checked(xcb_connection_t * c,
                                xcb_timestamp_t time);
extern xcb_void_cookie_t xcb_ungrab_server(xcb_connection_t * c);
extern                                xcb_void_cookie_t
xcb_ungrab_server_checked(xcb_connection_t * c);
extern xcb_void_cookie_t xcb_uninstall_colormap(xcb_connection_t *
c,
                                xcb_colormap_t cmap);
extern                                xcb_void_cookie_t
xcb_uninstall_colormap_checked(xcb_connection_t *
c,
                                xcb_colormap_t
                                cmap);
extern xcb_void_cookie_t xcb_unmap_subwindows(xcb_connection_t * c,
                                xcb_window_t window);
extern                                xcb_void_cookie_t
xcb_unmap_subwindows_checked(xcb_connection_t * c,
                                xcb_window_t window);
extern xcb_void_cookie_t xcb_unmap_window(xcb_connection_t * c,

```



```

                                xcb_window_t window);
extern xcb_void_cookie_t xcb_unmap_window_checked(xcb_connection_t
* c,
                                xcb_window_t window);
extern                                xcb_generic_iterator_t
xcb_visualid_end(xcb_visualid_iterator_t i);
extern void xcb_visualid_next(xcb_visualid_iterator_t * i);
extern                                xcb_generic_iterator_t
xcb_visualtype_end(xcb_visualtype_iterator_t
i);
extern void xcb_visualtype_next(xcb_visualtype_iterator_t * i);
extern xcb_void_cookie_t xcb_warp_pointer(xcb_connection_t * c,
                                xcb_window_t src_window,
                                xcb_window_t dst_window,
                                int16_t src_x, int16_t src_y,
                                uint16_t src_width,
                                uint16_t src_height,
                                int16_t dst_x, int16_t dst_y);
extern xcb_void_cookie_t xcb_warp_pointer_checked(xcb_connection_t
* c,
                                xcb_window_t src_window,
                                xcb_window_t dst_window,
                                int16_t src_x,
                                int16_t src_y,
                                uint16_t src_width,
                                uint16_t src_height,
                                int16_t dst_x,
                                int16_t dst_y);
extern xcb_generic_iterator_t xcb_window_end(xcb_window_iterator_t
i);
extern void xcb_window_next(xcb_window_iterator_t * i);

```

6.21 Interface Definitions for libxcb

The interfaces defined on the following pages are included in libxcb and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 6.19 shall behave as described in the referenced base document.

III OpenGL Libraries

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

7 Libraries

7.1 Interfaces for libGL

Table 7-1 defines the library name and shared object name for the libGL library

Table 7-1 libGL Definition

Library:	libGL
SONAME:	libGL.so.1

The behavior of the interfaces in this library is specified by the following specifications:

[GLX] OpenGL Extensions
 [OGL 2.1] OpenGL 2.1
 [OGL ABI] OpenGL ABI

7.1.1 GL X interface

7.1.1.1 Interfaces for GL X interface

An LSB conforming implementation shall provide the generic functions for GL X interface specified in Table 7-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 7-2 libGL - GL X interface Function Interfaces

glXChooseFBConfig [GLX]	glXChooseVisual [GLX]	glXCopyContext [GLX]	glXCreateContext [GLX]
glXCreateGLXPixmap [GLX]	glXCreateNewContext [GLX]	glXCreatePbuffer [GLX]	glXCreatePixmap [GLX]
glXCreateWindow [GLX]	glXDestroyContext [GLX]	glXDestroyGLXPixmap [GLX]	glXDestroyPbuffer [GLX]
glXDestroyPixmap [GLX]	glXDestroyWindow [GLX]	glXFreeContextEXT [GLX]	glXGetClientString [GLX]
glXGetConfig [GLX]	glXGetContextIDEXT [GLX]	glXGetCurrentContext [GLX]	glXGetCurrentDisplay [GLX]
glXGetCurrentDrawable [GLX]	glXGetCurrentReadDrawable [GLX]	glXGetFBConfigAttrib [GLX]	glXGetFBConfigs [GLX]
glXGetProcAddress [OGL 2.1]	glXGetProcAddressARB [OGL ABI]	glXGetSelectedEvent [GLX]	glXGetVisualFromFBConfig [GLX]
glXImportContext [GLX]	glXIsDirect [GLX]	glXMakeContextCurrent [GLX]	glXMakeCurrent [GLX]
glXQueryContext [GLX]	glXQueryContextInfoEXT [GLX]	glXQueryDrawable [GLX]	glXQueryExtension [GLX]
glXQueryExtensionsString [GLX]	glXQueryServerString [GLX]	glXQueryVersion [GLX]	glXSelectEvent [GLX]

glXSwapBuffers [GLX]	glXUseXFont [GLX]	glXWaitGL [GLX]	glXWaitX [GLX]

7.1.2 OpenGL

7.1.2.1 Interfaces for OpenGL

An LSB conforming implementation shall provide the generic functions for OpenGL specified in Table 7-3, with the full mandatory functionality as described in the referenced underlying specification.

Table 7-3 libGL - OpenGL Function Interfaces

glAccum [OGL 2.1]	glActiveTexture [OGL 2.1]	glActiveTexture ARB [OGL 2.1]	glAlphaFunc [OGL 2.1]
glAreTexturesResident [OGL 2.1]	glArrayElement [OGL 2.1]	glAttachShader [OGL 2.1]	glBegin [OGL 2.1]
glBeginQuery [OGL 2.1]	glBindAttribLocation [OGL 2.1]	glBindBuffer [OGL 2.1]	glBindTexture [OGL 2.1]
glBitmap [OGL 2.1]	glBlendColor [OGL 2.1]	glBlendEquation [OGL 2.1]	glBlendEquation Separate [OGL 2.1]
glBlendFunc [OGL 2.1]	glBlendFuncSeparate [OGL 2.1]	glBufferData [OGL 2.1]	glBufferSubData [OGL 2.1]
glCallList [OGL 2.1]	glCallLists [OGL 2.1]	glClear [OGL 2.1]	glClearAccum [OGL 2.1]
glClearColor [OGL 2.1]	glClearDepth [OGL 2.1]	glClearIndex [OGL 2.1]	glClearStencil [OGL 2.1]
glClientActiveTexture [OGL 2.1]	glClientActiveTexture ARB [OGL 2.1]	glClipPlane [OGL 2.1]	glColor3b [OGL 2.1]
glColor3bv [OGL 2.1]	glColor3d [OGL 2.1]	glColor3dv [OGL 2.1]	glColor3f [OGL 2.1]
glColor3fv [OGL 2.1]	glColor3i [OGL 2.1]	glColor3iv [OGL 2.1]	glColor3s [OGL 2.1]
glColor3sv [OGL 2.1]	glColor3ub [OGL 2.1]	glColor3ubv [OGL 2.1]	glColor3ui [OGL 2.1]
glColor3uiv [OGL 2.1]	glColor3us [OGL 2.1]	glColor3usv [OGL 2.1]	glColor4b [OGL 2.1]
glColor4bv [OGL 2.1]	glColor4d [OGL 2.1]	glColor4dv [OGL 2.1]	glColor4f [OGL 2.1]
glColor4fv [OGL 2.1]	glColor4i [OGL 2.1]	glColor4iv [OGL 2.1]	glColor4s [OGL 2.1]
glColor4sv [OGL 2.1]	glColor4ub [OGL 2.1]	glColor4ubv [OGL 2.1]	glColor4ui [OGL 2.1]

glColor4uiv [OGL 2.1]	glColor4us [OGL 2.1]	glColor4usv [OGL 2.1]	glColorMask [OGL 2.1]
glColorMaterial [OGL 2.1]	glColorPointer [OGL 2.1]	glColorSubTable [OGL 2.1]	glColorTable [OGL 2.1]
glColorTableParameterfv [OGL 2.1]	glColorTableParameteriv [OGL 2.1]	glCompileShader [OGL 2.1]	glCompressedTexImage1D [OGL 2.1]
glCompressedTexImage2D [OGL 2.1]	glCompressedTexImage3D [OGL 2.1]	glCompressedTexSubImage1D [OGL 2.1]	glCompressedTexSubImage2D [OGL 2.1]
glCompressedTexSubImage3D [OGL 2.1]	glConvolutionFilter1D [OGL 2.1]	glConvolutionFilter2D [OGL 2.1]	glConvolutionParameterf [OGL 2.1]
glConvolutionParameterfv [OGL 2.1]	glConvolutionParameteri [OGL 2.1]	glConvolutionParameteriv [OGL 2.1]	glCopyColorSubTable [OGL 2.1]
glCopyColorTable [OGL 2.1]	glCopyConvolutionFilter1D [OGL 2.1]	glCopyConvolutionFilter2D [OGL 2.1]	glCopyPixels [OGL 2.1]
glCopyTexImage1D [OGL 2.1]	glCopyTexImage2D [OGL 2.1]	glCopyTexSubImage1D [OGL 2.1]	glCopyTexSubImage2D [OGL 2.1]
glCopyTexSubImage3D [OGL 2.1]	glCreateProgram [OGL 2.1]	glCreateShader [OGL 2.1]	glCullFace [OGL 2.1]
glDeleteBuffers [OGL 2.1]	glDeleteLists [OGL 2.1]	glDeleteProgram [OGL 2.1]	glDeleteQueries [OGL 2.1]
glDeleteShader [OGL 2.1]	glDeleteTextures [OGL 2.1]	glDepthFunc [OGL 2.1]	glDepthMask [OGL 2.1]
glDepthRange [OGL 2.1]	glDetachShader [OGL 2.1]	glDisable [OGL 2.1]	glDisableClientState [OGL 2.1]
glDisableVertexAttribArray [OGL 2.1]	glDrawArrays [OGL 2.1]	glDrawBuffer [OGL 2.1]	glDrawBuffers [OGL 2.1]
glDrawElements [OGL 2.1]	glDrawPixels [OGL 2.1]	glDrawRangeElements [OGL 2.1]	glEdgeFlag [OGL 2.1]
glEdgeFlagPointer [OGL 2.1]	glEdgeFlagv [OGL 2.1]	glEnable [OGL 2.1]	glEnableClientState [OGL 2.1]
glEnableVertexAttribArray [OGL 2.1]	glEnd [OGL 2.1]	glEndList [OGL 2.1]	glEvalCoord1d [OGL 2.1]
glEvalCoord1dv [OGL 2.1]	glEvalCoord1f [OGL 2.1]	glEvalCoord1fv [OGL 2.1]	glEvalCoord2d [OGL 2.1]
glEvalCoord2dv [OGL 2.1]	glEvalCoord2f [OGL 2.1]	glEvalCoord2fv [OGL 2.1]	glEvalMesh1 [OGL 2.1]

glEvalMesh2 [OGL 2.1]	glEvalPoint1 [OGL 2.1]	glEvalPoint2 [OGL 2.1]	glFeedbackBuffer [OGL 2.1]
glFinish [OGL 2.1]	glFlush [OGL 2.1]	glFogCoordPoint er [OGL 2.1]	glFogCoorddd [OGL 2.1]
glFogCoorddv [OGL 2.1]	glFogCoordf [OGL 2.1]	glFogCoordfv [OGL 2.1]	glFogf [OGL 2.1]
glFogfv [OGL 2.1]	glFogi [OGL 2.1]	glFogiv [OGL 2.1]	glFrontFace [OGL 2.1]
glFrustum [OGL 2.1]	glGenBuffers [OGL 2.1]	glGenLists [OGL 2.1]	glGenQueries [OGL 2.1]
glGenTextures [OGL 2.1]	glGetActiveAttri b [OGL 2.1]	glGetActiveUnif orm [OGL 2.1]	glGetAttachedSh aders [OGL 2.1]
glGetAttribLocat ion [OGL 2.1]	glGetBooleantv [OGL 2.1]	glGetBufferPara meteriv [OGL 2.1]	glGetBufferPoint erv [OGL 2.1]
glGetBufferSubD ata [OGL 2.1]	glGetClipPlane [OGL 2.1]	glGetColorTable [OGL 2.1]	glGetColorTable Parameterfv [OGL 2.1]
glGetColorTable Parameteriv [OGL 2.1]	glGetCompressed TexImage [OGL 2.1]	glGetConvolutio nFilter [OGL 2.1]	glGetConvolutio nParameterfv [OGL 2.1]
glGetConvolutio nParameteriv [OGL 2.1]	glGetDoublev [OGL 2.1]	glGetError [OGL 2.1]	glGetFloatv [OGL 2.1]
glGetHistogram [OGL 2.1]	glGetHistogramP arameterfv [OGL 2.1]	glGetHistogramP arameteriv [OGL 2.1]	glGetIntegerv [OGL 2.1]
glGetLightfv [OGL 2.1]	glGetLightiv [OGL 2.1]	glGetMapdv [OGL 2.1]	glGetMapfv [OGL 2.1]
glGetMapiv [OGL 2.1]	glGetMaterialfv [OGL 2.1]	glGetMaterialiv [OGL 2.1]	glGetMinmax [OGL 2.1]
glGetMinmaxPar ameterfv [OGL 2.1]	glGetMinmaxPar ameteriv [OGL 2.1]	glGetPixelMapfv [OGL 2.1]	glGetPixelMapui v [OGL 2.1]
glGetPixelMapus v [OGL 2.1]	glGetPointerv [OGL 2.1]	glGetPolygonSti pple [OGL 2.1]	glGetProgramInf oLog [OGL 2.1]
glGetProgramiv [OGL 2.1]	glGetQueryObjec tiv [OGL 2.1]	glGetQueryObjec tuiv [OGL 2.1]	glGetQueryiv [OGL 2.1]
glGetSeparableFi lter [OGL 2.1]	glGetShaderInfo Log [OGL 2.1]	glGetShaderSour ce [OGL 2.1]	glGetShaderiv [OGL 2.1]
glGetString [OGL 2.1]	glGetTexEnvfv [OGL 2.1]	glGetTexEnviv [OGL 2.1]	glGetTexGendv [OGL 2.1]

glGetTexGenfv [OGL 2.1]	glGetTexGeniv [OGL 2.1]	glGetTexImage [OGL 2.1]	glGetTexLevelParameterfv [OGL 2.1]
glGetTexLevelParameteriv [OGL 2.1]	glGetTexParameterfv [OGL 2.1]	glGetTexParameteriv [OGL 2.1]	glGetUniformLocation [OGL 2.1]
glGetUniformfv [OGL 2.1]	glGetUniformiv [OGL 2.1]	glGetVertexAttribPointer [OGL 2.1]	glGetVertexAttribbv [OGL 2.1]
glGetVertexAttribfv [OGL 2.1]	glGetVertexAttribiv [OGL 2.1]	glHint [OGL 2.1]	glHistogram [OGL 2.1]
glIndexMask [OGL 2.1]	glIndexPointer [OGL 2.1]	glIndexd [OGL 2.1]	glIndexdv [OGL 2.1]
glIndexf [OGL 2.1]	glIndexfv [OGL 2.1]	glIndexi [OGL 2.1]	glIndexiv [OGL 2.1]
glIndexs [OGL 2.1]	glIndexsv [OGL 2.1]	glIndexub [OGL 2.1]	glIndexubv [OGL 2.1]
glInitNames [OGL 2.1]	glInterleavedArrays [OGL 2.1]	glIsBuffer [OGL 2.1]	glIsEnabled [OGL 2.1]
glIsList [OGL 2.1]	glIsProgram [OGL 2.1]	glIsQuery [OGL 2.1]	glIsShader [OGL 2.1]
glIsTexture [OGL 2.1]	glLightModelf [OGL 2.1]	glLightModelfv [OGL 2.1]	glLightModeli [OGL 2.1]
glLightModeliv [OGL 2.1]	glLightf [OGL 2.1]	glLightfv [OGL 2.1]	glLighti [OGL 2.1]
glLightiv [OGL 2.1]	glLineStipple [OGL 2.1]	glLineWidth [OGL 2.1]	glLinkProgram [OGL 2.1]
glListBase [OGL 2.1]	glLoadIdentity [OGL 2.1]	glLoadMatrixd [OGL 2.1]	glLoadMatrixf [OGL 2.1]
glLoadName [OGL 2.1]	glLoadTransposeMatrixd [OGL 2.1]	glLoadTransposeMatrixf [OGL 2.1]	glLogicOp [OGL 2.1]
glMap1d [OGL 2.1]	glMap1f [OGL 2.1]	glMap2d [OGL 2.1]	glMap2f [OGL 2.1]
glMapBuffer [OGL 2.1]	glMapGrid1d [OGL 2.1]	glMapGrid1f [OGL 2.1]	glMapGrid2d [OGL 2.1]
glMapGrid2f [OGL 2.1]	glMaterialf [OGL 2.1]	glMaterialfv [OGL 2.1]	glMateriali [OGL 2.1]
glMaterialiv [OGL 2.1]	glMatrixMode [OGL 2.1]	glMinmax [OGL 2.1]	glMultMatrixd [OGL 2.1]
glMultMatrixf [OGL 2.1]	glMultTransposeMatrixd [OGL 2.1]	glMultTransposeMatrixf [OGL 2.1]	glMultiDrawArrays [OGL 2.1]

glMultiDrawElements [OGL 2.1]	glMultiTexCoord1d [OGL 2.1]	glMultiTexCoord1dARB [OGL 2.1]	glMultiTexCoord1dv [OGL 2.1]
glMultiTexCoord1dvARB [OGL 2.1]	glMultiTexCoord1f [OGL 2.1]	glMultiTexCoord1fARB [OGL 2.1]	glMultiTexCoord1fv [OGL 2.1]
glMultiTexCoord1fvARB [OGL 2.1]	glMultiTexCoord1i [OGL 2.1]	glMultiTexCoord1iARB [OGL 2.1]	glMultiTexCoord1iv [OGL 2.1]
glMultiTexCoord1ivARB [OGL 2.1]	glMultiTexCoord1s [OGL 2.1]	glMultiTexCoord1sARB [OGL 2.1]	glMultiTexCoord1sv [OGL 2.1]
glMultiTexCoord1svARB [OGL 2.1]	glMultiTexCoord2d [OGL 2.1]	glMultiTexCoord2dARB [OGL 2.1]	glMultiTexCoord2dv [OGL 2.1]
glMultiTexCoord2dvARB [OGL 2.1]	glMultiTexCoord2f [OGL 2.1]	glMultiTexCoord2fARB [OGL 2.1]	glMultiTexCoord2fv [OGL 2.1]
glMultiTexCoord2fvARB [OGL 2.1]	glMultiTexCoord2i [OGL 2.1]	glMultiTexCoord2iARB [OGL 2.1]	glMultiTexCoord2iv [OGL 2.1]
glMultiTexCoord2ivARB [OGL 2.1]	glMultiTexCoord2s [OGL 2.1]	glMultiTexCoord2sARB [OGL 2.1]	glMultiTexCoord2sv [OGL 2.1]
glMultiTexCoord2svARB [OGL 2.1]	glMultiTexCoord3d [OGL 2.1]	glMultiTexCoord3dARB [OGL 2.1]	glMultiTexCoord3dv [OGL 2.1]
glMultiTexCoord3dvARB [OGL 2.1]	glMultiTexCoord3f [OGL 2.1]	glMultiTexCoord3fARB [OGL 2.1]	glMultiTexCoord3fv [OGL 2.1]
glMultiTexCoord3fvARB [OGL 2.1]	glMultiTexCoord3i [OGL 2.1]	glMultiTexCoord3iARB [OGL 2.1]	glMultiTexCoord3iv [OGL 2.1]
glMultiTexCoord3ivARB [OGL 2.1]	glMultiTexCoord3s [OGL 2.1]	glMultiTexCoord3sARB [OGL 2.1]	glMultiTexCoord3sv [OGL 2.1]
glMultiTexCoord3svARB [OGL 2.1]	glMultiTexCoord4d [OGL 2.1]	glMultiTexCoord4dARB [OGL 2.1]	glMultiTexCoord4dv [OGL 2.1]
glMultiTexCoord4dvARB [OGL 2.1]	glMultiTexCoord4f [OGL 2.1]	glMultiTexCoord4fARB [OGL 2.1]	glMultiTexCoord4fv [OGL 2.1]
glMultiTexCoord4fvARB [OGL 2.1]	glMultiTexCoord4i [OGL 2.1]	glMultiTexCoord4iARB [OGL 2.1]	glMultiTexCoord4iv [OGL 2.1]

glMultiTexCoord4ivARB [OGL 2.1]	glMultiTexCoord4s [OGL 2.1]	glMultiTexCoord4sARB [OGL 2.1]	glMultiTexCoord4sv [OGL 2.1]
glMultiTexCoord4svARB [OGL 2.1]	glNewList [OGL 2.1]	glNormal3b [OGL 2.1]	glNormal3bv [OGL 2.1]
glNormal3d [OGL 2.1]	glNormal3dv [OGL 2.1]	glNormal3f [OGL 2.1]	glNormal3fv [OGL 2.1]
glNormal3i [OGL 2.1]	glNormal3iv [OGL 2.1]	glNormal3s [OGL 2.1]	glNormal3sv [OGL 2.1]
glNormalPointer [OGL 2.1]	glOrtho [OGL 2.1]	glPassThrough [OGL 2.1]	glPixelMapfv [OGL 2.1]
glPixelMapuiv [OGL 2.1]	glPixelMapusv [OGL 2.1]	glPixelStoref [OGL 2.1]	glPixelStorei [OGL 2.1]
glPixelTransferf [OGL 2.1]	glPixelTransferi [OGL 2.1]	glPixelZoom [OGL 2.1]	glPointParameterf [OGL 2.1]
glPointParameterfv [OGL 2.1]	glPointParameteri [OGL 2.1]	glPointParameteriv [OGL 2.1]	glPointSize [OGL 2.1]
glPolygonMode [OGL 2.1]	glPolygonOffset [OGL 2.1]	glPolygonStipple [OGL 2.1]	glPopAttrib [OGL 2.1]
glPopClientAttrib [OGL 2.1]	glPopMatrix [OGL 2.1]	glPopName [OGL 2.1]	glPrioritizeTextures [OGL 2.1]
glPushAttrib [OGL 2.1]	glPushClientAttrib [OGL 2.1]	glPushMatrix [OGL 2.1]	glPushName [OGL 2.1]
glRasterPos2d [OGL 2.1]	glRasterPos2dv [OGL 2.1]	glRasterPos2f [OGL 2.1]	glRasterPos2fv [OGL 2.1]
glRasterPos2i [OGL 2.1]	glRasterPos2iv [OGL 2.1]	glRasterPos2s [OGL 2.1]	glRasterPos2sv [OGL 2.1]
glRasterPos3d [OGL 2.1]	glRasterPos3dv [OGL 2.1]	glRasterPos3f [OGL 2.1]	glRasterPos3fv [OGL 2.1]
glRasterPos3i [OGL 2.1]	glRasterPos3iv [OGL 2.1]	glRasterPos3s [OGL 2.1]	glRasterPos3sv [OGL 2.1]
glRasterPos4d [OGL 2.1]	glRasterPos4dv [OGL 2.1]	glRasterPos4f [OGL 2.1]	glRasterPos4fv [OGL 2.1]
glRasterPos4i [OGL 2.1]	glRasterPos4iv [OGL 2.1]	glRasterPos4s [OGL 2.1]	glRasterPos4sv [OGL 2.1]
glReadBuffer [OGL 2.1]	glReadPixels [OGL 2.1]	glRectd [OGL 2.1]	glRectdv [OGL 2.1]
glRectf [OGL 2.1]	glRectfv [OGL 2.1]	glRecti [OGL 2.1]	glRectiv [OGL 2.1]
glRects [OGL 2.1]	glRectsv [OGL 2.1]	glRenderMode [OGL 2.1]	glResetHistogram [OGL 2.1]

glResetMinmax [OGL 2.1]	glRotated [OGL 2.1]	glRotatef [OGL 2.1]	glSampleCovera ge [OGL 2.1]
glScaled [OGL 2.1]	glScalef [OGL 2.1]	glScissor [OGL 2.1]	glSecondaryColo r3b [OGL 2.1]
glSecondaryColo r3bv [OGL 2.1]	glSecondaryColo r3d [OGL 2.1]	glSecondaryColo r3dv [OGL 2.1]	glSecondaryColo r3f [OGL 2.1]
glSecondaryColo r3fv [OGL 2.1]	glSecondaryColo r3i [OGL 2.1]	glSecondaryColo r3iv [OGL 2.1]	glSecondaryColo r3s [OGL 2.1]
glSecondaryColo r3sv [OGL 2.1]	glSecondaryColo r3ub [OGL 2.1]	glSecondaryColo r3ubv [OGL 2.1]	glSecondaryColo r3ui [OGL 2.1]
glSecondaryColo r3uiv [OGL 2.1]	glSecondaryColo r3us [OGL 2.1]	glSecondaryColo r3usv [OGL 2.1]	glSecondaryColo rPointer [OGL 2.1]
glSelectBuffer [OGL 2.1]	glSeparableFilter 2D [OGL 2.1]	glShadeModel [OGL 2.1]	glShaderSource [OGL 2.1]
glStencilFunc [OGL 2.1]	glStencilFuncSep arate [OGL 2.1]	glStencilMask [OGL 2.1]	glStencilMaskSe parate [OGL 2.1]
glStencilOp [OGL 2.1]	glStencilOpSepar ate [OGL 2.1]	glTexCoord1d [OGL 2.1]	glTexCoord1dv [OGL 2.1]
glTexCoord1f [OGL 2.1]	glTexCoord1fv [OGL 2.1]	glTexCoord1i [OGL 2.1]	glTexCoord1iv [OGL 2.1]
glTexCoord1s [OGL 2.1]	glTexCoord1sv [OGL 2.1]	glTexCoord2d [OGL 2.1]	glTexCoord2dv [OGL 2.1]
glTexCoord2f [OGL 2.1]	glTexCoord2fv [OGL 2.1]	glTexCoord2i [OGL 2.1]	glTexCoord2iv [OGL 2.1]
glTexCoord2s [OGL 2.1]	glTexCoord2sv [OGL 2.1]	glTexCoord3d [OGL 2.1]	glTexCoord3dv [OGL 2.1]
glTexCoord3f [OGL 2.1]	glTexCoord3fv [OGL 2.1]	glTexCoord3i [OGL 2.1]	glTexCoord3iv [OGL 2.1]
glTexCoord3s [OGL 2.1]	glTexCoord3sv [OGL 2.1]	glTexCoord4d [OGL 2.1]	glTexCoord4dv [OGL 2.1]
glTexCoord4f [OGL 2.1]	glTexCoord4fv [OGL 2.1]	glTexCoord4i [OGL 2.1]	glTexCoord4iv [OGL 2.1]
glTexCoord4s [OGL 2.1]	glTexCoord4sv [OGL 2.1]	glTexCoordPoint er [OGL 2.1]	glTexEnvf [OGL 2.1]
glTexEnvfv [OGL 2.1]	glTexEnvi [OGL 2.1]	glTexEnviv [OGL 2.1]	glTexGend [OGL 2.1]
glTexGendv [OGL 2.1]	glTexGenf [OGL 2.1]	glTexGenfv [OGL 2.1]	glTexGeni [OGL 2.1]
glTexGeniv [OGL 2.1]	glTexImage1D [OGL 2.1]	glTexImage2D [OGL 2.1]	glTexImage3D [OGL 2.1]

glTexParameterf [OGL 2.1]	glTexParameterfv [OGL 2.1]	glTexParameteriv [OGL 2.1]	glTexParameteriv [OGL 2.1]
glTexSubImage1D [OGL 2.1]	glTexSubImage2D [OGL 2.1]	glTexSubImage3D [OGL 2.1]	glTranslated [OGL 2.1]
glTranslatef [OGL 2.1]	glUniform1f [OGL 2.1]	glUniform1fv [OGL 2.1]	glUniform1i [OGL 2.1]
glUniform1iv [OGL 2.1]	glUniform2f [OGL 2.1]	glUniform2fv [OGL 2.1]	glUniform2i [OGL 2.1]
glUniform2iv [OGL 2.1]	glUniform3f [OGL 2.1]	glUniform3fv [OGL 2.1]	glUniform3i [OGL 2.1]
glUniform3iv [OGL 2.1]	glUniform4f [OGL 2.1]	glUniform4fv [OGL 2.1]	glUniform4i [OGL 2.1]
glUniform4iv [OGL 2.1]	glUniformMatrix2fv [OGL 2.1]	glUniformMatrix2x3fv [OGL 2.1]	glUniformMatrix2x4fv [OGL 2.1]
glUniformMatrix3fv [OGL 2.1]	glUniformMatrix3x2fv [OGL 2.1]	glUniformMatrix3x4fv [OGL 2.1]	glUniformMatrix4fv [OGL 2.1]
glUniformMatrix4x2fv [OGL 2.1]	glUniformMatrix4x3fv [OGL 2.1]	glUnmapBuffer [OGL 2.1]	glUseProgram [OGL 2.1]
glValidateProgram [OGL 2.1]	glVertex2d [OGL 2.1]	glVertex2dv [OGL 2.1]	glVertex2f [OGL 2.1]
glVertex2fv [OGL 2.1]	glVertex2i [OGL 2.1]	glVertex2iv [OGL 2.1]	glVertex2s [OGL 2.1]
glVertex2sv [OGL 2.1]	glVertex3d [OGL 2.1]	glVertex3dv [OGL 2.1]	glVertex3f [OGL 2.1]
glVertex3fv [OGL 2.1]	glVertex3i [OGL 2.1]	glVertex3iv [OGL 2.1]	glVertex3s [OGL 2.1]
glVertex3sv [OGL 2.1]	glVertex4d [OGL 2.1]	glVertex4dv [OGL 2.1]	glVertex4f [OGL 2.1]
glVertex4fv [OGL 2.1]	glVertex4i [OGL 2.1]	glVertex4iv [OGL 2.1]	glVertex4s [OGL 2.1]
glVertex4sv [OGL 2.1]	glVertexAttrib1d [OGL 2.1]	glVertexAttrib1dv [OGL 2.1]	glVertexAttrib1f [OGL 2.1]
glVertexAttrib1fv [OGL 2.1]	glVertexAttrib1s [OGL 2.1]	glVertexAttrib1sv [OGL 2.1]	glVertexAttrib2d [OGL 2.1]
glVertexAttrib2dv [OGL 2.1]	glVertexAttrib2f [OGL 2.1]	glVertexAttrib2fv [OGL 2.1]	glVertexAttrib2s [OGL 2.1]
glVertexAttrib2sv [OGL 2.1]	glVertexAttrib3d [OGL 2.1]	glVertexAttrib3dv [OGL 2.1]	glVertexAttrib3f [OGL 2.1]
glVertexAttrib3fv [OGL 2.1]	glVertexAttrib3s [OGL 2.1]	glVertexAttrib3sv [OGL 2.1]	glVertexAttrib4Nbv [OGL 2.1]
glVertexAttrib4Niv [OGL 2.1]	glVertexAttrib4Nsv [OGL 2.1]	glVertexAttrib4Nub [OGL 2.1]	glVertexAttrib4Nubv [OGL 2.1]

glVertexAttrib4Nuiv [OGL 2.1]	glVertexAttrib4Nusv [OGL 2.1]	glVertexAttrib4bv [OGL 2.1]	glVertexAttrib4dv [OGL 2.1]
glVertexAttrib4dv [OGL 2.1]	glVertexAttrib4fv [OGL 2.1]	glVertexAttrib4fv [OGL 2.1]	glVertexAttrib4iv [OGL 2.1]
glVertexAttrib4s [OGL 2.1]	glVertexAttrib4sv [OGL 2.1]	glVertexAttrib4ubv [OGL 2.1]	glVertexAttrib4uiv [OGL 2.1]
glVertexAttrib4usv [OGL 2.1]	glVertexAttribPointer [OGL 2.1]	glVertexPointer [OGL 2.1]	glViewport [OGL 2.1]
glWindowPos2d [OGL 2.1]	glWindowPos2dv [OGL 2.1]	glWindowPos2f [OGL 2.1]	glWindowPos2fv [OGL 2.1]
glWindowPos2i [OGL 2.1]	glWindowPos2iv [OGL 2.1]	glWindowPos2s [OGL 2.1]	glWindowPos2sv [OGL 2.1]
glWindowPos3d [OGL 2.1]	glWindowPos3dv [OGL 2.1]	glWindowPos3f [OGL 2.1]	glWindowPos3fv [OGL 2.1]
glWindowPos3i [OGL 2.1]	glWindowPos3iv [OGL 2.1]	glWindowPos3s [OGL 2.1]	glWindowPos3sv [OGL 2.1]

7.2 Data Definitions for libGL

This section defines global identifiers and their values that are associated with interfaces contained in libGL. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

7.2.1 GL/gl.h

```
#define GL_VERSION_1_1 1
#define GL_VERSION_1_2 1
#define GL_VERSION_1_3 1

typedef unsigned int GLenum;
typedef unsigned char GLboolean;
typedef unsigned int GLbitfield;
typedef void GLvoid;
typedef char GLbyte;
typedef short GLshort;
typedef int GLint;
typedef unsigned char GLubyte;
typedef unsigned short GLushort;
typedef unsigned int GLuint;
typedef int GLsizei;
typedef float GLfloat;
```

```

typedef float GLclampf;
typedef double GLdouble;
typedef double GLclampd;

#define GL_BYTE 0x1400
#define GL_UNSIGNED_BYTE 0x1401
#define GL_SHORT 0x1402
#define GL_UNSIGNED_SHORT 0x1403
#define GL_INT 0x1404
#define GL_UNSIGNED_INT 0x1405
#define GL_FLOAT 0x1406
#define GL_2_BYTES 0x1407
#define GL_3_BYTES 0x1408
#define GL_4_BYTES 0x1409
#define GL_DOUBLE 0x140A

#define GL_POINTS 0x0000
#define GL_LINES 0x0001
#define GL_LINE_LOOP 0x0002
#define GL_LINE_STRIP 0x0003
#define GL_TRIANGLES 0x0004
#define GL_TRIANGLE_STRIP 0x0005
#define GL_TRIANGLE_FAN 0x0006
#define GL_QUADS 0x0007
#define GL_QUAD_STRIP 0x0008
#define GL_POLYGON 0x0009

#define GL_V2F 0x2A20
#define GL_V3F 0x2A21
#define GL_C4UB_V2F 0x2A22
#define GL_C4UB_V3F 0x2A23
#define GL_C3F_V3F 0x2A24
#define GL_N3F_V3F 0x2A25
#define GL_C4F_N3F_V3F 0x2A26
#define GL_T2F_V3F 0x2A27
#define GL_T4F_V4F 0x2A28
#define GL_T2F_C4UB_V3F 0x2A29
#define GL_T2F_C3F_V3F 0x2A2A
#define GL_T2F_N3F_V3F 0x2A2B
#define GL_T2F_C4F_N3F_V3F 0x2A2C
#define GL_T4F_C4F_N3F_V4F 0x2A2D
#define GL_VERTEX_ARRAY 0x8074
#define GL_NORMAL_ARRAY 0x8075
#define GL_COLOR_ARRAY 0x8076
#define GL_INDEX_ARRAY 0x8077
#define GL_TEXTURE_COORD_ARRAY 0x8078
#define GL_EDGE_FLAG_ARRAY 0x8079
#define GL_VERTEX_ARRAY_SIZE 0x807A
#define GL_VERTEX_ARRAY_TYPE 0x807B
#define GL_VERTEX_ARRAY_STRIDE 0x807C
#define GL_NORMAL_ARRAY_TYPE 0x807E
#define GL_NORMAL_ARRAY_STRIDE 0x807F
#define GL_COLOR_ARRAY_SIZE 0x8081
#define GL_COLOR_ARRAY_TYPE 0x8082
#define GL_COLOR_ARRAY_STRIDE 0x8083
#define GL_INDEX_ARRAY_TYPE 0x8085
#define GL_INDEX_ARRAY_STRIDE 0x8086
#define GL_TEXTURE_COORD_ARRAY_SIZE 0x8088
#define GL_TEXTURE_COORD_ARRAY_TYPE 0x8089
#define GL_TEXTURE_COORD_ARRAY_STRIDE 0x808A
#define GL_EDGE_FLAG_ARRAY_STRIDE 0x808C
#define GL_VERTEX_ARRAY_POINTER 0x808E
#define GL_NORMAL_ARRAY_POINTER 0x808F
#define GL_COLOR_ARRAY_POINTER 0x8090
#define GL_INDEX_ARRAY_POINTER 0x8091
#define GL_TEXTURE_COORD_ARRAY_POINTER 0x8092

```

```

#define GL_EDGE_FLAG_ARRAY_POINTER 0x8093

#define GL_MATRIX_MODE 0x0BA0
#define GL_MODELVIEW 0x1700
#define GL_PROJECTION 0x1701
#define GL_TEXTURE 0x1702

#define GL_POINT_SMOOTH 0x0B10
#define GL_POINT_SIZE 0x0B11
#define GL_POINT_SIZE_RANGE 0x0B12
#define GL_POINT_SIZE_GRANULARITY 0x0B13

#define GL_LINE_SMOOTH 0x0B20
#define GL_LINE_WIDTH 0x0B21
#define GL_LINE_WIDTH_RANGE 0x0B22
#define GL_LINE_WIDTH_GRANULARITY 0x0B23
#define GL_LINE_STIPPLE 0x0B24
#define GL_LINE_STIPPLE_PATTERN 0x0B25
#define GL_LINE_STIPPLE_REPEAT 0x0B26

#define GL_FRONT 0x0404
#define GL_BACK 0x0405
#define GL_CW 0x0900
#define GL_CCW 0x0901
#define GL_POLYGON_MODE 0x0B40
#define GL_POLYGON_SMOOTH 0x0B41
#define GL_POLYGON_STIPPLE 0x0B42
#define GL_EDGE_FLAG 0x0B43
#define GL_CULL_FACE 0x0B44
#define GL_CULL_FACE_MODE 0x0B45
#define GL_FRONT_FACE 0x0B46
#define GL_POINT 0x1B00
#define GL_LINE 0x1B01
#define GL_FILL 0x1B02
#define GL_POLYGON_OFFSET_UNITS 0x2A00
#define GL_POLYGON_OFFSET_POINT 0x2A01
#define GL_POLYGON_OFFSET_LINE 0x2A02
#define GL_POLYGON_OFFSET_FILL 0x8037
#define GL_POLYGON_OFFSET_FACTOR 0x8038

#define GL_LIST_MODE 0x0B30
#define GL_LIST_BASE 0x0B32
#define GL_LIST_INDEX 0x0B33
#define GL_COMPILE 0x1300
#define GL_COMPILE_AND_EXECUTE 0x1301

#define GL_NEVER 0x0200
#define GL_LESS 0x0201
#define GL_EQUAL 0x0202
#define GL_LEQUAL 0x0203
#define GL_GREATER 0x0204
#define GL_NOTEQUAL 0x0205
#define GL_GEQUAL 0x0206
#define GL_ALWAYS 0x0207
#define GL_DEPTH_RANGE 0x0B70
#define GL_DEPTH_TEST 0x0B71
#define GL_DEPTH_WRITEMASK 0x0B72
#define GL_DEPTH_CLEAR_VALUE 0x0B73
#define GL_DEPTH_FUNC 0x0B74
#define GL_DEPTH_BITS 0x0D56
#define GL_DEPTH_COMPONENT 0x1902

#define GL_FRONT_AND_BACK 0x0408
#define GL_LIGHTING 0x0B50
#define GL_LIGHT_MODEL_LOCAL_VIEWER 0x0B51
#define GL_LIGHT_MODEL_TWO_SIDE 0x0B52

```

```

#define GL_LIGHT_MODEL_AMBIENT 0x0B53
#define GL_SHADE_MODEL 0x0B54
#define GL_COLOR_MATERIAL_FACE 0x0B55
#define GL_COLOR_MATERIAL_PARAMETER 0x0B56
#define GL_COLOR_MATERIAL 0x0B57
#define GL_NORMALIZE 0x0BA1
#define GL_AMBIENT 0x1200
#define GL_DIFFUSE 0x1201
#define GL_SPECULAR 0x1202
#define GL_POSITION 0x1203
#define GL_SPOT_DIRECTION 0x1204
#define GL_SPOT_EXPONENT 0x1205
#define GL_SPOT_CUTOFF 0x1206
#define GL_CONSTANT_ATTENUATION 0x1207
#define GL_LINEAR_ATTENUATION 0x1208
#define GL_QUADRATIC_ATTENUATION 0x1209
#define GL_EMISSION 0x1600
#define GL_SHININESS 0x1601
#define GL_AMBIENT_AND_DIFFUSE 0x1602
#define GL_COLOR_INDEXES 0x1603
#define GL_FLAT 0x1D00
#define GL_SMOOTH 0x1D01
#define GL_LIGHT0 0x4000
#define GL_LIGHT1 0x4001
#define GL_LIGHT2 0x4002
#define GL_LIGHT3 0x4003
#define GL_LIGHT4 0x4004
#define GL_LIGHT5 0x4005
#define GL_LIGHT6 0x4006
#define GL_LIGHT7 0x4007

#define GL_CLIP_PLANE0 0x3000
#define GL_CLIP_PLANE1 0x3001
#define GL_CLIP_PLANE2 0x3002
#define GL_CLIP_PLANE3 0x3003
#define GL_CLIP_PLANE4 0x3004
#define GL_CLIP_PLANE5 0x3005

#define GL_ACCUM 0x0100
#define GL_LOAD 0x0101
#define GL_RETURN 0x0102
#define GL_MULT 0x0103
#define GL_ADD 0x0104
#define GL_ACCUM_CLEAR_VALUE 0x0B80
#define GL_ACCUM_RED_BITS 0x0D58
#define GL_ACCUM_GREEN_BITS 0x0D59
#define GL_ACCUM_BLUE_BITS 0x0D5A
#define GL_ACCUM_ALPHA_BITS 0x0D5B

#define GL_ALPHA_TEST 0x0BC0
#define GL_ALPHA_TEST_FUNC 0x0BC1
#define GL_ALPHA_TEST_REF 0x0BC2

#define GL_ZERO 0x0
#define GL_SRC_COLOR 0x0300
#define GL_ONE_MINUS_SRC_COLOR 0x0301
#define GL_SRC_ALPHA 0x0302
#define GL_ONE_MINUS_SRC_ALPHA 0x0303
#define GL_DST_ALPHA 0x0304
#define GL_ONE_MINUS_DST_ALPHA 0x0305
#define GL_DST_COLOR 0x0306
#define GL_ONE_MINUS_DST_COLOR 0x0307
#define GL_SRC_ALPHA_SATURATE 0x0308
#define GL_BLEND_DST 0x0BE0
#define GL_BLEND_SRC 0x0BE1
#define GL_BLEND 0x0BE2

```



```

#define GL_ONE 0x1

#define GL_RENDER 0x1C00
#define GL_FEEDBACK 0x1C01
#define GL_SELECT 0x1C02

#define GL_2D 0x0600
#define GL_3D 0x0601
#define GL_3D_COLOR 0x0602
#define GL_3D_COLOR_TEXTURE 0x0603
#define GL_4D_COLOR_TEXTURE 0x0604
#define GL_PASS_THROUGH_TOKEN 0x0700
#define GL_POINT_TOKEN 0x0701
#define GL_LINE_TOKEN 0x0702
#define GL_POLYGON_TOKEN 0x0703
#define GL_BITMAP_TOKEN 0x0704
#define GL_DRAW_PIXEL_TOKEN 0x0705
#define GL_COPY_PIXEL_TOKEN 0x0706
#define GL_LINE_RESET_TOKEN 0x0707
#define GL_FEEDBACK_BUFFER_POINTER 0x0DF0
#define GL_FEEDBACK_BUFFER_SIZE 0x0DF1
#define GL_FEEDBACK_BUFFER_TYPE 0x0DF2
#define GL_ALPHA_BLEND_EQUATION_ATI 0x883D
#define GL_ATI_blend_equation_separate 1

#define GL_CURRENT_BIT 0x00000001
#define GL_POINT_BIT 0x00000002
#define GL_LINE_BIT 0x00000004
#define GL_POLYGON_BIT 0x00000008
#define GL_POLYGON_STIPPLE_BIT 0x00000010
#define GL_PIXEL_MODE_BIT 0x00000020
#define GL_LIGHTING_BIT 0x00000040
#define GL_FOG_BIT 0x00000080
#define GL_DEPTH_BUFFER_BIT 0x00000100
#define GL_ACCUM_BUFFER_BIT 0x00000200
#define GL_STENCIL_BUFFER_BIT 0x00000400
#define GL_VIEWPORT_BIT 0x00000800
#define GL_TRANSFORM_BIT 0x00001000
#define GL_ENABLE_BIT 0x00002000
#define GL_COLOR_BUFFER_BIT 0x00004000
#define GL_HINT_BIT 0x00008000
#define GL_EVAL_BIT 0x00010000
#define GL_LIST_BIT 0x00020000
#define GL_TEXTURE_BIT 0x00040000
#define GL_SCISSOR_BIT 0x00080000
#define GL_ALL_ATTRIB_BITS 0x000FFFFF

#define GL_CLIENT_PIXEL_STORE_BIT 0x00000001
#define GL_CLIENT_VERTEX_ARRAY_BIT 0x00000002
#define GL_TEXTURE_INTERNAL_FORMAT 0x1003
#define GL_R3_G3_B2 0x2A10
#define GL_ALPHA4 0x803B
#define GL_ALPHA8 0x803C
#define GL_ALPHA12 0x803D
#define GL_ALPHA16 0x803E
#define GL_LUMINANCE4 0x803F
#define GL_LUMINANCE8 0x8040
#define GL_LUMINANCE12 0x8041
#define GL_LUMINANCE16 0x8042
#define GL_LUMINANCE4_ALPHA4 0x8043
#define GL_LUMINANCE6_ALPHA2 0x8044
#define GL_LUMINANCE8_ALPHA8 0x8045
#define GL_LUMINANCE12_ALPHA4 0x8046
#define GL_LUMINANCE12_ALPHA12 0x8047
#define GL_LUMINANCE16_ALPHA16 0x8048
#define GL_INTENSITY 0x8049

```



```

#define GL_INTENSITY4      0x804A
#define GL_INTENSITY8      0x804B
#define GL_INTENSITY12     0x804C
#define GL_INTENSITY16     0x804D
#define GL_RGB4             0x804F
#define GL_RGB5             0x8050
#define GL_RGB8             0x8051
#define GL_RGB10            0x8052
#define GL_RGB12            0x8053
#define GL_RGB16            0x8054
#define GL_RGBA2            0x8055
#define GL_RGBA4            0x8056
#define GL_RGB5_A1          0x8057
#define GL_RGBA8            0x8058
#define GL_RGB10_A2         0x8059
#define GL_RGBA12           0x805A
#define GL_RGBA16           0x805B
#define GL_PROXY_TEXTURE_1D 0x8063
#define GL_PROXY_TEXTURE_2D 0x8064
#define GL_TEXTURE_PRIORITY 0x8066
#define GL_TEXTURE_RESIDENT 0x8067
#define GL_TEXTURE_BINDING_1D 0x8068
#define GL_TEXTURE_BINDING_2D 0x8069
#define GL_ALL_CLIENT_ATTRIB_BITS 0xFFFFFFFF
#define GL_CLIENT_ALL_ATTRIB_BITS 0xFFFFFFFF

#define GL_SMOOTH_POINT_SIZE_RANGE 0x0B12
#define GL_SMOOTH_POINT_SIZE_GRANULARITY 0x0B13
#define GL_SMOOTH_LINE_WIDTH_RANGE 0x0B22
#define GL_SMOOTH_LINE_WIDTH_GRANULARITY 0x0B23
#define GL_UNSIGNED_BYTE_3_3_2 0x8032
#define GL_UNSIGNED_SHORT_4_4_4_4 0x8033
#define GL_UNSIGNED_SHORT_5_5_5_4 0x8034
#define GL_UNSIGNED_INT_8_8_8_8 0x8035
#define GL_UNSIGNED_INT_10_10_10_2 0x8036
#define GL_RESCALE_NORMAL 0x803A
#define GL_TEXTURE_BINDING_3D 0x806A
#define GL_PACK_SKIP_IMAGES 0x806B
#define GL_PACK_IMAGE_HEIGHT 0x806C
#define GL_UNPACK_SKIP_IMAGES 0x806D
#define GL_UNPACK_IMAGE_HEIGHT 0x806E
#define GL_TEXTURE_3D 0x806F
#define GL_PROXY_TEXTURE_3D 0x8070
#define GL_TEXTURE_DEPTH 0x8071
#define GL_TEXTURE_WRAP_R 0x8072
#define GL_MAX_3D_TEXTURE_SIZE 0x8073
#define GL_BGR 0x80E0
#define GL_BGRA 0x80E1
#define GL_MAX_ELEMENTS_VERTICES 0x80E8
#define GL_MAX_ELEMENTS_INDICES 0x80E9
#define GL_CLAMP_TO_EDGE 0x812F
#define GL_TEXTURE_MIN_LOD 0x813A
#define GL_TEXTURE_MAX_LOD 0x813B
#define GL_TEXTURE_BASE_LEVEL 0x813C
#define GL_TEXTURE_MAX_LEVEL 0x813D
#define GL_LIGHT_MODEL_COLOR_CONTROL 0x81F8
#define GL_SINGLE_COLOR 0x81F9
#define GL_SEPARATE_SPECULAR_COLOR 0x81FA
#define GL_UNSIGNED_BYTE_2_3_3_REV 0x8362
#define GL_UNSIGNED_SHORT_5_6_5 0x8363
#define GL_UNSIGNED_SHORT_5_6_5_REV 0x8364
#define GL_UNSIGNED_SHORT_4_4_4_4_REV 0x8365
#define GL_UNSIGNED_SHORT_1_5_5_5_REV 0x8366
#define GL_UNSIGNED_INT_8_8_8_8_REV 0x8367
#define GL_UNSIGNED_INT_2_10_10_10_REV 0x8368
#define GL_ALIASED_POINT_SIZE_RANGE 0x846D

```

```

#define GL_ALIASED_LINE_WIDTH_RANGE 0x846E

#define GL_CONSTANT_COLOR 0x8001
#define GL_ONE_MINUS_CONSTANT_COLOR 0x8002
#define GL_CONSTANT_ALPHA 0x8003
#define GL_ONE_MINUS_CONSTANT_ALPHA 0x8004
#define GL_BLEND_COLOR 0x8005
#define GL_FUNC_ADD 0x8006
#define GL_MIN 0x8007
#define GL_MAX 0x8008
#define GL_BLEND_EQUATION 0x8009
#define GL_FUNC_SUBTRACT 0x800A
#define GL_FUNC_REVERSE_SUBTRACT 0x800B
#define GL_CONVOLUTION_1D 0x8010
#define GL_CONVOLUTION_2D 0x8011
#define GL_SEPARABLE_2D 0x8012
#define GL_CONVOLUTION_BORDER_MODE 0x8013
#define GL_CONVOLUTION_FILTER_SCALE 0x8014
#define GL_CONVOLUTION_FILTER_BIAS 0x8015
#define GL_REDUCE 0x8016
#define GL_CONVOLUTION_FORMAT 0x8017
#define GL_CONVOLUTION_WIDTH 0x8018
#define GL_CONVOLUTION_HEIGHT 0x8019
#define GL_MAX_CONVOLUTION_WIDTH 0x801A
#define GL_MAX_CONVOLUTION_HEIGHT 0x801B
#define GL_POST_CONVOLUTION_RED_SCALE 0x801C
#define GL_POST_CONVOLUTION_GREEN_SCALE 0x801D
#define GL_POST_CONVOLUTION_BLUE_SCALE 0x801E
#define GL_POST_CONVOLUTION_ALPHA_SCALE 0x801F
#define GL_POST_CONVOLUTION_RED_BIAS 0x8020
#define GL_POST_CONVOLUTION_GREEN_BIAS 0x8021
#define GL_POST_CONVOLUTION_BLUE_BIAS 0x8022
#define GL_POST_CONVOLUTION_ALPHA_BIAS 0x8023
#define GL_HISTOGRAM 0x8024
#define GL_PROXY_HISTOGRAM 0x8025
#define GL_HISTOGRAM_WIDTH 0x8026
#define GL_HISTOGRAM_FORMAT 0x8027
#define GL_HISTOGRAM_RED_SIZE 0x8028
#define GL_HISTOGRAM_GREEN_SIZE 0x8029
#define GL_HISTOGRAM_BLUE_SIZE 0x802A
#define GL_HISTOGRAM_ALPHA_SIZE 0x802B
#define GL_HISTOGRAM_LUMINANCE_SIZE 0x802C
#define GL_HISTOGRAM_SINK 0x802D
#define GL_MINMAX 0x802E
#define GL_MINMAX_FORMAT 0x802F
#define GL_MINMAX_SINK 0x8030
#define GL_TABLE_TOO_LARGE 0x8031
#define GL_COLOR_MATRIX 0x80B1
#define GL_COLOR_MATRIX_STACK_DEPTH 0x80B2
#define GL_MAX_COLOR_MATRIX_STACK_DEPTH 0x80B3
#define GL_POST_COLOR_MATRIX_RED_SCALE 0x80B4
#define GL_POST_COLOR_MATRIX_GREEN_SCALE 0x80B5
#define GL_POST_COLOR_MATRIX_BLUE_SCALE 0x80B6
#define GL_POST_COLOR_MATRIX_ALPHA_SCALE 0x80B7
#define GL_POST_COLOR_MATRIX_RED_BIAS 0x80B8
#define GL_POST_COLOR_MATRIX_GREEN_BIAS 0x80B9
#define GL_POST_COLOR_MATRIX_BLUE_BIAS 0x80BA
#define GL_POST_COLOR_MATRIX_ALPHA_BIAS 0x80BB
#define GL_COLOR_TABLE 0x80D0
#define GL_POST_CONVOLUTION_COLOR_TABLE 0x80D1
#define GL_POST_COLOR_MATRIX_COLOR_TABLE 0x80D2
#define GL_PROXY_COLOR_TABLE 0x80D3
#define GL_PROXY_POST_CONVOLUTION_COLOR_TABLE 0x80D4
#define GL_PROXY_POST_COLOR_MATRIX_COLOR_TABLE 0x80D5
#define GL_COLOR_TABLE_SCALE 0x80D6
#define GL_COLOR_TABLE_BIAS 0x80D7

```

```

#define GL_COLOR_TABLE_FORMAT      0x80D8
#define GL_COLOR_TABLE_WIDTH       0x80D9
#define GL_COLOR_TABLE_RED_SIZE    0x80DA
#define GL_COLOR_TABLE_GREEN_SIZE  0x80DB
#define GL_COLOR_TABLE_BLUE_SIZE   0x80DC
#define GL_COLOR_TABLE_ALPHA_SIZE  0x80DD
#define GL_COLOR_TABLE_LUMINANCE_SIZE 0x80DE
#define GL_COLOR_TABLE_INTENSITY_SIZE 0x80DF
#define GL_CONSTANT_BORDER         0x8151
#define GL_REPLICATE_BORDER        0x8153
#define GL_CONVOLUTION_BORDER_COLOR 0x8154

#define GL_TEXTURE0                 0x84C0
#define GL_TEXTURE1                 0x84C1
#define GL_TEXTURE2                 0x84C2
#define GL_TEXTURE3                 0x84C3
#define GL_TEXTURE4                 0x84C4
#define GL_TEXTURE5                 0x84C5
#define GL_TEXTURE6                 0x84C6
#define GL_TEXTURE7                 0x84C7
#define GL_TEXTURE8                 0x84C8
#define GL_TEXTURE9                 0x84C9
#define GL_TEXTURE10                0x84CA
#define GL_TEXTURE11                0x84CB
#define GL_TEXTURE12                0x84CC
#define GL_TEXTURE13                0x84CD
#define GL_TEXTURE14                0x84CE
#define GL_TEXTURE15                0x84CF
#define GL_TEXTURE16                0x84D0
#define GL_TEXTURE17                0x84D1
#define GL_TEXTURE18                0x84D2
#define GL_TEXTURE19                0x84D3
#define GL_TEXTURE20                0x84D4
#define GL_TEXTURE21                0x84D5
#define GL_TEXTURE22                0x84D6
#define GL_TEXTURE23                0x84D7
#define GL_TEXTURE24                0x84D8
#define GL_TEXTURE25                0x84D9
#define GL_TEXTURE26                0x84DA
#define GL_TEXTURE27                0x84DB
#define GL_TEXTURE28                0x84DC
#define GL_TEXTURE29                0x84DD
#define GL_TEXTURE30                0x84DE
#define GL_TEXTURE31                0x84DF
#define GL_ACTIVE_TEXTURE           0x84E0
#define GL_CLIENT_ACTIVE_TEXTURE    0x84E1
#define GL_MAX_TEXTURE_UNITS        0x84E2

#define GL_NORMAL_MAP               0x8511
#define GL_REFLECTION_MAP           0x8512
#define GL_TEXTURE_CUBE_MAP         0x8513
#define GL_TEXTURE_BINDING_CUBE_MAP 0x8514
#define GL_TEXTURE_CUBE_MAP_POSITIVE_X 0x8515
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_X 0x8516
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Y 0x8517
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y 0x8518
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Z 0x8519
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z 0x851A
#define GL_PROXY_TEXTURE_CUBE_MAP   0x851B
#define GL_MAX_CUBE_MAP_TEXTURE_SIZE 0x851C

#define GL_COMPRESSED_ALPHA          0x84E9
#define GL_COMPRESSED_LUMINANCE     0x84EA
#define GL_COMPRESSED_LUMINANCE_ALPHA 0x84EB
#define GL_COMPRESSED_INTENSITY     0x84EC
#define GL_COMPRESSED_RGB           0x84ED

```

```

#define GL_COMPRESSED_RGBA          0x84EE
#define GL_TEXTURE_COMPRESSION_HINT 0x84EF
#define GL_TEXTURE_COMPRESSED_IMAGE_SIZE 0x86A0
#define GL_TEXTURE_COMPRESSED        0x86A1
#define GL_NUM_COMPRESSED_TEXTURE_FORMATS 0x86A2
#define GL_COMPRESSED_TEXTURE_FORMATS 0x86A3

#define GL_MULTISAMPLE_BIT          0x20000000
#define GL_MULTISAMPLE              0x809D
#define GL_SAMPLE_ALPHA_TO_COVERAGE 0x809E
#define GL_SAMPLE_ALPHA_TO_ONE     0x809F
#define GL_SAMPLE_COVERAGE         0x80A0
#define GL_SAMPLE_BUFFERS          0x80A8
#define GL_SAMPLES                  0x80A9
#define GL_SAMPLE_COVERAGE_VALUE    0x80AA
#define GL_SAMPLE_COVERAGE_INVERT   0x80AB

#define GL_TRANSPOSE_MODELVIEW_MATRIX 0x84E3
#define GL_TRANSPOSE_PROJECTION_MATRIX 0x84E4
#define GL_TRANSPOSE_TEXTURE_MATRIX   0x84E5
#define GL_TRANSPOSE_COLOR_MATRIX     0x84E6

#define GL_SUBTRACT                 0x84E7
#define GL_COMBINE                  0x8570
#define GL_COMBINE_RGB              0x8571
#define GL_COMBINE_ALPHA            0x8572
#define GL_RGB_SCALE                 0x8573
#define GL_ADD_SIGNED                0x8574
#define GL_INTERPOLATE               0x8575
#define GL_CONSTANT                  0x8576
#define GL_PRIMARY_COLOR             0x8577
#define GL_PREVIOUS                  0x8578
#define GL_SOURCE0_RGB               0x8580
#define GL_SOURCE1_RGB               0x8581
#define GL_SOURCE2_RGB               0x8582
#define GL_SOURCE0_ALPHA             0x8588
#define GL_SOURCE1_ALPHA             0x8589
#define GL_SOURCE2_ALPHA             0x858A
#define GL_OPERAND0_RGB              0x8590
#define GL_OPERAND1_RGB              0x8591
#define GL_OPERAND2_RGB              0x8592
#define GL_OPERAND0_ALPHA            0x8598
#define GL_OPERAND1_ALPHA            0x8599
#define GL_OPERAND2_ALPHA            0x859A

#define GL_DOT3_RGB                 0x86AE
#define GL_DOT3_RGBA                 0x86AF

#define GL_CLAMP_TO_BORDER           0x812D

#define GL_TEXTURE0_ARB              0x84C0
#define GL_TEXTURE1_ARB              0x84C1
#define GL_TEXTURE2_ARB              0x84C2
#define GL_TEXTURE3_ARB              0x84C3
#define GL_TEXTURE4_ARB              0x84C4
#define GL_TEXTURE5_ARB              0x84C5
#define GL_TEXTURE6_ARB              0x84C6
#define GL_TEXTURE7_ARB              0x84C7
#define GL_TEXTURE8_ARB              0x84C8
#define GL_TEXTURE9_ARB              0x84C9
#define GL_TEXTURE10_ARB             0x84CA
#define GL_TEXTURE11_ARB             0x84CB
#define GL_TEXTURE12_ARB             0x84CC
#define GL_TEXTURE13_ARB             0x84CD
#define GL_TEXTURE14_ARB             0x84CE
#define GL_TEXTURE15_ARB             0x84CF

```

```

#define GL_TEXTURE16_ARB      0x84D0
#define GL_TEXTURE17_ARB      0x84D1
#define GL_TEXTURE18_ARB      0x84D2
#define GL_TEXTURE19_ARB      0x84D3
#define GL_TEXTURE20_ARB      0x84D4
#define GL_TEXTURE21_ARB      0x84D5
#define GL_TEXTURE22_ARB      0x84D6
#define GL_TEXTURE23_ARB      0x84D7
#define GL_TEXTURE24_ARB      0x84D8
#define GL_TEXTURE25_ARB      0x84D9
#define GL_TEXTURE26_ARB      0x84DA
#define GL_TEXTURE27_ARB      0x84DB
#define GL_TEXTURE28_ARB      0x84DC
#define GL_TEXTURE29_ARB      0x84DD
#define GL_TEXTURE30_ARB      0x84DE
#define GL_TEXTURE31_ARB      0x84DF
#define GL_ACTIVE_TEXTURE_ARB  0x84E0
#define GL_CLIENT_ACTIVE_TEXTURE_ARB 0x84E1
#define GL_MAX_TEXTURE_UNITS_ARB 0x84E2
#define GL_ARB_imaging 1
#ifndef GL_ARB_multitexture
#define GL_ARB_multitexture 1
#endif

#define GL_NO_ERROR           0x0
#define GL_INVALID_ENUM      0x0500
#define GL_INVALID_VALUE     0x0501
#define GL_INVALID_OPERATION 0x0502
#define GL_STACK_OVERFLOW    0x0503
#define GL_STACK_UNDERFLOW   0x0504
#define GL_OUT_OF_MEMORY     0x0505

#define GL_VENDOR             0x1F00
#define GL_RENDERER           0x1F01
#define GL_VERSION            0x1F02
#define GL_EXTENSIONS         0x1F03

#define GL_SELECTION_BUFFER_POINTER 0x0DF3
#define GL_SELECTION_BUFFER_SIZE    0x0DF4

#define GL_EXP 0x0800
#define GL_EXP2 0x0801
#define GL_FOG 0x0B60
#define GL_FOG_INDEX 0x0B61
#define GL_FOG_DENSITY 0x0B62
#define GL_FOG_START 0x0B63
#define GL_FOG_END 0x0B64
#define GL_FOG_MODE 0x0B65
#define GL_FOG_COLOR 0x0B66
#define GL_LINEAR 0x2601

#define GL_LOGIC_OP_MODE 0x0BF0
#define GL_INDEX_LOGIC_OP 0x0BF1
#define GL_LOGIC_OP 0x0BF1
#define GL_COLOR_LOGIC_OP 0x0BF2
#define GL_CLEAR 0x1500
#define GL_AND 0x1501
#define GL_AND_REVERSE 0x1502
#define GL_COPY 0x1503
#define GL_AND_INVERTED 0x1504
#define GL_NOOP 0x1505
#define GL_XOR 0x1506
#define GL_OR 0x1507
#define GL_NOR 0x1508
#define GL_EQUIV 0x1509
#define GL_INVERT 0x150A

```

```

#define GL_OR_REVERSE 0x150B
#define GL_COPY_INVERTED 0x150C
#define GL_OR_INVERTED 0x150D
#define GL_NAND 0x150E
#define GL_SET 0x150F

#define GL_STENCIL_TEST 0x0B90
#define GL_STENCIL_CLEAR_VALUE 0x0B91
#define GL_STENCIL_FUNC 0x0B92
#define GL_STENCIL_VALUE_MASK 0x0B93
#define GL_STENCIL_FAIL 0x0B94
#define GL_STENCIL_PASS_DEPTH_FAIL 0x0B95
#define GL_STENCIL_PASS_DEPTH_PASS 0x0B96
#define GL_STENCIL_REF 0x0B97
#define GL_STENCIL_WRITEMASK 0x0B98
#define GL_STENCIL_BITS 0x0D57
#define GL_STENCIL_INDEX 0x1901
#define GL_KEEP 0x1E00
#define GL_REPLACE 0x1E01
#define GL_INCR 0x1E02
#define GL_DECR 0x1E03

#define GL_NONE 0x0
#define GL_FRONT_LEFT 0x0400
#define GL_FRONT_RIGHT 0x0401
#define GL_BACK_LEFT 0x0402
#define GL_BACK_RIGHT 0x0403
#define GL_LEFT 0x0406
#define GL_RIGHT 0x0407
#define GL_AUX0 0x0409
#define GL_AUX1 0x040A
#define GL_AUX2 0x040B
#define GL_AUX3 0x040C
#define GL_DITHER 0x0BD0
#define GL_AUX_BUFFERS 0x0C00
#define GL_DRAW_BUFFER 0x0C01
#define GL_READ_BUFFER 0x0C02
#define GL_DOUBLEBUFFER 0x0C32
#define GL_STEREO 0x0C33
#define GL_SUBPIXEL_BITS 0x0D50
#define GL_INDEX_BITS 0x0D51
#define GL_RED_BITS 0x0D52
#define GL_GREEN_BITS 0x0D53
#define GL_BLUE_BITS 0x0D54
#define GL_ALPHA_BITS 0x0D55
#define GL_COLOR 0x1800
#define GL_DEPTH 0x1801
#define GL_STENCIL 0x1802
#define GL_COLOR_INDEX 0x1900
#define GL_RED 0x1903
#define GL_GREEN 0x1904
#define GL_BLUE 0x1905
#define GL_ALPHA 0x1906
#define GL_RGB 0x1907
#define GL_RGBA 0x1908
#define GL_LUMINANCE 0x1909
#define GL_LUMINANCE_ALPHA 0x190A
#define GL_BITMAP 0x1A00

#define GL_MAX_LIST_NESTING 0x0B31
#define GL_MAX_EVAL_ORDER 0x0D30
#define GL_MAX_LIGHTS 0x0D31
#define GL_MAX_CLIP_PLANES 0x0D32
#define GL_MAX_TEXTURE_SIZE 0x0D33
#define GL_MAX_PIXEL_MAP_TABLE 0x0D34
#define GL_MAX_ATTRIB_STACK_DEPTH 0x0D35

```

```

#define GL_MAX_MODELVIEW_STACK_DEPTH 0x0D36
#define GL_MAX_NAME_STACK_DEPTH 0x0D37
#define GL_MAX_PROJECTION_STACK_DEPTH 0x0D38
#define GL_MAX_TEXTURE_STACK_DEPTH 0x0D39
#define GL_MAX_VIEWPORT_DIMS 0x0D3A
#define GL_MAX_CLIENT_ATTRIB_STACK_DEPTH 0x0D3B

#define GL_CURRENT_COLOR 0x0B00
#define GL_CURRENT_INDEX 0x0B01
#define GL_CURRENT_NORMAL 0x0B02
#define GL_CURRENT_TEXTURE_COORDS 0x0B03
#define GL_CURRENT_RASTER_COLOR 0x0B04
#define GL_CURRENT_RASTER_INDEX 0x0B05
#define GL_CURRENT_RASTER_TEXTURE_COORDS 0x0B06
#define GL_CURRENT_RASTER_POSITION 0x0B07
#define GL_CURRENT_RASTER_POSITION_VALID 0x0B08
#define GL_CURRENT_RASTER_DISTANCE 0x0B09
#define GL_VIEWPORT 0x0BA2
#define GL_MODELVIEW_STACK_DEPTH 0x0BA3
#define GL_PROJECTION_STACK_DEPTH 0x0BA4
#define GL_TEXTURE_STACK_DEPTH 0x0BA5
#define GL_MODELVIEW_MATRIX 0x0BA6
#define GL_PROJECTION_MATRIX 0x0BA7
#define GL_TEXTURE_MATRIX 0x0BA8
#define GL_ATTRIB_STACK_DEPTH 0x0BB0
#define GL_CLIENT_ATTRIB_STACK_DEPTH 0x0BB1
#define GL_INDEX_CLEAR_VALUE 0x0C20
#define GL_INDEX_WRITEMASK 0x0C21
#define GL_COLOR_CLEAR_VALUE 0x0C22
#define GL_COLOR_WRITEMASK 0x0C23
#define GL_INDEX_MODE 0x0C30
#define GL_RGBA_MODE 0x0C31
#define GL_RENDER_MODE 0x0C40
#define GL_NAME_STACK_DEPTH 0x0D70

#define GL_COEFF 0x0A00
#define GL_ORDER 0x0A01
#define GL_DOMAIN 0x0A02
#define GL_AUTO_NORMAL 0x0D80
#define GL_MAP1_COLOR_4 0x0D90
#define GL_MAP1_INDEX 0x0D91
#define GL_MAP1_NORMAL 0x0D92
#define GL_MAP1_TEXTURE_COORD_1 0x0D93
#define GL_MAP1_TEXTURE_COORD_2 0x0D94
#define GL_MAP1_TEXTURE_COORD_3 0x0D95
#define GL_MAP1_TEXTURE_COORD_4 0x0D96
#define GL_MAP1_VERTEX_3 0x0D97
#define GL_MAP1_VERTEX_4 0x0D98
#define GL_MAP2_COLOR_4 0x0DB0
#define GL_MAP2_INDEX 0x0DB1
#define GL_MAP2_NORMAL 0x0DB2
#define GL_MAP2_TEXTURE_COORD_1 0x0DB3
#define GL_MAP2_TEXTURE_COORD_2 0x0DB4
#define GL_MAP2_TEXTURE_COORD_3 0x0DB5
#define GL_MAP2_TEXTURE_COORD_4 0x0DB6
#define GL_MAP2_VERTEX_3 0x0DB7
#define GL_MAP2_VERTEX_4 0x0DB8
#define GL_MAP1_GRID_DOMAIN 0x0DD0
#define GL_MAP1_GRID_SEGMENTS 0x0DD1
#define GL_MAP2_GRID_DOMAIN 0x0DD2
#define GL_MAP2_GRID_SEGMENTS 0x0DD3

#define GL_PERSPECTIVE_CORRECTION_HINT 0x0C50
#define GL_POINT_SMOOTH_HINT 0x0C51
#define GL_LINE_SMOOTH_HINT 0x0C52
#define GL_POLYGON_SMOOTH_HINT 0x0C53

```



```

#define GL_FOG_HINT 0x0C54
#define GL_DONT_CARE 0x1100
#define GL_FASTEST 0x1101
#define GL_NICEST 0x1102

#define GL_SCISSOR_BOX 0x0C10
#define GL_SCISSOR_TEST 0x0C11

#define GL_PIXEL_MAP_I_TO_I 0x0C70
#define GL_PIXEL_MAP_S_TO_S 0x0C71
#define GL_PIXEL_MAP_I_TO_R 0x0C72
#define GL_PIXEL_MAP_I_TO_G 0x0C73
#define GL_PIXEL_MAP_I_TO_B 0x0C74
#define GL_PIXEL_MAP_I_TO_A 0x0C75
#define GL_PIXEL_MAP_R_TO_R 0x0C76
#define GL_PIXEL_MAP_G_TO_G 0x0C77
#define GL_PIXEL_MAP_B_TO_B 0x0C78
#define GL_PIXEL_MAP_A_TO_A 0x0C79
#define GL_PIXEL_MAP_I_TO_I_SIZE 0x0CB0
#define GL_PIXEL_MAP_S_TO_S_SIZE 0x0CB1
#define GL_PIXEL_MAP_I_TO_R_SIZE 0x0CB2
#define GL_PIXEL_MAP_I_TO_G_SIZE 0x0CB3
#define GL_PIXEL_MAP_I_TO_B_SIZE 0x0CB4
#define GL_PIXEL_MAP_I_TO_A_SIZE 0x0CB5
#define GL_PIXEL_MAP_R_TO_R_SIZE 0x0CB6
#define GL_PIXEL_MAP_G_TO_G_SIZE 0x0CB7
#define GL_PIXEL_MAP_B_TO_B_SIZE 0x0CB8
#define GL_PIXEL_MAP_A_TO_A_SIZE 0x0CB9
#define GL_UNPACK_SWAP_BYTES 0x0CF0
#define GL_UNPACK_LSB_FIRST 0x0CF1
#define GL_UNPACK_ROW_LENGTH 0x0CF2
#define GL_UNPACK_SKIP_ROWS 0x0CF3
#define GL_UNPACK_SKIP_PIXELS 0x0CF4
#define GL_UNPACK_ALIGNMENT 0x0CF5
#define GL_PACK_SWAP_BYTES 0x0D00
#define GL_PACK_LSB_FIRST 0x0D01
#define GL_PACK_ROW_LENGTH 0x0D02
#define GL_PACK_SKIP_ROWS 0x0D03
#define GL_PACK_SKIP_PIXELS 0x0D04
#define GL_PACK_ALIGNMENT 0x0D05
#define GL_MAP_COLOR 0x0D10
#define GL_MAP_STENCIL 0x0D11
#define GL_INDEX_SHIFT 0x0D12
#define GL_INDEX_OFFSET 0x0D13
#define GL_RED_SCALE 0x0D14
#define GL_RED_BIAS 0x0D15
#define GL_ZOOM_X 0x0D16
#define GL_ZOOM_Y 0x0D17
#define GL_GREEN_SCALE 0x0D18
#define GL_GREEN_BIAS 0x0D19
#define GL_BLUE_SCALE 0x0D1A
#define GL_BLUE_BIAS 0x0D1B
#define GL_ALPHA_SCALE 0x0D1C
#define GL_ALPHA_BIAS 0x0D1D
#define GL_DEPTH_SCALE 0x0D1E
#define GL_DEPTH_BIAS 0x0D1F

#define GL_TEXTURE_GEN_S 0x0C60
#define GL_TEXTURE_GEN_T 0x0C61
#define GL_TEXTURE_GEN_R 0x0C62
#define GL_TEXTURE_GEN_Q 0x0C63
#define GL_TEXTURE_1D 0x0DE0
#define GL_TEXTURE_2D 0x0DE1
#define GL_TEXTURE_WIDTH 0x1000
#define GL_TEXTURE_HEIGHT 0x1001
#define GL_TEXTURE_COMPONENTS 0x1003

```



```

#define GL_TEXTURE_BORDER_COLOR 0x1004
#define GL_TEXTURE_BORDER      0x1005
#define GL_S      0x2000
#define GL_T      0x2001
#define GL_R      0x2002
#define GL_Q      0x2003
#define GL_MODULATE      0x2100
#define GL_DECAL      0x2101
#define GL_TEXTURE_ENV_MODE      0x2200
#define GL_TEXTURE_ENV_COLOR      0x2201
#define GL_TEXTURE_ENV      0x2300
#define GL_EYE_LINEAR      0x2400
#define GL_OBJECT_LINEAR      0x2401
#define GL_SPHERE_MAP      0x2402
#define GL_TEXTURE_GEN_MODE      0x2500
#define GL_OBJECT_PLANE      0x2501
#define GL_EYE_PLANE      0x2502
#define GL_NEAREST      0x2600
#define GL_NEAREST_MIPMAP_NEAREST      0x2700
#define GL_LINEAR_MIPMAP_NEAREST      0x2701
#define GL_NEAREST_MIPMAP_LINEAR      0x2702
#define GL_LINEAR_MIPMAP_LINEAR      0x2703
#define GL_TEXTURE_MAG_FILTER      0x2800
#define GL_TEXTURE_MIN_FILTER      0x2801
#define GL_TEXTURE_WRAP_S      0x2802
#define GL_TEXTURE_WRAP_T      0x2803
#define GL_CLAMP      0x2900
#define GL_REPEAT      0x2901
#define GL_TEXTURE_RED_SIZE      0x805C
#define GL_TEXTURE_GREEN_SIZE      0x805D
#define GL_TEXTURE_BLUE_SIZE      0x805E
#define GL_TEXTURE_ALPHA_SIZE      0x805F
#define GL_TEXTURE_LUMINANCE_SIZE      0x8060
#define GL_TEXTURE_INTENSITY_SIZE      0x8061

#define GL_FALSE      0x0
#define GL_TRUE      0x1

extern void glAccum(GLenum op, GLfloat value);
extern void glActiveTexture(GLenum texture);
extern void glActiveTextureARB(GLenum texture);
extern void glAlphaFunc(GLenum func, GLclampf ref);
extern GLboolean glAreTexturesResident(GLsizei n, const GLuint *
textures,
                                GLboolean * residences);
extern void glArrayElement(GLint i);
extern void glBegin(GLenum mode);
extern void glBindTexture(GLenum target, GLuint texture);
extern void glBitmap(GLsizei width, GLsizei height, GLfloat xorig,
GLfloat yorig, GLfloat xmove, GLfloat ymove,
const GLubyte * bitmap);
extern void glBlendColor(GLclampf red, GLclampf green, GLclampf
blue,
                                GLclampf alpha);
extern void glBlendEquation(GLenum mode);
extern void glBlendFunc(GLenum sfactor, GLenum dfactor);
extern void glCallList(GLuint list);
extern void glCallLists(GLsizei n, GLenum type, const GLvoid *
lists);
extern void glClear(GLbitfield mask);
extern void glClearAccum(GLfloat red, GLfloat green, GLfloat blue,
GLfloat alpha);
extern void glClearColor(GLclampf red, GLclampf green, GLclampf
blue,
                                GLclampf alpha);
extern void glClearDepth(GLclampd depth);

```

```

extern void glClearIndex(GLfloat c);
extern void glClearStencil(GLint s);
extern void glClientActiveTexture(GLenum texture);
extern void glClientActiveTextureARB(GLenum texture);
extern void glClipPlane(GLenum plane, const GLdouble * equation);
extern void glColor3b(GLbyte red, GLbyte green, GLbyte blue);
extern void glColor3bv(const GLbyte * v);
extern void glColor3d(GLdouble red, GLdouble green, GLdouble blue);
extern void glColor3dv(const GLdouble * v);
extern void glColor3f(GLfloat red, GLfloat green, GLfloat blue);
extern void glColor3fv(const GLfloat * v);
extern void glColor3i(GLint red, GLint green, GLint blue);
extern void glColor3iv(const GLint * v);
extern void glColor3s(GLshort red, GLshort green, GLshort blue);
extern void glColor3sv(const GLshort * v);
extern void glColor3ub(GLubyte red, GLubyte green, GLubyte blue);
extern void glColor3ubv(const GLubyte * v);
extern void glColor3ui(GLuint red, GLuint green, GLuint blue);
extern void glColor3uiv(const GLuint * v);
extern void glColor3us(GLushort red, GLushort green, GLushort blue);
extern void glColor3usv(const GLushort * v);
extern void glColor4b(GLbyte red, GLbyte green, GLbyte blue, GLbyte
alpha);
extern void glColor4bv(const GLbyte * v);
extern void glColor4d(GLdouble red, GLdouble green, GLdouble blue,
GLdouble alpha);
extern void glColor4dv(const GLdouble * v);
extern void glColor4f(GLfloat red, GLfloat green, GLfloat blue,
GLfloat alpha);
extern void glColor4fv(const GLfloat * v);
extern void glColor4i(GLint red, GLint green, GLint blue, GLint
alpha);
extern void glColor4iv(const GLint * v);
extern void glColor4s(GLshort red, GLshort green, GLshort blue,
GLshort alpha);
extern void glColor4sv(const GLshort * v);
extern void glColor4ub(GLubyte red, GLubyte green, GLubyte blue,
GLubyte alpha);
extern void glColor4ubv(const GLubyte * v);
extern void glColor4ui(GLuint red, GLuint green, GLuint blue,
GLuint alpha);
extern void glColor4uiv(const GLuint * v);
extern void glColor4us(GLushort red, GLushort green, GLushort blue,
GLushort alpha);
extern void glColor4usv(const GLushort * v);
extern void glColorMask(GLboolean red, GLboolean green, GLboolean
blue,
GLboolean alpha);
extern void glColorMaterial(GLenum face, GLenum mode);
extern void glColorPointer(GLint size, GLenum type, GLsizei stride,
const GLvoid * pointer);
extern void glColorSubTable(GLenum target, GLsizei start, GLsizei
count,
GLenum format, GLenum type,
const GLvoid * data);
extern void glColorTable(GLenum target, GLenum internalformat,
GLsizei width, GLenum format, GLenum type,
const GLvoid * table);
extern void glColorTableParameterfv(GLenum target, GLenum pname,
const GLfloat * params);
extern void glColorTableParameteriv(GLenum target, GLenum pname,
const GLint * params);
extern void glCompressedTexImage1D(GLenum target, GLint level,
GLenum internalformat, GLsizei width,
GLint border, GLsizei imageSize,
const GLvoid * data);

```

```

extern void glCompressedTexImage2D(GLenum target, GLint level,
                                   GLenum internalformat, GLsizei width,
                                   GLsizei height, GLint border,
                                   GLsizei imageSize, const GLvoid *
data);
extern void glCompressedTexImage3D(GLenum target, GLint level,
                                   GLenum internalformat, GLsizei width,
                                   GLsizei height, GLsizei depth,
                                   GLint border, GLsizei imageSize,
                                   const GLvoid * data);
extern void glCompressedTexSubImage1D(GLenum target, GLint level,
                                       GLint xoffset, GLsizei width,
                                       GLenum format, GLsizei imageSize,
                                       const GLvoid * data);
extern void glCompressedTexSubImage2D(GLenum target, GLint level,
                                       GLint xoffset, GLint yoffset,
                                       GLsizei width, GLsizei height,
                                       GLenum format, GLsizei imageSize,
                                       const GLvoid * data);
extern void glCompressedTexSubImage3D(GLenum target, GLint level,
                                       GLint xoffset, GLint yoffset,
                                       GLint zoffset, GLsizei width,
                                       GLsizei height, GLsizei depth,
                                       GLenum format, GLsizei imageSize,
                                       const GLvoid * data);
extern void glConvolutionFilter1D(GLenum target, GLenum
internalformat,
                                   GLsizei width, GLenum format,
                                   GLenum type, const GLvoid * image);
extern void glConvolutionFilter2D(GLenum target, GLenum
internalformat,
                                   GLsizei width, GLsizei height,
                                   GLenum format, GLenum type,
                                   const GLvoid * image);
extern void glConvolutionParameterf(GLenum target, GLenum pname,
                                     GLfloat params);
extern void glConvolutionParameterfv(GLenum target, GLenum pname,
                                     const GLfloat * params);
extern void glConvolutionParameteri(GLenum target, GLenum pname,
                                     GLint params);
extern void glConvolutionParameteriv(GLenum target, GLenum pname,
                                     const GLint * params);
extern void glCopyColorSubTable(GLenum target, GLsizei start, GLint
x,
                                   GLint y, GLsizei width);
extern void glCopyColorTable(GLenum target, GLenum internalformat,
GLint x,
                                   GLint y, GLsizei width);
extern void glCopyConvolutionFilter1D(GLenum target, GLenum
internalformat,
                                   GLint x, GLint y, GLsizei width);
extern void glCopyConvolutionFilter2D(GLenum target, GLenum
internalformat,
                                   GLint x, GLint y, GLsizei width,
                                   GLsizei height);
extern void glCopyPixels(GLint x, GLint y, GLsizei width, GLsizei
height,
                                   GLenum type);
extern void glCopyTexImage1D(GLenum target, GLint level,
                             GLenum internalFormat, GLint x, GLint y,
                             GLsizei width, GLint border);
extern void glCopyTexImage2D(GLenum target, GLint level,
                             GLenum internalFormat, GLint x, GLint y,
                             GLsizei width, GLsizei height, GLint
border);

```

```

extern void glCopyTexSubImage1D(GLenum target, GLint level, GLint
xoffset,
                                GLint x, GLint y, GLsizei width);
extern void glCopyTexSubImage2D(GLenum target, GLint level, GLint
xoffset,
                                GLint yoffset, GLint x, GLint y,
                                GLsizei width, GLsizei height);
extern void glCopyTexSubImage3D(GLenum target, GLint level, GLint
xoffset,
                                GLint yoffset, GLint zoffset, GLint x,
                                GLint y, GLsizei width, GLsizei height);
extern void glCullFace(GLenum mode);
extern void glDeleteLists(GLuint list, GLsizei range);
extern void glDeleteTextures(GLsizei n, const GLuint * textures);
extern void glDepthFunc(GLenum func);
extern void glDepthMask(GLboolean flag);
extern void glDepthRange(GLclampd zNear, GLclampd zFar);
extern void glDisable(GLenum cap);
extern void glDisableClientState(GLenum array);
extern void glDrawArrays(GLenum mode, GLint first, GLsizei count);
extern void glDrawBuffer(GLenum mode);
extern void glDrawElements(GLenum mode, GLsizei count, GLenum type,
                            const GLvoid * indices);
extern void glDrawPixels(GLsizei width, GLsizei height, GLenum
format,
                            GLenum type, const GLvoid * pixels);
extern void glDrawRangeElements(GLenum mode, GLuint start, GLuint
end,
                            GLsizei count, GLenum type,
                            const GLvoid * indices);
extern void glEdgeFlag(GLboolean flag);
extern void glEdgeFlagPointer(GLsizei stride, const GLvoid *
pointer);
extern void glEdgeFlagv(const GLboolean * flag);
extern void glEnable(GLenum cap);
extern void glEnableClientState(GLenum array);
extern void glEnd(void);
extern void glEndList(void);
extern void glEvalCoord1d(GLdouble u);
extern void glEvalCoord1dv(const GLdouble * u);
extern void glEvalCoord1f(GLfloat u);
extern void glEvalCoord1fv(const GLfloat * u);
extern void glEvalCoord2d(GLdouble u, GLdouble v);
extern void glEvalCoord2dv(const GLdouble * u);
extern void glEvalCoord2f(GLfloat u, GLfloat v);
extern void glEvalCoord2fv(const GLfloat * u);
extern void glEvalMesh1(GLenum mode, GLint i1, GLint i2);
extern void glEvalMesh2(GLenum mode, GLint i1, GLint i2, GLint j1,
                        GLint j2);
extern void glEvalPoint1(GLint i);
extern void glEvalPoint2(GLint i, GLint j);
extern void glFeedbackBuffer(GLsizei size, GLenum type, GLfloat *
buffer);
extern void glFinish(void);
extern void glFlush(void);
extern void glFogf(GLenum pname, GLfloat param);
extern void glFogfv(GLenum pname, const GLfloat * params);
extern void glFogi(GLenum pname, GLint param);
extern void glFogiv(GLenum pname, const GLint * params);
extern void glFrontFace(GLenum mode);
extern void glFrustum(GLdouble left, GLdouble right, GLdouble
bottom,
                        GLdouble top, GLdouble zNear, GLdouble zFar);
extern GLuint glGenLists(GLsizei range);
extern void glGenTextures(GLsizei n, GLuint * textures);
extern void glGetBooleanv(GLenum pname, GLboolean * params);

```

```

extern void glGetClipPlane(GLenum plane, GLdouble * equation);
extern void glGetColorTable(GLenum target, GLenum format, GLenum
type,
        GLvoid * table);
extern void glGetColorTableParameterfv(GLenum target, GLenum pname,
        GLfloat * params);
extern void glGetColorTableParameteriv(GLenum target, GLenum pname,
        GLint * params);
extern void glGetCompressedTexImage(GLenum target, GLint lod,
        GLvoid * img);
extern void glGetConvolutionFilter(GLenum target, GLenum format,
        GLenum type, GLvoid * image);
extern void glGetConvolutionParameterfv(GLenum target, GLenum pname,
        GLfloat * params);
extern void glGetConvolutionParameteriv(GLenum target, GLenum pname,
        GLint * params);
extern void glGetDoublev(GLenum pname, GLdouble * params);
extern GLenum glGetError(void);
extern void glGetFloatv(GLenum pname, GLfloat * params);
extern void glGetHistogram(GLenum target, GLboolean reset, GLenum
format,
        GLenum type, GLvoid * values);
extern void glGetHistogramParameterfv(GLenum target, GLenum pname,
        GLfloat * params);
extern void glGetHistogramParameteriv(GLenum target, GLenum pname,
        GLint * params);
extern void glGetIntegerv(GLenum pname, GLint * params);
extern void glGetLightfv(GLenum light, GLenum pname, GLfloat *
params);
extern void glGetLightiv(GLenum light, GLenum pname, GLint *
params);
extern void glGetMapdv(GLenum target, GLenum query, GLdouble * v);
extern void glGetMapfv(GLenum target, GLenum query, GLfloat * v);
extern void glGetMapiv(GLenum target, GLenum query, GLint * v);
extern void glGetMaterialfv(GLenum face, GLenum pname, GLfloat *
params);
extern void glGetMaterialiv(GLenum face, GLenum pname, GLint *
params);
extern void glGetMinmax(GLenum target, GLboolean reset, GLenum
format,
        GLenum types, GLvoid * values);
extern void glGetMinmaxParameterfv(GLenum target, GLenum pname,
        GLfloat * params);
extern void glGetMinmaxParameteriv(GLenum target, GLenum pname,
        GLint * params);
extern void glGetPixelMapfv(GLenum map, GLfloat * values);
extern void glGetPixelMapuiv(GLenum map, GLuint * values);
extern void glGetPixelMapusv(GLenum map, GLushort * values);
extern void glGetPointerv(GLenum pname, void **params);
extern void glGetPolygonStipple(GLubyte * mask);
extern void glGetSeparableFilter(GLenum target, GLenum format,
        GLenum type,
        GLvoid * row, GLvoid * column,
        GLvoid * span);
extern const GLubyte *glGetString(GLenum name);
extern void glGetTexEnvfv(GLenum target, GLenum pname, GLfloat *
params);
extern void glGetTexEnviv(GLenum target, GLenum pname, GLint *
params);
extern void glGetTexGendv(GLenum coord, GLenum pname, GLdouble *
params);
extern void glGetTexGenfv(GLenum coord, GLenum pname, GLfloat *
params);
extern void glGetTexGeniv(GLenum coord, GLenum pname, GLint *
params);
extern void glGetTexImage(GLenum target, GLint level, GLenum format,

```

```

        GLenum type, GLvoid * pixels);
extern void glGetTexLevelParameterfv(GLenum target, GLint level,
        GLenum pname, GLfloat * params);
extern void glGetTexLevelParameteriv(GLenum target, GLint level,
        GLenum pname, GLint * params);
extern void glGetTexParameterfv(GLenum target, GLenum pname,
        GLfloat * params);
extern void glGetTexParameteriv(GLenum target, GLenum pname,
        GLint * params);
extern void glHint(GLenum target, GLenum mode);
extern void glHistogram(GLenum target, GLsizei width,
        GLenum internalformat, GLboolean sink);
extern void glIndexMask(GLuint mask);
extern void glIndexPointer(GLenum type, GLsizei stride,
        const GLvoid * pointer);
extern void glIndexd(GLdouble c);
extern void glIndexdv(const GLdouble * c);
extern void glIndexf(GLfloat c);
extern void glIndexfv(const GLfloat * c);
extern void glIndexi(GLint c);
extern void glIndexiv(const GLint * c);
extern void glIndexs(GLshort c);
extern void glIndexsv(const GLshort * c);
extern void glIndexub(GLubyte c);
extern void glIndexubv(const GLubyte * c);
extern void glInitNames(void);
extern void glInterleavedArrays(GLenum format, GLsizei stride,
        const GLvoid * pointer);
extern GLboolean glIsEnabled(GLenum cap);
extern GLboolean glIsList(GLuint list);
extern GLboolean glIsTexture(GLuint texture);
extern void glLightModelfv(GLenum pname, GLfloat param);
extern void glLightModelfiv(GLenum pname, const GLfloat * params);
extern void glLightModeli(GLenum pname, GLint param);
extern void glLightModeliv(GLenum pname, const GLint * params);
extern void glLightf(GLenum light, GLenum pname, GLfloat param);
extern void glLightfv(GLenum light, GLenum pname, const GLfloat *
params);
extern void glLighti(GLenum light, GLenum pname, GLint param);
extern void glLightiv(GLenum light, GLenum pname, const GLint *
params);
extern void glLineStipple(GLint factor, GLushort pattern);
extern void glLineWidth(GLfloat width);
extern void glListBase(GLuint base);
extern void glLoadIdentity(void);
extern void glLoadMatrixd(const GLdouble * m);
extern void glLoadMatrixf(const GLfloat * m);
extern void glLoadName(GLuint name);
extern void glLoadTransposeMatrixd(const GLdouble * m);
extern void glLoadTransposeMatrixf(const GLfloat * m);
extern void glLogicOp(GLenum opcode);
extern void glMap1d(GLenum target, GLdouble u1, GLdouble u2, GLint
stride,
        GLint order, const GLdouble * points);
extern void glMap1f(GLenum target, GLfloat u1, GLfloat u2, GLint
stride,
        GLint order, const GLfloat * points);
extern void glMap2d(GLenum target, GLdouble u1, GLdouble u2, GLint
ustride,
        GLint uorder, GLdouble v1, GLdouble v2, GLint
vstride,
        GLint vorder, const GLdouble * points);
extern void glMap2f(GLenum target, GLfloat u1, GLfloat u2, GLint
ustride,
        GLint uorder, GLfloat v1, GLfloat v2, GLint
vstride,

```

```

        GLint vorder, const GLfloat * points);
extern void glMapGrid1d(GLint un, GLdouble u1, GLdouble u2);
extern void glMapGrid1f(GLint un, GLfloat u1, GLfloat u2);
extern void glMapGrid2d(GLint un, GLdouble u1, GLdouble u2, GLint
vn,
        GLdouble v1, GLdouble v2);
extern void glMapGrid2f(GLint un, GLfloat u1, GLfloat u2, GLint vn,
        GLfloat v1, GLfloat v2);
extern void glMaterialf(GGLenum face, GGLenum pname, GLfloat param);
extern void glMaterialfv(GGLenum face, GGLenum pname,
        const GLfloat * params);
extern void glMateriali(GGLenum face, GGLenum pname, GLint param);
extern void glMaterialiv(GGLenum face, GGLenum pname, const GLint *
params);
extern void glMatrixMode(GGLenum mode);
extern void glMinmax(GGLenum target, GGLenum internalformat,
GLboolean sink);
extern void glMultMatrixd(const GLdouble * m);
extern void glMultMatrixf(const GLfloat * m);
extern void glMultTransposeMatrixd(const GLdouble * m);
extern void glMultTransposeMatrixf(const GLfloat * m);
extern void glMultiTexCoord1d(GGLenum target, GLdouble s);
extern void glMultiTexCoord1dARB(GGLenum target, GLdouble s);
extern void glMultiTexCoord1dv(GGLenum target, const GLdouble * v);
extern void glMultiTexCoord1dvARB(GGLenum target, const GLdouble *
v);
extern void glMultiTexCoord1f(GGLenum target, GLfloat s);
extern void glMultiTexCoord1fARB(GGLenum target, GLfloat s);
extern void glMultiTexCoord1fv(GGLenum target, const GLfloat * v);
extern void glMultiTexCoord1fvARB(GGLenum target, const GLfloat *
v);
extern void glMultiTexCoord1i(GGLenum target, GLint s);
extern void glMultiTexCoord1iARB(GGLenum target, GLint s);
extern void glMultiTexCoord1iv(GGLenum target, const GLint * v);
extern void glMultiTexCoord1ivARB(GGLenum target, const GLint * v);
extern void glMultiTexCoord1s(GGLenum target, GLshort s);
extern void glMultiTexCoord1sARB(GGLenum target, GLshort s);
extern void glMultiTexCoord1sv(GGLenum target, const GLshort * v);
extern void glMultiTexCoord1svARB(GGLenum target, const GLshort *
v);
extern void glMultiTexCoord2d(GGLenum target, GLdouble s, GLdouble
t);
extern void glMultiTexCoord2dARB(GGLenum target, GLdouble s,
GLdouble t);
extern void glMultiTexCoord2dv(GGLenum target, const GLdouble * v);
extern void glMultiTexCoord2dvARB(GGLenum target, const GLdouble *
v);
extern void glMultiTexCoord2f(GGLenum target, GLfloat s, GLfloat t);
extern void glMultiTexCoord2fARB(GGLenum target, GLfloat s, GLfloat
t);
extern void glMultiTexCoord2fv(GGLenum target, const GLfloat * v);
extern void glMultiTexCoord2fvARB(GGLenum target, const GLfloat *
v);
extern void glMultiTexCoord2i(GGLenum target, GLint s, GLint t);
extern void glMultiTexCoord2iARB(GGLenum target, GLint s, GLint t);
extern void glMultiTexCoord2iv(GGLenum target, const GLint * v);
extern void glMultiTexCoord2ivARB(GGLenum target, const GLint * v);
extern void glMultiTexCoord2s(GGLenum target, GLshort s, GLshort t);
extern void glMultiTexCoord2sARB(GGLenum target, GLshort s, GLshort
t);
extern void glMultiTexCoord2sv(GGLenum target, const GLshort * v);
extern void glMultiTexCoord2svARB(GGLenum target, const GLshort *
v);
extern void glMultiTexCoord3d(GGLenum target, GLdouble s, GLdouble
t,
        GLdouble r);

```



```

extern void glMultiTexCoord3dARB(GLenum target, GLdouble s,
GLdouble t,
GLdouble r);
extern void glMultiTexCoord3dv(GLenum target, const GLdouble * v);
extern void glMultiTexCoord3dvARB(GLenum target, const GLdouble *
v);
extern void glMultiTexCoord3f(GLenum target, GLfloat s, GLfloat t,
GLfloat r);
extern void glMultiTexCoord3fARB(GLenum target, GLfloat s, GLfloat
t,
GLfloat r);
extern void glMultiTexCoord3fv(GLenum target, const GLfloat * v);
extern void glMultiTexCoord3fvARB(GLenum target, const GLfloat *
v);
extern void glMultiTexCoord3i(GLenum target, GLint s, GLint t,
GLint r);
extern void glMultiTexCoord3iARB(GLenum target, GLint s, GLint t,
GLint r);
extern void glMultiTexCoord3iv(GLenum target, const GLint * v);
extern void glMultiTexCoord3ivARB(GLenum target, const GLint * v);
extern void glMultiTexCoord3s(GLenum target, GLshort s, GLshort t,
GLshort r);
extern void glMultiTexCoord3sARB(GLenum target, GLshort s, GLshort
t,
GLshort r);
extern void glMultiTexCoord3sv(GLenum target, const GLshort * v);
extern void glMultiTexCoord3svARB(GLenum target, const GLshort *
v);
extern void glMultiTexCoord4d(GLenum target, GLdouble s, GLdouble
t,
GLdouble r, GLdouble q);
extern void glMultiTexCoord4dARB(GLenum target, GLdouble s,
GLdouble t,
GLdouble r, GLdouble q);
extern void glMultiTexCoord4dv(GLenum target, const GLdouble * v);
extern void glMultiTexCoord4dvARB(GLenum target, const GLdouble *
v);
extern void glMultiTexCoord4f(GLenum target, GLfloat s, GLfloat t,
GLfloat r, GLfloat q);
extern void glMultiTexCoord4fARB(GLenum target, GLfloat s, GLfloat
t,
GLfloat r, GLfloat q);
extern void glMultiTexCoord4fv(GLenum target, const GLfloat * v);
extern void glMultiTexCoord4fvARB(GLenum target, const GLfloat *
v);
extern void glMultiTexCoord4i(GLenum target, GLint s, GLint t,
GLint r,
GLint q);
extern void glMultiTexCoord4iARB(GLenum target, GLint s, GLint t,
GLint r,
GLint q);
extern void glMultiTexCoord4iv(GLenum target, const GLint * v);
extern void glMultiTexCoord4ivARB(GLenum target, const GLint * v);
extern void glMultiTexCoord4s(GLenum target, GLshort s, GLshort t,
GLshort r, GLshort q);
extern void glMultiTexCoord4sARB(GLenum target, GLshort s, GLshort
t,
GLshort r, GLshort q);
extern void glMultiTexCoord4sv(GLenum target, const GLshort * v);
extern void glMultiTexCoord4svARB(GLenum target, const GLshort *
v);
extern void glNewList(GLuint list, GLenum mode);
extern void glNormal3b(GLbyte nx, GLbyte ny, GLbyte nz);
extern void glNormal3bv(const GLbyte * v);
extern void glNormal3d(GLdouble nx, GLdouble ny, GLdouble nz);
extern void glNormal3dv(const GLdouble * v);

```



```

extern void glNormal3f(GLfloat nx, GLfloat ny, GLfloat nz);
extern void glNormal3fv(const GLfloat * v);
extern void glNormal3i(GLint nx, GLint ny, GLint nz);
extern void glNormal3iv(const GLint * v);
extern void glNormal3s(GLshort nx, GLshort ny, GLshort nz);
extern void glNormal3sv(const GLshort * v);
extern void glNormalPointer(GLenum type, GLsizei stride,
                           const GLvoid * pointer);
extern void glOrtho(GLdouble left, GLdouble right, GLdouble bottom,
                   GLdouble top, GLdouble zNear, GLdouble zFar);
extern void glPassThrough(GLfloat token);
extern void glPixelMapfv(GLenum map, GLint mapsize,
                        const GLfloat * values);
extern void glPixelMapuiv(GLenum map, GLint mapsize,
                        const GLuint * values);
extern void glPixelMapusv(GLenum map, GLint mapsize,
                        const GLushort * values);
extern void glPixelStoref(GLenum pname, GLfloat param);
extern void glPixelStorei(GLenum pname, GLint param);
extern void glPixelTransferf(GLenum pname, GLfloat param);
extern void glPixelTransferi(GLenum pname, GLint param);
extern void glPixelZoom(GLfloat xfactor, GLfloat yfactor);
extern void glPointSize(GLfloat size);
extern void glPolygonMode(GLenum face, GLenum mode);
extern void glPolygonOffset(GLfloat factor, GLfloat units);
extern void glPolygonStipple(const GLubyte * mask);
extern void glPopAttrib(void);
extern void glPopClientAttrib(void);
extern void glPopMatrix(void);
extern void glPopName(void);
extern void glPrioritizeTextures(GLsizei n, const GLuint * textures,
                                const GLclampf * priorities);
extern void glPushAttrib(GLbitfield mask);
extern void glPushClientAttrib(GLbitfield mask);
extern void glPushMatrix(void);
extern void glPushName(GLuint name);
extern void glRasterPos2d(GLdouble x, GLdouble y);
extern void glRasterPos2dv(const GLdouble * v);
extern void glRasterPos2f(GLfloat x, GLfloat y);
extern void glRasterPos2fv(const GLfloat * v);
extern void glRasterPos2i(GLint x, GLint y);
extern void glRasterPos2iv(const GLint * v);
extern void glRasterPos2s(GLshort x, GLshort y);
extern void glRasterPos2sv(const GLshort * v);
extern void glRasterPos3d(GLdouble x, GLdouble y, GLdouble z);
extern void glRasterPos3dv(const GLdouble * v);
extern void glRasterPos3f(GLfloat x, GLfloat y, GLfloat z);
extern void glRasterPos3fv(const GLfloat * v);
extern void glRasterPos3i(GLint x, GLint y, GLint z);
extern void glRasterPos3iv(const GLint * v);
extern void glRasterPos3s(GLshort x, GLshort y, GLshort z);
extern void glRasterPos3sv(const GLshort * v);
extern void glRasterPos4d(GLdouble x, GLdouble y, GLdouble z,
                        GLdouble w);
extern void glRasterPos4dv(const GLdouble * v);
extern void glRasterPos4f(GLfloat x, GLfloat y, GLfloat z, GLfloat
                        w);
extern void glRasterPos4fv(const GLfloat * v);
extern void glRasterPos4i(GLint x, GLint y, GLint z, GLint w);
extern void glRasterPos4iv(const GLint * v);
extern void glRasterPos4s(GLshort x, GLshort y, GLshort z, GLshort
                        w);
extern void glRasterPos4sv(const GLshort * v);
extern void glReadBuffer(GLenum mode);
extern void glReadPixels(GLint x, GLint y, GLsizei width, GLsizei
                        height,

```

```

        GLenum format, GLenum type, GLvoid * pixels);
extern void glRectd(GLdouble x1, GLdouble y1, GLdouble x2, GLdouble
y2);
extern void glRectdv(const GLdouble * v1, const GLdouble * v2);
extern void glRectf(GLfloat x1, GLfloat y1, GLfloat x2, GLfloat y2);
extern void glRectfv(const GLfloat * v1, const GLfloat * v2);
extern void glRecti(GLint x1, GLint y1, GLint x2, GLint y2);
extern void glRectiv(const GLint * v1, const GLint * v2);
extern void glRects(GLshort x1, GLshort y1, GLshort x2, GLshort y2);
extern void glRectsv(const GLshort * v1, const GLshort * v2);
extern GLint glRenderMode(GLenum mode);
extern void glResetHistogram(GLenum target);
extern void glResetMinmax(GLenum target);
extern void glRotated(GLdouble angle, GLdouble x, GLdouble y,
GLdouble z);
extern void glRotatef(GLfloat angle, GLfloat x, GLfloat y, GLfloat
z);
extern void glSampleCoverage(GLclampf value, GLboolean invert);
extern void glScaled(GLdouble x, GLdouble y, GLdouble z);
extern void glScalef(GLfloat x, GLfloat y, GLfloat z);
extern void glScissor(GLint x, GLint y, GLsizei width, GLsizei
height);
extern void glSelectBuffer(GLsizei size, GLuint * buffer);
extern void glSeparableFilter2D(GLenum target, GLenum
internalformat,
                                GLsizei width, GLsizei height,
                                GLenum format, GLenum type,
                                const GLvoid * row, const GLvoid *
column);
extern void glShadeModel(GLenum mode);
extern void glStencilFunc(GLenum func, GLint ref, GLuint mask);
extern void glStencilMask(GLuint mask);
extern void glStencilOp(GLenum fail, GLenum zfail, GLenum zpass);
extern void glTexCoord1d(GLdouble s);
extern void glTexCoord1dv(const GLdouble * v);
extern void glTexCoord1f(GLfloat s);
extern void glTexCoord1fv(const GLfloat * v);
extern void glTexCoord1i(GLint s);
extern void glTexCoord1iv(const GLint * v);
extern void glTexCoord1s(GLshort s);
extern void glTexCoord1sv(const GLshort * v);
extern void glTexCoord2d(GLdouble s, GLdouble t);
extern void glTexCoord2dv(const GLdouble * v);
extern void glTexCoord2f(GLfloat s, GLfloat t);
extern void glTexCoord2fv(const GLfloat * v);
extern void glTexCoord2i(GLint s, GLint t);
extern void glTexCoord2iv(const GLint * v);
extern void glTexCoord2s(GLshort s, GLshort t);
extern void glTexCoord2sv(const GLshort * v);
extern void glTexCoord3d(GLdouble s, GLdouble t, GLdouble r);
extern void glTexCoord3dv(const GLdouble * v);
extern void glTexCoord3f(GLfloat s, GLfloat t, GLfloat r);
extern void glTexCoord3fv(const GLfloat * v);
extern void glTexCoord3i(GLint s, GLint t, GLint r);
extern void glTexCoord3iv(const GLint * v);
extern void glTexCoord3s(GLshort s, GLshort t, GLshort r);
extern void glTexCoord3sv(const GLshort * v);
extern void glTexCoord4d(GLdouble s, GLdouble t, GLdouble r,
GLdouble q);
extern void glTexCoord4dv(const GLdouble * v);
extern void glTexCoord4f(GLfloat s, GLfloat t, GLfloat r, GLfloat
q);
extern void glTexCoord4fv(const GLfloat * v);
extern void glTexCoord4i(GLint s, GLint t, GLint r, GLint q);
extern void glTexCoord4iv(const GLint * v);

```

```

extern void glTexCoord4s(GLshort s, GLshort t, GLshort r, GLshort
q);
extern void glTexCoord4sv(const GLshort * v);
extern void glTexCoordPointer(GLint size, GLenum type, GLsizei
stride,
                                const GLvoid * pointer);
extern void glTexEnvf(GLenum target, GLenum pname, GLfloat param);
extern void glTexEnvfv(GLenum target, GLenum pname,
                                const GLfloat * params);
extern void glTexEnvi(GLenum target, GLenum pname, GLint param);
extern void glTexEnviv(GLenum target, GLenum pname, const GLint *
params);
extern void glTexGend(GLenum coord, GLenum pname, GLdouble param);
extern void glTexGendv(GLenum coord, GLenum pname,
                                const GLdouble * params);
extern void glTexGenf(GLenum coord, GLenum pname, GLfloat param);
extern void glTexGenfv(GLenum coord, GLenum pname, const GLfloat *
params);
extern void glTexGeni(GLenum coord, GLenum pname, GLint param);
extern void glTexGeniv(GLenum coord, GLenum pname, const GLint *
params);
extern void glTexImage1D(GLenum target, GLint level, GLint
internalformat,
                                GLsizei width, GLint border, GLenum format,
                                GLenum type, const GLvoid * pixels);
extern void glTexImage2D(GLenum target, GLint level, GLint
internalformat,
                                GLsizei width, GLsizei height, GLint border,
                                GLenum format, GLenum type,
                                const GLvoid * pixels);
extern void glTexImage3D(GLenum target, GLint level, GLint
internalFormat,
                                GLsizei width, GLsizei height, GLsizei depth,
                                GLint border, GLenum format, GLenum type,
                                const GLvoid * pixels);
extern void glTexParameterf(GLenum target, GLenum pname, GLfloat
param);
extern void glTexParameterfv(GLenum target, GLenum pname,
                                const GLfloat * params);
extern void glTexParameteri(GLenum target, GLenum pname, GLint
param);
extern void glTexParameteriv(GLenum target, GLenum pname,
                                const GLint * params);
extern void glTexSubImage1D(GLenum target, GLint level, GLint
xoffset,
                                GLsizei width, GLenum format, GLenum type,
                                const GLvoid * pixels);
extern void glTexSubImage2D(GLenum target, GLint level, GLint
xoffset,
                                GLint yoffset, GLsizei width, GLsizei
height,
                                GLenum format, GLenum type,
                                const GLvoid * pixels);
extern void glTexSubImage3D(GLenum target, GLint level, GLint
xoffset,
                                GLint yoffset, GLint zoffset, GLsizei width,
                                GLsizei height, GLsizei depth, GLenum
format,
                                GLenum type, const GLvoid * pixels);
extern void glTranslated(GLdouble x, GLdouble y, GLdouble z);
extern void glTranslatef(GLfloat x, GLfloat y, GLfloat z);
extern void glVertex2d(GLdouble x, GLdouble y);
extern void glVertex2dv(const GLdouble * v);
extern void glVertex2f(GLfloat x, GLfloat y);
extern void glVertex2fv(const GLfloat * v);
extern void glVertex2i(GLint x, GLint y);

```

```

extern void glVertex2iv(const GLint * v);
extern void glVertex2s(GLshort x, GLshort y);
extern void glVertex2sv(const GLshort * v);
extern void glVertex3d(GLdouble x, GLdouble y, GLdouble z);
extern void glVertex3dv(const GLdouble * v);
extern void glVertex3f(GLfloat x, GLfloat y, GLfloat z);
extern void glVertex3fv(const GLfloat * v);
extern void glVertex3i(GLint x, GLint y, GLint z);
extern void glVertex3iv(const GLint * v);
extern void glVertex3s(GLshort x, GLshort y, GLshort z);
extern void glVertex3sv(const GLshort * v);
extern void glVertex4d(GLdouble x, GLdouble y, GLdouble z, GLdouble
w);
extern void glVertex4dv(const GLdouble * v);
extern void glVertex4f(GLfloat x, GLfloat y, GLfloat z, GLfloat w);
extern void glVertex4fv(const GLfloat * v);
extern void glVertex4i(GLint x, GLint y, GLint z, GLint w);
extern void glVertex4iv(const GLint * v);
extern void glVertex4s(GLshort x, GLshort y, GLshort z, GLshort w);
extern void glVertex4sv(const GLshort * v);
extern void glVertexAttribPointer(GLint size, GLenum type, GLsizei stride,
                                const GLvoid * pointer);
extern void glViewport(GLint x, GLint y, GLsizei width, GLsizei
height);

```

7.2.2 GL/glexth.h

```

#define GLEXT_64_TYPES_DEFINED
#define GL_2X_BIT_ATI 0x00000001
#define GL_RED_BIT_ATI 0x00000001
#define GL_4X_BIT_ATI 0x00000002
#define GL_COMP_BIT_ATI 0x00000002
#define GL_GREEN_BIT_ATI 0x00000002
#define GL_8X_BIT_ATI 0x00000004
#define GL_BLUE_BIT_ATI 0x00000004
#define GL_NEGATE_BIT_ATI 0x00000004
#define GL_VERTEX23_BIT_PGI 0x00000004
#define GL_BIAS_BIT_ATI 0x00000008
#define GL_HALF_BIT_ATI 0x00000008
#define GL_VERTEX4_BIT_PGI 0x00000008
#define GL_QUARTER_BIT_ATI 0x00000010
#define GL_EIGHTH_BIT_ATI 0x00000020
#define GL_SATURATE_BIT_ATI 0x00000040
#define GL_RESTART_SUN 0x0001
#define GL_COLOR3_BIT_PGI 0x00010000
#define GL_REPLACE_MIDDLE_SUN 0x0002
#define GL_COLOR4_BIT_PGI 0x00020000
#define GL_REPLACE_OLDEST_SUN 0x0003
#define GL_EDGEFLAG_BIT_PGI 0x00040000
#define GL_INDEX_BIT_PGI 0x00080000
#define GL_MAT_AMBIENT_BIT_PGI 0x00100000
#define GL_MAT_AMBIENT_AND_DIFFUSE_BIT_PGI 0x00200000
#define GL_MAT_DIFFUSE_BIT_PGI 0x00400000
#define GL_MAT_EMISSION_BIT_PGI 0x00800000
#define GL_MAT_COLOR_INDEXES_BIT_PGI 0x01000000
#define GL_MAT_SHININESS_BIT_PGI 0x02000000
#define GL_MAT_SPECULAR_BIT_PGI 0x04000000
#define GL_INVALID_FRAMEBUFFER_OPERATION_EXT 0x0506
#define GL_NORMAL_BIT_PGI 0x08000000
#define GL_TEXCOORD1_BIT_PGI 0x10000000
#define GL_HALF_FLOAT_ARB 0x140B
#define GL_MODELVIEW0_ARB 0x1700
#define GL_RASTER_POSITION_UNCLIPPED_IBM 0x19262
#define GL_PREFER_DOUBLEBUFFER_HINT_PGI 0x1A1F8
#define GL_CONSERVE_MEMORY_HINT_PGI 0x1A1FD

```

```

#define GL_RECLAIM_MEMORY_HINT_PGI 0x1A1FE
#define GL_NATIVE_GRAPHICS_HANDLE_PGI 0x1A202
#define GL_NATIVE_GRAPHICS_BEGIN_HINT_PGI 0x1A203
#define GL_NATIVE_GRAPHICS_END_HINT_PGI 0x1A204
#define GL_ALWAYS_FAST_HINT_PGI 0x1A20C
#define GL_ALWAYS_SOFT_HINT_PGI 0x1A20D
#define GL_ALLOW_DRAW_OBJ_HINT_PGI 0x1A20E
#define GL_ALLOW_DRAW_WIN_HINT_PGI 0x1A20F
#define GL_ALLOW_DRAW_FRG_HINT_PGI 0x1A210
#define GL_ALLOW_DRAW_MEM_HINT_PGI 0x1A211
#define GL_STRICT_DEPTHFUNC_HINT_PGI 0x1A216
#define GL_STRICT_LIGHTING_HINT_PGI 0x1A217
#define GL_STRICT_SCISSOR_HINT_PGI 0x1A218
#define GL_FULL_STIPPLE_HINT_PGI 0x1A219
#define GL_CLIP_NEAR_HINT_PGI 0x1A220
#define GL_CLIP_FAR_HINT_PGI 0x1A221
#define GL_WIDE_LINE_HINT_PGI 0x1A222
#define GL_BACK_NORMALS_HINT_PGI 0x1A223
#define GL_VERTEX_DATA_HINT_PGI 0x1A22A
#define GL_VERTEX_CONSISTENT_HINT_PGI 0x1A22B
#define GL_MATERIAL_SIDE_HINT_PGI 0x1A22C
#define GL_MAX_VERTEX_HINT_PGI 0x1A22D
#define GL_MULTISAMPLE_BIT_3DFX 0x20000000
#define GL_MULTISAMPLE_BIT_ARB 0x20000000
#define GL_MULTISAMPLE_BIT_EXT 0x20000000
#define GL_TEXCOORD2_BIT_PGI 0x20000000
#define GL_TEXCOORD3_BIT_PGI 0x40000000
#define GL_ABGR_EXT 0x8000
#define GL_TEXCOORD4_BIT_PGI 0x80000000
#define GL_CONSTANT_COLOR_EXT 0x8001
#define GL_ONE_MINUS_CONSTANT_COLOR_EXT 0x8002
#define GL_CONSTANT_ALPHA_EXT 0x8003
#define GL_ONE_MINUS_CONSTANT_ALPHA_EXT 0x8004
#define GL_BLEND_COLOR_EXT 0x8005
#define GL_FUNC_ADD_EXT 0x8006
#define GL_MIN_EXT 0x8007
#define GL_MAX_EXT 0x8008
#define GL_BLEND_EQUATION_EXT 0x8009
#define GL_BLEND_EQUATION_RGB 0x8009
#define GL_FUNC_SUBTRACT_EXT 0x800A
#define GL_FUNC_REVERSE_SUBTRACT_EXT 0x800B
#define GL_CMYK_EXT 0x800C
#define GL_CMYKA_EXT 0x800D
#define GL_PACK_CMYK_HINT_EXT 0x800E
#define GL_UNPACK_CMYK_HINT_EXT 0x800F
#define GL_CONVOLUTION_1D_EXT 0x8010
#define GL_CONVOLUTION_2D_EXT 0x8011
#define GL_SEPARABLE_2D_EXT 0x8012
#define GL_CONVOLUTION_BORDER_MODE_EXT 0x8013
#define GL_CONVOLUTION_FILTER_SCALE_EXT 0x8014
#define GL_CONVOLUTION_FILTER_BIAS_EXT 0x8015
#define GL_REDUCE_EXT 0x8016
#define GL_CONVOLUTION_FORMAT_EXT 0x8017
#define GL_CONVOLUTION_WIDTH_EXT 0x8018
#define GL_CONVOLUTION_HEIGHT_EXT 0x8019
#define GL_MAX_CONVOLUTION_WIDTH_EXT 0x801A
#define GL_MAX_CONVOLUTION_HEIGHT_EXT 0x801B
#define GL_POST_CONVOLUTION_RED_SCALE_EXT 0x801C
#define GL_POST_CONVOLUTION_GREEN_SCALE_EXT 0x801D
#define GL_POST_CONVOLUTION_BLUE_SCALE_EXT 0x801E
#define GL_POST_CONVOLUTION_ALPHA_SCALE_EXT 0x801F
#define GL_POST_CONVOLUTION_RED_BIAS_EXT 0x8020
#define GL_POST_CONVOLUTION_GREEN_BIAS_EXT 0x8021
#define GL_POST_CONVOLUTION_BLUE_BIAS_EXT 0x8022
#define GL_POST_CONVOLUTION_ALPHA_BIAS_EXT 0x8023
#define GL_HISTOGRAM_EXT 0x8024

```

```

#define GL_PROXY_HISTOGRAM_EXT 0x8025
#define GL_HISTOGRAM_WIDTH_EXT 0x8026
#define GL_HISTOGRAM_FORMAT_EXT 0x8027
#define GL_HISTOGRAM_RED_SIZE_EXT 0x8028
#define GL_HISTOGRAM_GREEN_SIZE_EXT 0x8029
#define GL_HISTOGRAM_BLUE_SIZE_EXT 0x802A
#define GL_HISTOGRAM_ALPHA_SIZE_EXT 0x802B
#define GL_HISTOGRAM_LUMINANCE_SIZE_EXT 0x802C
#define GL_HISTOGRAM_SINK_EXT 0x802D
#define GL_MINMAX_EXT 0x802E
#define GL_MINMAX_FORMAT_EXT 0x802F
#define GL_MINMAX_SINK_EXT 0x8030
#define GL_TABLE_TOO_LARGE_EXT 0x8031
#define GL_UNSIGNED_BYTE_3_3_2_EXT 0x8032
#define GL_UNSIGNED_SHORT_4_4_4_4_EXT 0x8033
#define GL_UNSIGNED_SHORT_5_5_5_1_EXT 0x8034
#define GL_UNSIGNED_INT_8_8_8_8_EXT 0x8035
#define GL_UNSIGNED_INT_10_10_10_2_EXT 0x8036
#define GL_POLYGON_OFFSET_EXT 0x8037
#define GL_POLYGON_OFFSET_FACTOR_EXT 0x8038
#define GL_POLYGON_OFFSET_BIAS_EXT 0x8039
#define GL_RESCALE_NORMAL_EXT 0x803A
#define GL_ALPHA4_EXT 0x803B
#define GL_ALPHA8_EXT 0x803C
#define GL_ALPHA12_EXT 0x803D
#define GL_ALPHA16_EXT 0x803E
#define GL_LUMINANCE4_EXT 0x803F
#define GL_LUMINANCE8_EXT 0x8040
#define GL_LUMINANCE12_EXT 0x8041
#define GL_LUMINANCE16_EXT 0x8042
#define GL_LUMINANCE4_ALPHA4_EXT 0x8043
#define GL_LUMINANCE6_ALPHA2_EXT 0x8044
#define GL_LUMINANCE8_ALPHA8_EXT 0x8045
#define GL_LUMINANCE12_ALPHA4_EXT 0x8046
#define GL_LUMINANCE12_ALPHA12_EXT 0x8047
#define GL_LUMINANCE16_ALPHA16_EXT 0x8048
#define GL_INTENSITY_EXT 0x8049
#define GL_INTENSITY4_EXT 0x804A
#define GL_INTENSITY8_EXT 0x804B
#define GL_INTENSITY12_EXT 0x804C
#define GL_INTENSITY16_EXT 0x804D
#define GL_RGB2_EXT 0x804E
#define GL_RGB4_EXT 0x804F
#define GL_RGB5_EXT 0x8050
#define GL_RGB8_EXT 0x8051
#define GL_RGB10_EXT 0x8052
#define GL_RGB12_EXT 0x8053
#define GL_RGB16_EXT 0x8054
#define GL_RGBA2_EXT 0x8055
#define GL_RGBA4_EXT 0x8056
#define GL_RGB5_A1_EXT 0x8057
#define GL_RGBA8_EXT 0x8058
#define GL_RGB10_A2_EXT 0x8059
#define GL_RGBA12_EXT 0x805A
#define GL_RGBA16_EXT 0x805B
#define GL_TEXTURE_RED_SIZE_EXT 0x805C
#define GL_TEXTURE_GREEN_SIZE_EXT 0x805D
#define GL_TEXTURE_BLUE_SIZE_EXT 0x805E
#define GL_TEXTURE_ALPHA_SIZE_EXT 0x805F
#define GL_TEXTURE_LUMINANCE_SIZE_EXT 0x8060
#define GL_TEXTURE_INTENSITY_SIZE_EXT 0x8061
#define GL_REPLACE_EXT 0x8062
#define GL_PROXY_TEXTURE_1D_EXT 0x8063
#define GL_PROXY_TEXTURE_2D_EXT 0x8064
#define GL_TEXTURE_TOO_LARGE_EXT 0x8065
#define GL_TEXTURE_PRIORITY_EXT 0x8066

```



```

#define GL_TEXTURE_RESIDENT_EXT 0x8067
#define GL_TEXTURE_1D_BINDING_EXT 0x8068
#define GL_TEXTURE_2D_BINDING_EXT 0x8069
#define GL_TEXTURE_3D_BINDING_EXT 0x806A
#define GL_PACK_SKIP_IMAGES_EXT 0x806B
#define GL_PACK_IMAGE_HEIGHT_EXT 0x806C
#define GL_UNPACK_SKIP_IMAGES_EXT 0x806D
#define GL_UNPACK_IMAGE_HEIGHT_EXT 0x806E
#define GL_TEXTURE_3D_EXT 0x806F
#define GL_PROXY_TEXTURE_3D_EXT 0x8070
#define GL_TEXTURE_DEPTH_EXT 0x8071
#define GL_TEXTURE_WRAP_R_EXT 0x8072
#define GL_MAX_3D_TEXTURE_SIZE_EXT 0x8073
#define GL_VERTEX_ARRAY_EXT 0x8074
#define GL_NORMAL_ARRAY_EXT 0x8075
#define GL_COLOR_ARRAY_EXT 0x8076
#define GL_INDEX_ARRAY_EXT 0x8077
#define GL_TEXTURE_COORD_ARRAY_EXT 0x8078
#define GL_EDGE_FLAG_ARRAY_EXT 0x8079
#define GL_VERTEX_ARRAY_SIZE_EXT 0x807A
#define GL_VERTEX_ARRAY_TYPE_EXT 0x807B
#define GL_VERTEX_ARRAY_STRIDE_EXT 0x807C
#define GL_VERTEX_ARRAY_COUNT_EXT 0x807D
#define GL_NORMAL_ARRAY_TYPE_EXT 0x807E
#define GL_NORMAL_ARRAY_STRIDE_EXT 0x807F
#define GL_NORMAL_ARRAY_COUNT_EXT 0x8080
#define GL_COLOR_ARRAY_SIZE_EXT 0x8081
#define GL_COLOR_ARRAY_TYPE_EXT 0x8082
#define GL_COLOR_ARRAY_STRIDE_EXT 0x8083
#define GL_COLOR_ARRAY_COUNT_EXT 0x8084
#define GL_INDEX_ARRAY_TYPE_EXT 0x8085
#define GL_INDEX_ARRAY_STRIDE_EXT 0x8086
#define GL_INDEX_ARRAY_COUNT_EXT 0x8087
#define GL_TEXTURE_COORD_ARRAY_SIZE_EXT 0x8088
#define GL_TEXTURE_COORD_ARRAY_TYPE_EXT 0x8089
#define GL_TEXTURE_COORD_ARRAY_STRIDE_EXT 0x808A
#define GL_TEXTURE_COORD_ARRAY_COUNT_EXT 0x808B
#define GL_EDGE_FLAG_ARRAY_STRIDE_EXT 0x808C
#define GL_EDGE_FLAG_ARRAY_COUNT_EXT 0x808D
#define GL_VERTEX_ARRAY_POINTER_EXT 0x808E
#define GL_NORMAL_ARRAY_POINTER_EXT 0x808F
#define GL_COLOR_ARRAY_POINTER_EXT 0x8090
#define GL_INDEX_ARRAY_POINTER_EXT 0x8091
#define GL_TEXTURE_COORD_ARRAY_POINTER_EXT 0x8092
#define GL_EDGE_FLAG_ARRAY_POINTER_EXT 0x8093
#define GL_MULTISAMPLE_ARB 0x809D
#define GL_MULTISAMPLE_EXT 0x809D
#define GL_SAMPLE_ALPHA_TO_COVERAGE_ARB 0x809E
#define GL_SAMPLE_ALPHA_TO_MASK_EXT 0x809E
#define GL_SAMPLE_ALPHA_TO_ONE_ARB 0x809F
#define GL_SAMPLE_ALPHA_TO_ONE_EXT 0x809F
#define GL_SAMPLE_COVERAGE_ARB 0x80A0
#define GL_SAMPLE_MASK_EXT 0x80A0
#define GL_1PASS_EXT 0x80A1
#define GL_2PASS_0_EXT 0x80A2
#define GL_2PASS_1_EXT 0x80A3
#define GL_4PASS_0_EXT 0x80A4
#define GL_4PASS_1_EXT 0x80A5
#define GL_4PASS_2_EXT 0x80A6
#define GL_4PASS_3_EXT 0x80A7
#define GL_SAMPLE_BUFFERS_ARB 0x80A8
#define GL_SAMPLE_BUFFERS_EXT 0x80A8
#define GL_SAMPLES_ARB 0x80A9
#define GL_SAMPLES_EXT 0x80A9
#define GL_SAMPLE_COVERAGE_VALUE_ARB 0x80AA
#define GL_SAMPLE_MASK_VALUE_EXT 0x80AA

```

```

#define GL_SAMPLE_COVERAGE_INVERT_ARB 0x80AB
#define GL_SAMPLE_MASK_INVERT_EXT 0x80AB
#define GL_SAMPLE_PATTERN_EXT 0x80AC
#define GL_TEXTURE_COMPARE_FAIL_VALUE_ARB 0x80BF
#define GL_BLEND_DST_RGB 0x80C8
#define GL_BLEND_DST_RGB_EXT 0x80C8
#define GL_BLEND_SRC_RGB 0x80C9
#define GL_BLEND_SRC_RGB_EXT 0x80C9
#define GL_BLEND_DST_ALPHA 0x80CA
#define GL_BLEND_DST_ALPHA_EXT 0x80CA
#define GL_BLEND_SRC_ALPHA 0x80CB
#define GL_BLEND_SRC_ALPHA_EXT 0x80CB
#define GL_422_EXT 0x80CC
#define GL_422_REV_EXT 0x80CD
#define GL_422_AVERAGE_EXT 0x80CE
#define GL_422_REV_AVERAGE_EXT 0x80CF
#define GL_BGR_EXT 0x80E0
#define GL_BGRA_EXT 0x80E1
#define GL_COLOR_INDEX1_EXT 0x80E2
#define GL_COLOR_INDEX2_EXT 0x80E3
#define GL_COLOR_INDEX4_EXT 0x80E4
#define GL_COLOR_INDEX8_EXT 0x80E5
#define GL_COLOR_INDEX12_EXT 0x80E6
#define GL_COLOR_INDEX16_EXT 0x80E7
#define GL_MAX_ELEMENTS_VERTICES_EXT 0x80E8
#define GL_MAX_ELEMENTS_INDICES_EXT 0x80E9
#define GL_PHONG_WIN 0x80EA
#define GL_PHONG_HINT_WIN 0x80EB
#define GL_FOG_SPECULAR_TEXTURE_WIN 0x80EC
#define GL_TEXTURE_INDEX_SIZE_EXT 0x80ED
#define GL_CLIP_VOLUME_CLIPPING_HINT_EXT 0x80F0
#define GL_POINT_SIZE_MIN 0x8126
#define GL_POINT_SIZE_MIN_ARB 0x8126
#define GL_POINT_SIZE_MIN_EXT 0x8126
#define GL_POINT_SIZE_MAX 0x8127
#define GL_POINT_SIZE_MAX_ARB 0x8127
#define GL_POINT_SIZE_MAX_EXT 0x8127
#define GL_POINT_FADE_THRESHOLD_SIZE 0x8128
#define GL_POINT_FADE_THRESHOLD_SIZE_ARB 0x8128
#define GL_POINT_FADE_THRESHOLD_SIZE_EXT 0x8128
#define GL_DISTANCE_ATTENUATION_EXT 0x8129
#define GL_POINT_DISTANCE_ATTENUATION 0x8129
#define GL_POINT_DISTANCE_ATTENUATION_ARB 0x8129
#define GL_CLAMP_TO_BORDER_ARB 0x812D
#define GL_IGNORE_BORDER_HP 0x8150
#define GL_CONSTANT_BORDER_HP 0x8151
#define GL_REPLICATE_BORDER_HP 0x8153
#define GL_CONVOLUTION_BORDER_COLOR_HP 0x8154
#define GL_IMAGE_SCALE_X_HP 0x8155
#define GL_IMAGE_SCALE_Y_HP 0x8156
#define GL_IMAGE_TRANSLATE_X_HP 0x8157
#define GL_IMAGE_TRANSLATE_Y_HP 0x8158
#define GL_IMAGE_ROTATE_ANGLE_HP 0x8159
#define GL_IMAGE_ROTATE_ORIGIN_X_HP 0x815A
#define GL_IMAGE_ROTATE_ORIGIN_Y_HP 0x815B
#define GL_IMAGE_MAG_FILTER_HP 0x815C
#define GL_IMAGE_MIN_FILTER_HP 0x815D
#define GL_IMAGE_CUBIC_WEIGHT_HP 0x815E
#define GL_CUBIC_HP 0x815F
#define GL_AVERAGE_HP 0x8160
#define GL_IMAGE_TRANSFORM_2D_HP 0x8161
#define GL_POST_IMAGE_TRANSFORM_COLOR_TABLE_HP 0x8162
#define GL_PROXY_POST_IMAGE_TRANSFORM_COLOR_TABLE_HP 0x8163
#define GL_OCCLUSION_TEST_HP 0x8165
#define GL_OCCLUSION_TEST_RESULT_HP 0x8166
#define GL_TEXTURE_LIGHTING_MODE_HP 0x8167

```



```

#define GL_TEXTURE_POST_SPECULAR_HP 0x8168
#define GL_TEXTURE_PRE_SPECULAR_HP 0x8169
#define GL_GENERATE_MIPMAP 0x8191
#define GL_GENERATE_MIPMAP_HINT 0x8192
#define GL_DEPTH_COMPONENT16 0x81A5
#define GL_DEPTH_COMPONENT16_ARB 0x81A5
#define GL_DEPTH_COMPONENT24 0x81A6
#define GL_DEPTH_COMPONENT24_ARB 0x81A6
#define GL_DEPTH_COMPONENT32 0x81A7
#define GL_DEPTH_COMPONENT32_ARB 0x81A7
#define GL_ARRAY_ELEMENT_LOCK_FIRST_EXT 0x81A8
#define GL_ARRAY_ELEMENT_LOCK_COUNT_EXT 0x81A9
#define GL_CULL_VERTEX_EXT 0x81AA
#define GL_CULL_VERTEX_EYE_POSITION_EXT 0x81AB
#define GL_CULL_VERTEX_OBJECT_POSITION_EXT 0x81AC
#define GL_IUI_V2F_EXT 0x81AD
#define GL_IUI_V3F_EXT 0x81AE
#define GL_IUI_N3F_V2F_EXT 0x81AF
#define GL_IUI_N3F_V3F_EXT 0x81B0
#define GL_T2F_IUI_V2F_EXT 0x81B1
#define GL_T2F_IUI_V3F_EXT 0x81B2
#define GL_T2F_IUI_N3F_V2F_EXT 0x81B3
#define GL_T2F_IUI_N3F_V3F_EXT 0x81B4
#define GL_INDEX_TEST_EXT 0x81B5
#define GL_INDEX_TEST_FUNC_EXT 0x81B6
#define GL_INDEX_TEST_REF_EXT 0x81B7
#define GL_INDEX_MATERIAL_EXT 0x81B8
#define GL_INDEX_MATERIAL_PARAMETER_EXT 0x81B9
#define GL_INDEX_MATERIAL_FACE_EXT 0x81BA
#define GL_WRAP_BORDER_SUN 0x81D4
#define GL_UNPACK_CONSTANT_DATA_SUNX 0x81D5
#define GL_TEXTURE_CONSTANT_DATA_SUNX 0x81D6
#define GL_TRIANGLE_LIST_SUN 0x81D7
#define GL_REPLACEMENT_CODE_SUN 0x81D8
#define GL_GLOBAL_ALPHA_SUN 0x81D9
#define GL_GLOBAL_ALPHA_FACTOR_SUN 0x81DA
#define GL_LIGHT_MODEL_COLOR_CONTROL_EXT 0x81F8
#define GL_SINGLE_COLOR_EXT 0x81F9
#define GL_SEPARATE_SPECULAR_COLOR_EXT 0x81FA
#define GL_SHARED_TEXTURE_PALETTE_EXT 0x81FB
#define GL_TEXT_FRAGMENT_SHADER_ATI 0x8200
#define GL_PIXEL_TRANSFORM_2D_EXT 0x8330
#define GL_PIXEL_MAG_FILTER_EXT 0x8331
#define GL_PIXEL_MIN_FILTER_EXT 0x8332
#define GL_PIXEL_CUBIC_WEIGHT_EXT 0x8333
#define GL_CUBIC_EXT 0x8334
#define GL_AVERAGE_EXT 0x8335
#define GL_PIXEL_TRANSFORM_2D_STACK_DEPTH_EXT 0x8336
#define GL_MAX_PIXEL_TRANSFORM_2D_STACK_DEPTH_EXT 0x8337
#define GL_PIXEL_TRANSFORM_2D_MATRIX_EXT 0x8338
#define GL_FRAGMENT_MATERIAL_EXT 0x8349
#define GL_FRAGMENT_NORMAL_EXT 0x834A
#define GL_FRAGMENT_COLOR_EXT 0x834C
#define GL_ATTENUATION_EXT 0x834D
#define GL_SHADOW_ATTENUATION_EXT 0x834E
#define GL_TEXTURE_APPLICATION_MODE_EXT 0x834F
#define GL_TEXTURE_LIGHT_EXT 0x8350
#define GL_TEXTURE_MATERIAL_FACE_EXT 0x8351
#define GL_TEXTURE_MATERIAL_PARAMETER_EXT 0x8352
#define GL_MIRRORED_REPEAT 0x8370
#define GL_MIRRORED_REPEAT_ARB 0x8370
#define GL_MIRRORED_REPEAT_IBM 0x8370
#define GL_RGB_S3TC 0x83A0
#define GL_RGBA_S3TC 0x83A1
#define GL_RGBA_S3TC 0x83A2
#define GL_RGBA4_S3TC 0x83A3

```

```

#define GL_COMPRESSED_RGB_S3TC_DXT1_EXT 0x83F0
#define GL_COMPRESSED_RGBA_S3TC_DXT1_EXT 0x83F1
#define GL_COMPRESSED_RGBA_S3TC_DXT3_EXT 0x83F2
#define GL_COMPRESSED_RGBA_S3TC_DXT5_EXT 0x83F3
#define GL_PARALLEL_ARRAYS_INTEL 0x83F4
#define GL_VERTEX_ARRAY_PARALLEL_POINTERS_INTEL 0x83F5
#define GL_NORMAL_ARRAY_PARALLEL_POINTERS_INTEL 0x83F6
#define GL_COLOR_ARRAY_PARALLEL_POINTERS_INTEL 0x83F7
#define GL_TEXTURE_COORD_ARRAY_PARALLEL_POINTERS_INTEL 0x83F8
#define GL_TANGENT_ARRAY_EXT 0x8439
#define GL_BINORMAL_ARRAY_EXT 0x843A
#define GL_CURRENT_TANGENT_EXT 0x843B
#define GL_CURRENT_BINORMAL_EXT 0x843C
#define GL_TANGENT_ARRAY_TYPE_EXT 0x843E
#define GL_TANGENT_ARRAY_STRIDE_EXT 0x843F
#define GL_BINORMAL_ARRAY_TYPE_EXT 0x8440
#define GL_BINORMAL_ARRAY_STRIDE_EXT 0x8441
#define GL_TANGENT_ARRAY_POINTER_EXT 0x8442
#define GL_BINORMAL_ARRAY_POINTER_EXT 0x8443
#define GL_MAP1_TANGENT_EXT 0x8444
#define GL_MAP2_TANGENT_EXT 0x8445
#define GL_MAP1_BINORMAL_EXT 0x8446
#define GL_MAP2_BINORMAL_EXT 0x8447
#define GL_FOG_COORDINATE_SOURCE 0x8450
#define GL_FOG_COORDINATE_SOURCE_EXT 0x8450
#define GL_FOG_COORD_SRC 0x8450
#define GL_FOG_COORD 0x8451
#define GL_FOG_COORDINATE 0x8451
#define GL_FOG_COORDINATE_EXT 0x8451
#define GL_FRAGMENT_DEPTH 0x8452
#define GL_FRAGMENT_DEPTH_EXT 0x8452
#define GL_CURRENT_FOG_COORD 0x8453
#define GL_CURRENT_FOG_COORDINATE 0x8453
#define GL_CURRENT_FOG_COORDINATE_EXT 0x8453
#define GL_FOG_COORDINATE_ARRAY_TYPE 0x8454
#define GL_FOG_COORDINATE_ARRAY_TYPE_EXT 0x8454
#define GL_FOG_COORD_ARRAY_TYPE 0x8454
#define GL_FOG_COORDINATE_ARRAY_STRIDE 0x8455
#define GL_FOG_COORDINATE_ARRAY_STRIDE_EXT 0x8455
#define GL_FOG_COORD_ARRAY_STRIDE 0x8455
#define GL_FOG_COORDINATE_ARRAY_POINTER 0x8456
#define GL_FOG_COORDINATE_ARRAY_POINTER_EXT 0x8456
#define GL_FOG_COORD_ARRAY_POINTER 0x8456
#define GL_FOG_COORDINATE_ARRAY 0x8457
#define GL_FOG_COORDINATE_ARRAY_EXT 0x8457
#define GL_FOG_COORD_ARRAY 0x8457
#define GL_COLOR_SUM 0x8458
#define GL_COLOR_SUM_ARB 0x8458
#define GL_COLOR_SUM_EXT 0x8458
#define GL_CURRENT_SECONDARY_COLOR 0x8459
#define GL_CURRENT_SECONDARY_COLOR_EXT 0x8459
#define GL_SECONDARY_COLOR_ARRAY_SIZE 0x845A
#define GL_SECONDARY_COLOR_ARRAY_SIZE_EXT 0x845A
#define GL_SECONDARY_COLOR_ARRAY_TYPE 0x845B
#define GL_SECONDARY_COLOR_ARRAY_TYPE_EXT 0x845B
#define GL_SECONDARY_COLOR_ARRAY_STRIDE 0x845C
#define GL_SECONDARY_COLOR_ARRAY_STRIDE_EXT 0x845C
#define GL_SECONDARY_COLOR_ARRAY_POINTER 0x845D
#define GL_SECONDARY_COLOR_ARRAY_POINTER_EXT 0x845D
#define GL_SECONDARY_COLOR_ARRAY 0x845E
#define GL_SECONDARY_COLOR_ARRAY_EXT 0x845E
#define GL_CURRENT_RASTER_SECONDARY_COLOR 0x845F
#define GL_SCREEN_COORDINATES_REND 0x8490
#define GL_INVERTED_SCREEN_W_REND 0x8491
#define GL_TRANSPOSE_MODELVIEW_MATRIX_ARB 0x84E3
#define GL_TRANSPOSE_PROJECTION_MATRIX_ARB 0x84E4

```

```

#define GL_TRANSPOSE_TEXTURE_MATRIX_ARB 0x84E5
#define GL_TRANSPOSE_COLOR_MATRIX_ARB 0x84E6
#define GL_SUBTRACT_ARB 0x84E7
#define GL_MAX_RENDERBUFFER_SIZE_EXT 0x84E8
#define GL_COMPRESSED_ALPHA_ARB 0x84E9
#define GL_COMPRESSED_LUMINANCE_ARB 0x84EA
#define GL_COMPRESSED_LUMINANCE_ALPHA_ARB 0x84EB
#define GL_COMPRESSED_INTENSITY_ARB 0x84EC
#define GL_COMPRESSED_RGB_ARB 0x84ED
#define GL_COMPRESSED_RGBA_ARB 0x84EE
#define GL_TEXTURE_COMPRESSION_HINT_ARB 0x84EF
#define GL_TEXTURE_RECTANGLE_ARB 0x84F5
#define GL_TEXTURE_BINDING_RECTANGLE_ARB 0x84F6
#define GL_PROXY_TEXTURE_RECTANGLE_ARB 0x84F7
#define GL_MAX_RECTANGLE_TEXTURE_SIZE_ARB 0x84F8
#define GL_MAX_TEXTURE_LOD_BIAS 0x84FD
#define GL_MAX_TEXTURE_LOD_BIAS_EXT 0x84FD
#define GL_TEXTURE_MAX_ANISOTROPY_EXT 0x84FE
#define GL_MAX_TEXTURE_MAX_ANISOTROPY_EXT 0x84FF
#define GL_TEXTURE_FILTER_CONTROL 0x8500
#define GL_TEXTURE_FILTER_CONTROL_EXT 0x8500
#define GL_TEXTURE_LOD_BIAS 0x8501
#define GL_TEXTURE_LOD_BIAS_EXT 0x8501
#define GL_MODELVIEW1_STACK_DEPTH_EXT 0x8502
#define GL_MODELVIEW1_MATRIX_EXT 0x8506
#define GL_INCR_WRAP 0x8507
#define GL_INCR_WRAP_EXT 0x8507
#define GL_DECR_WRAP 0x8508
#define GL_DECR_WRAP_EXT 0x8508
#define GL_VERTEX_WEIGHTING_EXT 0x8509
#define GL_MODELVIEW1_ARB 0x850A
#define GL_MODELVIEW1_EXT 0x850A
#define GL_CURRENT_VERTEX_WEIGHT_EXT 0x850B
#define GL_VERTEX_WEIGHT_ARRAY_EXT 0x850C
#define GL_VERTEX_WEIGHT_ARRAY_SIZE_EXT 0x850D
#define GL_VERTEX_WEIGHT_ARRAY_TYPE_EXT 0x850E
#define GL_VERTEX_WEIGHT_ARRAY_STRIDE_EXT 0x850F
#define GL_VERTEX_WEIGHT_ARRAY_POINTER_EXT 0x8510
#define GL_NORMAL_MAP_ARB 0x8511
#define GL_NORMAL_MAP_EXT 0x8511
#define GL_REFLECTION_MAP_ARB 0x8512
#define GL_REFLECTION_MAP_EXT 0x8512
#define GL_TEXTURE_CUBE_MAP_ARB 0x8513
#define GL_TEXTURE_CUBE_MAP_EXT 0x8513
#define GL_TEXTURE_BINDING_CUBE_MAP_ARB 0x8514
#define GL_TEXTURE_BINDING_CUBE_MAP_EXT 0x8514
#define GL_TEXTURE_CUBE_MAP_POSITIVE_X_ARB 0x8515
#define GL_TEXTURE_CUBE_MAP_POSITIVE_X_EXT 0x8515
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_X_ARB 0x8516
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_X_EXT 0x8516
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Y_ARB 0x8517
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Y_EXT 0x8517
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y_ARB 0x8518
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y_EXT 0x8518
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Z_ARB 0x8519
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Z_EXT 0x8519
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z_ARB 0x851A
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z_EXT 0x851A
#define GL_PROXY_TEXTURE_CUBE_MAP_ARB 0x851B
#define GL_PROXY_TEXTURE_CUBE_MAP_EXT 0x851B
#define GL_MAX_CUBE_MAP_TEXTURE_SIZE_ARB 0x851C
#define GL_MAX_CUBE_MAP_TEXTURE_SIZE_EXT 0x851C
#define GL_RED_MIN_CLAMP_INGR 0x8560
#define GL_GREEN_MIN_CLAMP_INGR 0x8561
#define GL_BLUE_MIN_CLAMP_INGR 0x8562
#define GL_ALPHA_MIN_CLAMP_INGR 0x8563

```

```

#define GL_RED_MAX_CLAMP_INGR 0x8564
#define GL_GREEN_MAX_CLAMP_INGR 0x8565
#define GL_BLUE_MAX_CLAMP_INGR 0x8566
#define GL_ALPHA_MAX_CLAMP_INGR 0x8567
#define GL_INTERLACE_READ_INGR 0x8568
#define GL_COMBINE_ARB 0x8570
#define GL_COMBINE_EXT 0x8570
#define GL_COMBINE_RGB_ARB 0x8571
#define GL_COMBINE_RGB_EXT 0x8571
#define GL_COMBINE_ALPHA_ARB 0x8572
#define GL_COMBINE_ALPHA_EXT 0x8572
#define GL_RGB_SCALE_ARB 0x8573
#define GL_RGB_SCALE_EXT 0x8573
#define GL_ADD_SIGNED_ARB 0x8574
#define GL_ADD_SIGNED_EXT 0x8574
#define GL_INTERPOLATE_ARB 0x8575
#define GL_INTERPOLATE_EXT 0x8575
#define GL_CONSTANT_ARB 0x8576
#define GL_CONSTANT_EXT 0x8576
#define GL_PRIMARY_COLOR_ARB 0x8577
#define GL_PRIMARY_COLOR_EXT 0x8577
#define GL_PREVIOUS_ARB 0x8578
#define GL_PREVIOUS_EXT 0x8578
#define GL_SOURCE0_RGB_ARB 0x8580
#define GL_SOURCE0_RGB_EXT 0x8580
#define GL_SRC0_RGB 0x8580
#define GL_SOURCE1_RGB_ARB 0x8581
#define GL_SOURCE1_RGB_EXT 0x8581
#define GL_SRC1_RGB 0x8581
#define GL_SOURCE2_RGB_ARB 0x8582
#define GL_SOURCE2_RGB_EXT 0x8582
#define GL_SRC2_RGB 0x8582
#define GL_SOURCE0_ALPHA_ARB 0x8588
#define GL_SOURCE0_ALPHA_EXT 0x8588
#define GL_SRC0_ALPHA 0x8588
#define GL_SOURCE1_ALPHA_ARB 0x8589
#define GL_SOURCE1_ALPHA_EXT 0x8589
#define GL_SRC1_ALPHA 0x8589
#define GL_SOURCE2_ALPHA_ARB 0x858A
#define GL_SOURCE2_ALPHA_EXT 0x858A
#define GL_SRC2_ALPHA 0x858A
#define GL_OPERAND0_RGB_ARB 0x8590
#define GL_OPERAND0_RGB_EXT 0x8590
#define GL_OPERAND1_RGB_ARB 0x8591
#define GL_OPERAND1_RGB_EXT 0x8591
#define GL_OPERAND2_RGB_ARB 0x8592
#define GL_OPERAND2_RGB_EXT 0x8592
#define GL_OPERAND0_ALPHA_ARB 0x8598
#define GL_OPERAND0_ALPHA_EXT 0x8598
#define GL_OPERAND1_ALPHA_ARB 0x8599
#define GL_OPERAND1_ALPHA_EXT 0x8599
#define GL_OPERAND2_ALPHA_ARB 0x859A
#define GL_OPERAND2_ALPHA_EXT 0x859A
#define GL_PERTURB_EXT 0x85AE
#define GL_TEXTURE_NORMAL_EXT 0x85AF
#define GL_REPLACEMENT_CODE_ARRAY_SUN 0x85C0
#define GL_REPLACEMENT_CODE_ARRAY_TYPE_SUN 0x85C1
#define GL_REPLACEMENT_CODE_ARRAY_STRIDE_SUN 0x85C2
#define GL_REPLACEMENT_CODE_ARRAY_POINTER_SUN 0x85C3
#define GL_R1UI_V3F_SUN 0x85C4
#define GL_R1UI_C4UB_V3F_SUN 0x85C5
#define GL_R1UI_C3F_V3F_SUN 0x85C6
#define GL_R1UI_N3F_V3F_SUN 0x85C7
#define GL_R1UI_C4F_N3F_V3F_SUN 0x85C8
#define GL_R1UI_T2F_V3F_SUN 0x85C9
#define GL_R1UI_T2F_N3F_V3F_SUN 0x85CA

```

```

#define GL_R1UI_T2F_C4F_N3F_V3F_SUN 0x85CB
#define GL_SLICE_ACCUM_SUN 0x85CC
#define GL_QUAD_MESH_SUN 0x8614
#define GL_TRIANGLE_MESH_SUN 0x8615
#define GL_VERTEX_PROGRAM_ARB 0x8620
#define GL_VERTEX_ATTRIB_ARRAY_ENABLED 0x8622
#define GL_VERTEX_ATTRIB_ARRAY_ENABLED_ARB 0x8622
#define GL_VERTEX_ATTRIB_ARRAY_SIZE 0x8623
#define GL_VERTEX_ATTRIB_ARRAY_SIZE_ARB 0x8623
#define GL_VERTEX_ATTRIB_ARRAY_STRIDE 0x8624
#define GL_VERTEX_ATTRIB_ARRAY_STRIDE_ARB 0x8624
#define GL_VERTEX_ATTRIB_ARRAY_TYPE 0x8625
#define GL_VERTEX_ATTRIB_ARRAY_TYPE_ARB 0x8625
#define GL_CURRENT_VERTEX_ATTRIB 0x8626
#define GL_CURRENT_VERTEX_ATTRIB_ARB 0x8626
#define GL_PROGRAM_LENGTH_ARB 0x8627
#define GL_PROGRAM_STRING_ARB 0x8628
#define GL_MAX_PROGRAM_MATRIX_STACK_DEPTH_ARB 0x862E
#define GL_MAX_PROGRAM_MATRICES_ARB 0x862F
#define GL_CURRENT_MATRIX_STACK_DEPTH_ARB 0x8640
#define GL_CURRENT_MATRIX_ARB 0x8641
#define GL_VERTEX_PROGRAM_POINT_SIZE 0x8642
#define GL_VERTEX_PROGRAM_POINT_SIZE_ARB 0x8642
#define GL_VERTEX_PROGRAM_TWO_SIDE 0x8643
#define GL_VERTEX_PROGRAM_TWO_SIDE_ARB 0x8643
#define GL_VERTEX_ATTRIB_ARRAY_POINTER 0x8645
#define GL_VERTEX_ATTRIB_ARRAY_POINTER_ARB 0x8645
#define GL_PROGRAM_ERROR_POSITION_ARB 0x864B
#define GL_PROGRAM_BINDING_ARB 0x8677
#define GL_TEXTURE_COMPRESSED_IMAGE_SIZE_ARB 0x86A0
#define GL_TEXTURE_COMPRESSED_ARB 0x86A1
#define GL_NUM_COMPRESSED_TEXTURE_FORMATS_ARB 0x86A2
#define GL_COMPRESSED_TEXTURE_FORMATS_ARB 0x86A3
#define GL_MAX_VERTEX_UNITS_ARB 0x86A4
#define GL_ACTIVE_VERTEX_UNITS_ARB 0x86A5
#define GL_WEIGHT_SUM_UNITY_ARB 0x86A6
#define GL_VERTEX_BLEND_ARB 0x86A7
#define GL_CURRENT_WEIGHT_ARB 0x86A8
#define GL_WEIGHT_ARRAY_TYPE_ARB 0x86A9
#define GL_WEIGHT_ARRAY_STRIDE_ARB 0x86AA
#define GL_WEIGHT_ARRAY_SIZE_ARB 0x86AB
#define GL_WEIGHT_ARRAY_POINTER_ARB 0x86AC
#define GL_WEIGHT_ARRAY_ARB 0x86AD
#define GL_DOT3_RGB_ARB 0x86AE
#define GL_DOT3_RGBA_ARB 0x86AF
#define GL_COMPRESSED_RGB_FXT1_3DFX 0x86B0
#define GL_COMPRESSED_RGBA_FXT1_3DFX 0x86B1
#define GL_MULTISAMPLE_3DFX 0x86B2
#define GL_SAMPLE_BUFFERS_3DFX 0x86B3
#define GL_SAMPLES_3DFX 0x86B4
#define GL_MODELVIEW2_ARB 0x8722
#define GL_MODELVIEW3_ARB 0x8723
#define GL_MODELVIEW4_ARB 0x8724
#define GL_MODELVIEW5_ARB 0x8725
#define GL_MODELVIEW6_ARB 0x8726
#define GL_MODELVIEW7_ARB 0x8727
#define GL_MODELVIEW8_ARB 0x8728
#define GL_MODELVIEW9_ARB 0x8729
#define GL_MODELVIEW10_ARB 0x872A
#define GL_MODELVIEW11_ARB 0x872B
#define GL_MODELVIEW12_ARB 0x872C
#define GL_MODELVIEW13_ARB 0x872D
#define GL_MODELVIEW14_ARB 0x872E
#define GL_MODELVIEW15_ARB 0x872F
#define GL_MODELVIEW16_ARB 0x8730
#define GL_MODELVIEW17_ARB 0x8731

```

```

#define GL_MODELVIEW18_ARB 0x8732
#define GL_MODELVIEW19_ARB 0x8733
#define GL_MODELVIEW20_ARB 0x8734
#define GL_MODELVIEW21_ARB 0x8735
#define GL_MODELVIEW22_ARB 0x8736
#define GL_MODELVIEW23_ARB 0x8737
#define GL_MODELVIEW24_ARB 0x8738
#define GL_MODELVIEW25_ARB 0x8739
#define GL_MODELVIEW26_ARB 0x873A
#define GL_MODELVIEW27_ARB 0x873B
#define GL_MODELVIEW28_ARB 0x873C
#define GL_MODELVIEW29_ARB 0x873D
#define GL_MODELVIEW30_ARB 0x873E
#define GL_MODELVIEW31_ARB 0x873F
#define GL_DOT3_RGB_EXT 0x8740
#define GL_DOT3_RGBA_EXT 0x8741
#define GL_MIRROR_CLAMP_ATI 0x8742
#define GL_MIRROR_CLAMP_EXT 0x8742
#define GL_MIRROR_CLAMP_TO_EDGE_ATI 0x8743
#define GL_MIRROR_CLAMP_TO_EDGE_EXT 0x8743
#define GL_MODULATE_ADD_ATI 0x8744
#define GL_MODULATE_SIGNED_ADD_ATI 0x8745
#define GL_MODULATE_SUBTRACT_ATI 0x8746
#define GL_STATIC_ATI 0x8760
#define GL_DYNAMIC_ATI 0x8761
#define GL_PRESERVE_ATI 0x8762
#define GL_DISCARD_ATI 0x8763
#define GL_BUFFER_SIZE 0x8764
#define GL_BUFFER_SIZE_ARB 0x8764
#define GL_OBJECT_BUFFER_SIZE_ATI 0x8764
#define GL_BUFFER_USAGE 0x8765
#define GL_BUFFER_USAGE_ARB 0x8765
#define GL_OBJECT_BUFFER_USAGE_ATI 0x8765
#define GL_ARRAY_OBJECT_BUFFER_ATI 0x8766
#define GL_ARRAY_OBJECT_OFFSET_ATI 0x8767
#define GL_ELEMENT_ARRAY_ATI 0x8768
#define GL_ELEMENT_ARRAY_TYPE_ATI 0x8769
#define GL_ELEMENT_ARRAY_POINTER_ATI 0x876A
#define GL_MAX_VERTEX_STREAMS_ATI 0x876B
#define GL_VERTEX_STREAM0_ATI 0x876C
#define GL_VERTEX_STREAM1_ATI 0x876D
#define GL_VERTEX_STREAM2_ATI 0x876E
#define GL_VERTEX_STREAM3_ATI 0x876F
#define GL_VERTEX_STREAM4_ATI 0x8770
#define GL_VERTEX_STREAM5_ATI 0x8771
#define GL_VERTEX_STREAM6_ATI 0x8772
#define GL_VERTEX_STREAM7_ATI 0x8773
#define GL_VERTEX_SOURCE_ATI 0x8774
#define GL_BUMP_ROT_MATRIX_ATI 0x8775
#define GL_BUMP_ROT_MATRIX_SIZE_ATI 0x8776
#define GL_BUMP_NUM_TEX_UNITS_ATI 0x8777
#define GL_BUMP_TEX_UNITS_ATI 0x8778
#define GL_DUDV_ATI 0x8779
#define GL_DU8DV8_ATI 0x877A
#define GL_BUMP_ENVMAP_ATI 0x877B
#define GL_BUMP_TARGET_ATI 0x877C
#define GL_VERTEX_SHADER_EXT 0x8780
#define GL_VERTEX_SHADER_BINDING_EXT 0x8781
#define GL_OP_INDEX_EXT 0x8782
#define GL_OP_NEGATE_EXT 0x8783
#define GL_OP_DOT3_EXT 0x8784
#define GL_OP_DOT4_EXT 0x8785
#define GL_OP_MUL_EXT 0x8786
#define GL_OP_ADD_EXT 0x8787
#define GL_OP_MADD_EXT 0x8788
#define GL_OP_FRAC_EXT 0x8789

```



```

#define GL_OP_MAX_EXT      0x878A
#define GL_OP_MIN_EXT      0x878B
#define GL_OP_SET_GE_EXT    0x878C
#define GL_OP_SET_LT_EXT    0x878D
#define GL_OP_CLAMP_EXT    0x878E
#define GL_OP_FLOOR_EXT    0x878F
#define GL_OP_ROUND_EXT    0x8790
#define GL_OP_EXP_BASE_2_EXT 0x8791
#define GL_OP_LOG_BASE_2_EXT 0x8792
#define GL_OP_POWER_EXT    0x8793
#define GL_OP_RECIP_EXT    0x8794
#define GL_OP_RECIP_SQRT_EXT 0x8795
#define GL_OP_SUB_EXT      0x8796
#define GL_OP_CROSS_PRODUCT_EXT 0x8797
#define GL_OP_MULTIPLY_MATRIX_EXT 0x8798
#define GL_OP_MOV_EXT      0x8799
#define GL_OUTPUT_VERTEX_EXT 0x879A
#define GL_OUTPUT_COLOR0_EXT 0x879B
#define GL_OUTPUT_COLOR1_EXT 0x879C
#define GL_OUTPUT_TEXTURE_COORD0_EXT 0x879D
#define GL_OUTPUT_TEXTURE_COORD1_EXT 0x879E
#define GL_OUTPUT_TEXTURE_COORD2_EXT 0x879F
#define GL_OUTPUT_TEXTURE_COORD3_EXT 0x87A0
#define GL_OUTPUT_TEXTURE_COORD4_EXT 0x87A1
#define GL_OUTPUT_TEXTURE_COORD5_EXT 0x87A2
#define GL_OUTPUT_TEXTURE_COORD6_EXT 0x87A3
#define GL_OUTPUT_TEXTURE_COORD7_EXT 0x87A4
#define GL_OUTPUT_TEXTURE_COORD8_EXT 0x87A5
#define GL_OUTPUT_TEXTURE_COORD9_EXT 0x87A6
#define GL_OUTPUT_TEXTURE_COORD10_EXT 0x87A7
#define GL_OUTPUT_TEXTURE_COORD11_EXT 0x87A8
#define GL_OUTPUT_TEXTURE_COORD12_EXT 0x87A9
#define GL_OUTPUT_TEXTURE_COORD13_EXT 0x87AA
#define GL_OUTPUT_TEXTURE_COORD14_EXT 0x87AB
#define GL_OUTPUT_TEXTURE_COORD15_EXT 0x87AC
#define GL_OUTPUT_TEXTURE_COORD16_EXT 0x87AD
#define GL_OUTPUT_TEXTURE_COORD17_EXT 0x87AE
#define GL_OUTPUT_TEXTURE_COORD18_EXT 0x87AF
#define GL_OUTPUT_TEXTURE_COORD19_EXT 0x87B0
#define GL_OUTPUT_TEXTURE_COORD20_EXT 0x87B1
#define GL_OUTPUT_TEXTURE_COORD21_EXT 0x87B2
#define GL_OUTPUT_TEXTURE_COORD22_EXT 0x87B3
#define GL_OUTPUT_TEXTURE_COORD23_EXT 0x87B4
#define GL_OUTPUT_TEXTURE_COORD24_EXT 0x87B5
#define GL_OUTPUT_TEXTURE_COORD25_EXT 0x87B6
#define GL_OUTPUT_TEXTURE_COORD26_EXT 0x87B7
#define GL_OUTPUT_TEXTURE_COORD27_EXT 0x87B8
#define GL_OUTPUT_TEXTURE_COORD28_EXT 0x87B9
#define GL_OUTPUT_TEXTURE_COORD29_EXT 0x87BA
#define GL_OUTPUT_TEXTURE_COORD30_EXT 0x87BB
#define GL_OUTPUT_TEXTURE_COORD31_EXT 0x87BC
#define GL_OUTPUT_FOG_EXT    0x87BD
#define GL_SCALAR_EXT        0x87BE
#define GL_VECTOR_EXT        0x87BF
#define GL_MATRIX_EXT        0x87C0
#define GL_VARIANT_EXT       0x87C1
#define GL_INVARIANT_EXT     0x87C2
#define GL_LOCAL_CONSTANT_EXT 0x87C3
#define GL_LOCAL_EXT         0x87C4
#define GL_MAX_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87C5
#define GL_MAX_VERTEX_SHADER_VARIANTS_EXT 0x87C6
#define GL_MAX_VERTEX_SHADER_INVARIANTS_EXT 0x87C7
#define GL_MAX_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87C8
#define GL_MAX_VERTEX_SHADER_LOCALS_EXT 0x87C9
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87CA
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_VARIANTS_EXT 0x87CB

```

```

#define GL_MAX_OPTIMIZED_VERTEX_SHADER_LOCAL_CONSTANTS_EXT
0x87CC
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_INVARIANTS_EXT 0x87CD
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_LOCALS_EXT 0x87CE
#define GL_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87CF
#define GL_VERTEX_SHADER_VARIANTS_EXT 0x87D0
#define GL_VERTEX_SHADER_INVARIANTS_EXT 0x87D1
#define GL_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87D2
#define GL_VERTEX_SHADER_LOCALS_EXT 0x87D3
#define GL_VERTEX_SHADER_OPTIMIZED_EXT 0x87D4
#define GL_X_EXT 0x87D5
#define GL_Y_EXT 0x87D6
#define GL_Z_EXT 0x87D7
#define GL_W_EXT 0x87D8
#define GL_NEGATIVE_X_EXT 0x87D9
#define GL_NEGATIVE_Y_EXT 0x87DA
#define GL_NEGATIVE_Z_EXT 0x87DB
#define GL_NEGATIVE_W_EXT 0x87DC
#define GL_ZERO_EXT 0x87DD
#define GL_ONE_EXT 0x87DE
#define GL_NEGATIVE_ONE_EXT 0x87DF
#define GL_NORMALIZED_RANGE_EXT 0x87E0
#define GL_FULL_RANGE_EXT 0x87E1
#define GL_CURRENT_VERTEX_EXT 0x87E2
#define GL_MVP_MATRIX_EXT 0x87E3
#define GL_VARIANT_VALUE_EXT 0x87E4
#define GL_VARIANT_DATATYPE_EXT 0x87E5
#define GL_VARIANT_ARRAY_STRIDE_EXT 0x87E6
#define GL_VARIANT_ARRAY_TYPE_EXT 0x87E7
#define GL_VARIANT_ARRAY_EXT 0x87E8
#define GL_VARIANT_ARRAY_POINTER_EXT 0x87E9
#define GL_INVARIANT_VALUE_EXT 0x87EA
#define GL_INVARIANT_DATATYPE_EXT 0x87EB
#define GL_LOCAL_CONSTANT_VALUE_EXT 0x87EC
#define GL_LOCAL_CONSTANT_DATATYPE_EXT 0x87ED
#define GL_PN_TRIANGLES_ATI 0x87F0
#define GL_MAX_PN_TRIANGLES_TESSELATION_LEVEL_ATI 0x87F1
#define GL_PN_TRIANGLES_POINT_MODE_ATI 0x87F2
#define GL_PN_TRIANGLES_NORMAL_MODE_ATI 0x87F3
#define GL_PN_TRIANGLES_TESSELATION_LEVEL_ATI 0x87F4
#define GL_PN_TRIANGLES_POINT_MODE_LINEAR_ATI 0x87F5
#define GL_PN_TRIANGLES_POINT_MODE_CUBIC_ATI 0x87F6
#define GL_PN_TRIANGLES_NORMAL_MODE_LINEAR_ATI 0x87F7
#define GL_PN_TRIANGLES_NORMAL_MODE_QUADRATIC_ATI 0x87F8
#define GL_STENCIL_BACK_FUNC 0x8800
#define GL_STENCIL_BACK_FUNC_ATI 0x8800
#define GL_STENCIL_BACK_FAIL 0x8801
#define GL_STENCIL_BACK_FAIL_ATI 0x8801
#define GL_STENCIL_BACK_PASS_DEPTH_FAIL 0x8802
#define GL_STENCIL_BACK_PASS_DEPTH_FAIL_ATI 0x8802
#define GL_STENCIL_BACK_PASS_DEPTH_PASS 0x8803
#define GL_STENCIL_BACK_PASS_DEPTH_PASS_ATI 0x8803
#define GL_FRAGMENT_PROGRAM_ARB 0x8804
#define GL_PROGRAM_ALU_INSTRUCTIONS_ARB 0x8805
#define GL_PROGRAM_TEX_INSTRUCTIONS_ARB 0x8806
#define GL_PROGRAM_TEX_INDIRECTIONS_ARB 0x8807
#define GL_PROGRAM_NATIVE_ALU_INSTRUCTIONS_ARB 0x8808
#define GL_PROGRAM_NATIVE_TEX_INSTRUCTIONS_ARB 0x8809
#define GL_PROGRAM_NATIVE_TEX_INDIRECTIONS_ARB 0x880A
#define GL_MAX_PROGRAM_ALU_INSTRUCTIONS_ARB 0x880B
#define GL_MAX_PROGRAM_TEX_INSTRUCTIONS_ARB 0x880C
#define GL_MAX_PROGRAM_TEX_INDIRECTIONS_ARB 0x880D
#define GL_MAX_PROGRAM_NATIVE_ALU_INSTRUCTIONS_ARB 0x880E
#define GL_MAX_PROGRAM_NATIVE_TEX_INSTRUCTIONS_ARB 0x880F
#define GL_MAX_PROGRAM_NATIVE_TEX_INDIRECTIONS_ARB 0x8810
#define GL_RGBA32F_ARB 0x8814

```



```

#define GL_RGBA_FLOAT32_ATI      0x8814
#define GL_RGB32F_ARB            0x8815
#define GL_RGB_FLOAT32_ATI      0x8815
#define GL_ALPHA32F_ARB          0x8816
#define GL_ALPHA_FLOAT32_ATI    0x8816
#define GL_INTENSITY32F_ARB      0x8817
#define GL_INTENSITY_FLOAT32_ATI 0x8817
#define GL_LUMINANCE32F_ARB      0x8818
#define GL_LUMINANCE_FLOAT32_ATI 0x8818
#define GL_LUMINANCE_ALPHA32F_ARB 0x8819
#define GL_LUMINANCE_ALPHA_FLOAT32_ATI 0x8819
#define GL_RGBA16F_ARB           0x881A
#define GL_RGBA_FLOAT16_ATI      0x881A
#define GL_RGB16F_ARB            0x881B
#define GL_RGB_FLOAT16_ATI       0x881B
#define GL_ALPHA16F_ARB          0x881C
#define GL_ALPHA_FLOAT16_ATI     0x881C
#define GL_INTENSITY16F_ARB      0x881D
#define GL_INTENSITY_FLOAT16_ATI 0x881D
#define GL_LUMINANCE16F_ARB      0x881E
#define GL_LUMINANCE_FLOAT16_ATI 0x881E
#define GL_LUMINANCE_ALPHA16F_ARB 0x881F
#define GL_LUMINANCE_ALPHA_FLOAT16_ATI 0x881F
#define GL_RGBA_FLOAT_MODE_ARB   0x8820
#define GL_TYPE_RGBA_FLOAT_ATI   0x8820
#define GL_MAX_DRAW_BUFFERS      0x8824
#define GL_MAX_DRAW_BUFFERS_ARB  0x8824
#define GL_MAX_DRAW_BUFFERS_ATI  0x8824
#define GL_DRAW_BUFFER0           0x8825
#define GL_DRAW_BUFFER0_ARB       0x8825
#define GL_DRAW_BUFFER0_ATI       0x8825
#define GL_DRAW_BUFFER1           0x8826
#define GL_DRAW_BUFFER1_ARB       0x8826
#define GL_DRAW_BUFFER1_ATI       0x8826
#define GL_DRAW_BUFFER2           0x8827
#define GL_DRAW_BUFFER2_ARB       0x8827
#define GL_DRAW_BUFFER2_ATI       0x8827
#define GL_DRAW_BUFFER3           0x8828
#define GL_DRAW_BUFFER3_ARB       0x8828
#define GL_DRAW_BUFFER3_ATI       0x8828
#define GL_DRAW_BUFFER4           0x8829
#define GL_DRAW_BUFFER4_ARB       0x8829
#define GL_DRAW_BUFFER4_ATI       0x8829
#define GL_DRAW_BUFFER5           0x882A
#define GL_DRAW_BUFFER5_ARB       0x882A
#define GL_DRAW_BUFFER5_ATI       0x882A
#define GL_DRAW_BUFFER6           0x882B
#define GL_DRAW_BUFFER6_ARB       0x882B
#define GL_DRAW_BUFFER6_ATI       0x882B
#define GL_DRAW_BUFFER7           0x882C
#define GL_DRAW_BUFFER7_ARB       0x882C
#define GL_DRAW_BUFFER7_ATI       0x882C
#define GL_DRAW_BUFFER8           0x882D
#define GL_DRAW_BUFFER8_ARB       0x882D
#define GL_DRAW_BUFFER8_ATI       0x882D
#define GL_DRAW_BUFFER9           0x882E
#define GL_DRAW_BUFFER9_ARB       0x882E
#define GL_DRAW_BUFFER9_ATI       0x882E
#define GL_DRAW_BUFFER10          0x882F
#define GL_DRAW_BUFFER10_ARB      0x882F
#define GL_DRAW_BUFFER10_ATI      0x882F
#define GL_DRAW_BUFFER11          0x8830
#define GL_DRAW_BUFFER11_ARB      0x8830
#define GL_DRAW_BUFFER11_ATI      0x8830
#define GL_DRAW_BUFFER12          0x8831
#define GL_DRAW_BUFFER12_ARB      0x8831

```

```

#define GL_DRAW_BUFFER12_ATI 0x8831
#define GL_DRAW_BUFFER13 0x8832
#define GL_DRAW_BUFFER13_ARB 0x8832
#define GL_DRAW_BUFFER13_ATI 0x8832
#define GL_DRAW_BUFFER14 0x8833
#define GL_DRAW_BUFFER14_ARB 0x8833
#define GL_DRAW_BUFFER14_ATI 0x8833
#define GL_DRAW_BUFFER15 0x8834
#define GL_DRAW_BUFFER15_ARB 0x8834
#define GL_DRAW_BUFFER15_ATI 0x8834
#define GL_COLOR_CLEAR_UNCLAMPED_VALUE_ATI 0x8835
#define GL_BLEND_EQUATION_ALPHA 0x883D
#define GL_BLEND_EQUATION_ALPHA_EXT 0x883D
#define GL_MATRIX_PALETTE_ARB 0x8840
#define GL_MAX_MATRIX_PALETTE_STACK_DEPTH_ARB 0x8841
#define GL_MAX_PALETTE_MATRICES_ARB 0x8842
#define GL_CURRENT_PALETTE_MATRIX_ARB 0x8843
#define GL_MATRIX_INDEX_ARRAY_ARB 0x8844
#define GL_CURRENT_MATRIX_INDEX_ARB 0x8845
#define GL_MATRIX_INDEX_ARRAY_SIZE_ARB 0x8846
#define GL_MATRIX_INDEX_ARRAY_TYPE_ARB 0x8847
#define GL_MATRIX_INDEX_ARRAY_STRIDE_ARB 0x8848
#define GL_MATRIX_INDEX_ARRAY_POINTER_ARB 0x8849
#define GL_TEXTURE_DEPTH_SIZE 0x884A
#define GL_TEXTURE_DEPTH_SIZE_ARB 0x884A
#define GL_DEPTH_TEXTURE_MODE 0x884B
#define GL_DEPTH_TEXTURE_MODE_ARB 0x884B
#define GL_TEXTURE_COMPARE_MODE 0x884C
#define GL_TEXTURE_COMPARE_MODE_ARB 0x884C
#define GL_TEXTURE_COMPARE_FUNC 0x884D
#define GL_TEXTURE_COMPARE_FUNC_ARB 0x884D
#define GL_COMPARE_R_TO_TEXTURE 0x884E
#define GL_COMPARE_R_TO_TEXTURE_ARB 0x884E
#define GL_POINT_SPRITE 0x8861
#define GL_POINT_SPRITE_ARB 0x8861
#define GL_COORD_REPLACE 0x8862
#define GL_COORD_REPLACE_ARB 0x8862
#define GL_QUERY_COUNTER_BITS 0x8864
#define GL_QUERY_COUNTER_BITS_ARB 0x8864
#define GL_CURRENT_QUERY 0x8865
#define GL_CURRENT_QUERY_ARB 0x8865
#define GL_QUERY_RESULT 0x8866
#define GL_QUERY_RESULT_ARB 0x8866
#define GL_QUERY_RESULT_AVAILABLE 0x8867
#define GL_QUERY_RESULT_AVAILABLE_ARB 0x8867
#define GL_MAX_VERTEX_ATTRIBS 0x8869
#define GL_MAX_VERTEX_ATTRIBS_ARB 0x8869
#define GL_VERTEX_ATTRIB_ARRAY_NORMALIZED 0x886A
#define GL_VERTEX_ATTRIB_ARRAY_NORMALIZED_ARB 0x886A
#define GL_MAX_TEXTURE_COORDS 0x8871
#define GL_MAX_TEXTURE_COORDS_ARB 0x8871
#define GL_MAX_TEXTURE_IMAGE_UNITS 0x8872
#define GL_MAX_TEXTURE_IMAGE_UNITS_ARB 0x8872
#define GL_PROGRAM_ERROR_STRING_ARB 0x8874
#define GL_PROGRAM_FORMAT_ASCII_ARB 0x8875
#define GL_PROGRAM_FORMAT_ARB 0x8876
#define GL_DEPTH_BOUNDS_TEST_EXT 0x8890
#define GL_DEPTH_BOUNDS_EXT 0x8891
#define GL_ARRAY_BUFFER 0x8892
#define GL_ARRAY_BUFFER_ARB 0x8892
#define GL_ELEMENT_ARRAY_BUFFER 0x8893
#define GL_ELEMENT_ARRAY_BUFFER_ARB 0x8893
#define GL_ARRAY_BUFFER_BINDING 0x8894
#define GL_ARRAY_BUFFER_BINDING_ARB 0x8894
#define GL_ELEMENT_ARRAY_BUFFER_BINDING 0x8895
#define GL_ELEMENT_ARRAY_BUFFER_BINDING_ARB 0x8895

```

```

#define GL_VERTEX_ARRAY_BUFFER_BINDING 0x8896
#define GL_VERTEX_ARRAY_BUFFER_BINDING_ARB 0x8896
#define GL_NORMAL_ARRAY_BUFFER_BINDING 0x8897
#define GL_NORMAL_ARRAY_BUFFER_BINDING_ARB 0x8897
#define GL_COLOR_ARRAY_BUFFER_BINDING 0x8898
#define GL_COLOR_ARRAY_BUFFER_BINDING_ARB 0x8898
#define GL_INDEX_ARRAY_BUFFER_BINDING 0x8899
#define GL_INDEX_ARRAY_BUFFER_BINDING_ARB 0x8899
#define GL_TEXTURE_COORD_ARRAY_BUFFER_BINDING 0x889A
#define GL_TEXTURE_COORD_ARRAY_BUFFER_BINDING_ARB 0x889A
#define GL_EDGE_FLAG_ARRAY_BUFFER_BINDING 0x889B
#define GL_EDGE_FLAG_ARRAY_BUFFER_BINDING_ARB 0x889B
#define GL_SECONDARY_COLOR_ARRAY_BUFFER_BINDING 0x889C
#define GL_SECONDARY_COLOR_ARRAY_BUFFER_BINDING_ARB 0x889C
#define GL_FOG_COORDINATE_ARRAY_BUFFER_BINDING 0x889D
#define GL_FOG_COORDINATE_ARRAY_BUFFER_BINDING_ARB 0x889D
#define GL_FOG_COORD_ARRAY_BUFFER_BINDING 0x889D
#define GL_WEIGHT_ARRAY_BUFFER_BINDING 0x889E
#define GL_WEIGHT_ARRAY_BUFFER_BINDING_ARB 0x889E
#define GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING 0x889F
#define GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING_ARB 0x889F
#define GL_PROGRAM_INSTRUCTIONS_ARB 0x88A0
#define GL_MAX_PROGRAM_INSTRUCTIONS_ARB 0x88A1
#define GL_PROGRAM_NATIVE_INSTRUCTIONS_ARB 0x88A2
#define GL_MAX_PROGRAM_NATIVE_INSTRUCTIONS_ARB 0x88A3
#define GL_PROGRAM_TEMPORARIES_ARB 0x88A4
#define GL_MAX_PROGRAM_TEMPORARIES_ARB 0x88A5
#define GL_PROGRAM_NATIVE_TEMPORARIES_ARB 0x88A6
#define GL_MAX_PROGRAM_NATIVE_TEMPORARIES_ARB 0x88A7
#define GL_PROGRAM_PARAMETERS_ARB 0x88A8
#define GL_MAX_PROGRAM_PARAMETERS_ARB 0x88A9
#define GL_PROGRAM_NATIVE_PARAMETERS_ARB 0x88AA
#define GL_MAX_PROGRAM_NATIVE_PARAMETERS_ARB 0x88AB
#define GL_PROGRAM_ATTRIBS_ARB 0x88AC
#define GL_MAX_PROGRAM_ATTRIBS_ARB 0x88AD
#define GL_PROGRAM_NATIVE_ATTRIBS_ARB 0x88AE
#define GL_MAX_PROGRAM_NATIVE_ATTRIBS_ARB 0x88AF
#define GL_PROGRAM_ADDRESS_REGISTERS_ARB 0x88B0
#define GL_MAX_PROGRAM_ADDRESS_REGISTERS_ARB 0x88B1
#define GL_PROGRAM_NATIVE_ADDRESS_REGISTERS_ARB 0x88B2
#define GL_MAX_PROGRAM_NATIVE_ADDRESS_REGISTERS_ARB 0x88B3
#define GL_MAX_PROGRAM_LOCAL_PARAMETERS_ARB 0x88B4
#define GL_MAX_PROGRAM_ENV_PARAMETERS_ARB 0x88B5
#define GL_PROGRAM_UNDER_NATIVE_LIMITS_ARB 0x88B6
#define GL_TRANSPOSE_CURRENT_MATRIX_ARB 0x88B7
#define GL_READ_ONLY 0x88B8
#define GL_READ_ONLY_ARB 0x88B8
#define GL_WRITE_ONLY 0x88B9
#define GL_WRITE_ONLY_ARB 0x88B9
#define GL_READ_WRITE 0x88BA
#define GL_READ_WRITE_ARB 0x88BA
#define GL_BUFFER_ACCESS 0x88BB
#define GL_BUFFER_ACCESS_ARB 0x88BB
#define GL_BUFFER_MAPPED 0x88BC
#define GL_BUFFER_MAPPED_ARB 0x88BC
#define GL_BUFFER_MAP_POINTER 0x88BD
#define GL_BUFFER_MAP_POINTER_ARB 0x88BD
#define GL_MATRIX0_ARB 0x88C0
#define GL_MATRIX1_ARB 0x88C1
#define GL_MATRIX2_ARB 0x88C2
#define GL_MATRIX3_ARB 0x88C3
#define GL_MATRIX4_ARB 0x88C4
#define GL_MATRIX5_ARB 0x88C5
#define GL_MATRIX6_ARB 0x88C6
#define GL_MATRIX7_ARB 0x88C7
#define GL_MATRIX8_ARB 0x88C8

```

```

#define GL_MATRIX9_ARB 0x88C9
#define GL_MATRIX10_ARB 0x88CA
#define GL_MATRIX11_ARB 0x88CB
#define GL_MATRIX12_ARB 0x88CC
#define GL_MATRIX13_ARB 0x88CD
#define GL_MATRIX14_ARB 0x88CE
#define GL_MATRIX15_ARB 0x88CF
#define GL_MATRIX16_ARB 0x88D0
#define GL_MATRIX17_ARB 0x88D1
#define GL_MATRIX18_ARB 0x88D2
#define GL_MATRIX19_ARB 0x88D3
#define GL_MATRIX20_ARB 0x88D4
#define GL_MATRIX21_ARB 0x88D5
#define GL_MATRIX22_ARB 0x88D6
#define GL_MATRIX23_ARB 0x88D7
#define GL_MATRIX24_ARB 0x88D8
#define GL_MATRIX25_ARB 0x88D9
#define GL_MATRIX26_ARB 0x88DA
#define GL_MATRIX27_ARB 0x88DB
#define GL_MATRIX28_ARB 0x88DC
#define GL_MATRIX29_ARB 0x88DD
#define GL_MATRIX30_ARB 0x88DE
#define GL_MATRIX31_ARB 0x88DF
#define GL_STREAM_DRAW 0x88E0
#define GL_STREAM_DRAW_ARB 0x88E0
#define GL_STREAM_READ 0x88E1
#define GL_STREAM_READ_ARB 0x88E1
#define GL_STREAM_COPY 0x88E2
#define GL_STREAM_COPY_ARB 0x88E2
#define GL_STATIC_DRAW 0x88E4
#define GL_STATIC_DRAW_ARB 0x88E4
#define GL_STATIC_READ 0x88E5
#define GL_STATIC_READ_ARB 0x88E5
#define GL_STATIC_COPY 0x88E6
#define GL_STATIC_COPY_ARB 0x88E6
#define GL_DYNAMIC_DRAW 0x88E8
#define GL_DYNAMIC_DRAW_ARB 0x88E8
#define GL_DYNAMIC_READ 0x88E9
#define GL_DYNAMIC_READ_ARB 0x88E9
#define GL_DYNAMIC_COPY 0x88EA
#define GL_DYNAMIC_COPY_ARB 0x88EA
#define GL_PIXEL_PACK_BUFFER 0x88EB
#define GL_PIXEL_PACK_BUFFER_ARB 0x88EB
#define GL_PIXEL_PACK_BUFFER_EXT 0x88EB
#define GL_PIXEL_UNPACK_BUFFER 0x88EC
#define GL_PIXEL_UNPACK_BUFFER_ARB 0x88EC
#define GL_PIXEL_UNPACK_BUFFER_EXT 0x88EC
#define GL_PIXEL_PACK_BUFFER_BINDING 0x88ED
#define GL_PIXEL_PACK_BUFFER_BINDING_ARB 0x88ED
#define GL_PIXEL_PACK_BUFFER_BINDING_EXT 0x88ED
#define GL_PIXEL_UNPACK_BUFFER_BINDING 0x88EF
#define GL_PIXEL_UNPACK_BUFFER_BINDING_ARB 0x88EF
#define GL_PIXEL_UNPACK_BUFFER_BINDING_EXT 0x88EF
#define GL_STENCIL_TEST_TWO_SIDE_EXT 0x8910
#define GL_ACTIVE_STENCIL_FACE_EXT 0x8911
#define GL_MIRROR_CLAMP_TO_BORDER_EXT 0x8912
#define GL_SAMPLES_PASSED 0x8914
#define GL_SAMPLES_PASSED_ARB 0x8914
#define GL_CLAMP_VERTEX_COLOR_ARB 0x891A
#define GL_CLAMP_FRAGMENT_COLOR_ARB 0x891B
#define GL_CLAMP_READ_COLOR_ARB 0x891C
#define GL_FIXED_ONLY_ARB 0x891D
#define GL_FRAGMENT_SHADER_ATI 0x8920
#define GL_REG_0_ATI 0x8921
#define GL_REG_1_ATI 0x8922
#define GL_REG_2_ATI 0x8923

```

```

#define GL_REG_3_ATI 0x8924
#define GL_REG_4_ATI 0x8925
#define GL_REG_5_ATI 0x8926
#define GL_REG_6_ATI 0x8927
#define GL_REG_7_ATI 0x8928
#define GL_REG_8_ATI 0x8929
#define GL_REG_9_ATI 0x892A
#define GL_REG_10_ATI 0x892B
#define GL_REG_11_ATI 0x892C
#define GL_REG_12_ATI 0x892D
#define GL_REG_13_ATI 0x892E
#define GL_REG_14_ATI 0x892F
#define GL_REG_15_ATI 0x8930
#define GL_REG_16_ATI 0x8931
#define GL_REG_17_ATI 0x8932
#define GL_REG_18_ATI 0x8933
#define GL_REG_19_ATI 0x8934
#define GL_REG_20_ATI 0x8935
#define GL_REG_21_ATI 0x8936
#define GL_REG_22_ATI 0x8937
#define GL_REG_23_ATI 0x8938
#define GL_REG_24_ATI 0x8939
#define GL_REG_25_ATI 0x893A
#define GL_REG_26_ATI 0x893B
#define GL_REG_27_ATI 0x893C
#define GL_REG_28_ATI 0x893D
#define GL_REG_29_ATI 0x893E
#define GL_REG_30_ATI 0x893F
#define GL_REG_31_ATI 0x8940
#define GL_CON_0_ATI 0x8941
#define GL_CON_1_ATI 0x8942
#define GL_CON_2_ATI 0x8943
#define GL_CON_3_ATI 0x8944
#define GL_CON_4_ATI 0x8945
#define GL_CON_5_ATI 0x8946
#define GL_CON_6_ATI 0x8947
#define GL_CON_7_ATI 0x8948
#define GL_CON_8_ATI 0x8949
#define GL_CON_9_ATI 0x894A
#define GL_CON_10_ATI 0x894B
#define GL_CON_11_ATI 0x894C
#define GL_CON_12_ATI 0x894D
#define GL_CON_13_ATI 0x894E
#define GL_CON_14_ATI 0x894F
#define GL_CON_15_ATI 0x8950
#define GL_CON_16_ATI 0x8951
#define GL_CON_17_ATI 0x8952
#define GL_CON_18_ATI 0x8953
#define GL_CON_19_ATI 0x8954
#define GL_CON_20_ATI 0x8955
#define GL_CON_21_ATI 0x8956
#define GL_CON_22_ATI 0x8957
#define GL_CON_23_ATI 0x8958
#define GL_CON_24_ATI 0x8959
#define GL_CON_25_ATI 0x895A
#define GL_CON_26_ATI 0x895B
#define GL_CON_27_ATI 0x895C
#define GL_CON_28_ATI 0x895D
#define GL_CON_29_ATI 0x895E
#define GL_CON_30_ATI 0x895F
#define GL_CON_31_ATI 0x8960
#define GL_MOV_ATI 0x8961
#define GL_ADD_ATI 0x8963
#define GL_MUL_ATI 0x8964
#define GL_SUB_ATI 0x8965
#define GL_DOT3_ATI 0x8966

```

```

#define GL_DOT4_ATI 0x8967
#define GL_MAD_ATI 0x8968
#define GL_LERP_ATI 0x8969
#define GL_CND_ATI 0x896A
#define GL_CND0_ATI 0x896B
#define GL_DOT2_ADD_ATI 0x896C
#define GL_SECONDARY_INTERPOLATOR_ATI 0x896D
#define GL_NUM_FRAGMENT_REGISTERS_ATI 0x896E
#define GL_NUM_FRAGMENT_CONSTANTS_ATI 0x896F
#define GL_NUM_PASSES_ATI 0x8970
#define GL_NUM_INSTRUCTIONS_PER_PASS_ATI 0x8971
#define GL_NUM_INSTRUCTIONS_TOTAL_ATI 0x8972
#define GL_NUM_INPUT_INTERPOLATOR_COMPONENTS_ATI 0x8973
#define GL_NUM_LOOPBACK_COMPONENTS_ATI 0x8974
#define GL_COLOR_ALPHA_PAIRING_ATI 0x8975
#define GL_SWIZZLE_STR_ATI 0x8976
#define GL_SWIZZLE_STQ_ATI 0x8977
#define GL_SWIZZLE_STR_DR_ATI 0x8978
#define GL_SWIZZLE_STQ_DQ_ATI 0x8979
#define GL_SWIZZLE_STRQ_ATI 0x897A
#define GL_SWIZZLE_STRQ_DQ_ATI 0x897B
#define GL_FRAGMENT_SHADER 0x8B30
#define GL_FRAGMENT_SHADER_ARB 0x8B30
#define GL_VERTEX_SHADER 0x8B31
#define GL_VERTEX_SHADER_ARB 0x8B31
#define GL_PROGRAM_OBJECT_ARB 0x8B40
#define GL_SHADER_OBJECT_ARB 0x8B48
#define GL_MAX_FRAGMENT_UNIFORM_COMPONENTS 0x8B49
#define GL_MAX_FRAGMENT_UNIFORM_COMPONENTS_ARB 0x8B49
#define GL_MAX_VERTEX_UNIFORM_COMPONENTS 0x8B4A
#define GL_MAX_VERTEX_UNIFORM_COMPONENTS_ARB 0x8B4A
#define GL_MAX_VARYING_FLOATS 0x8B4B
#define GL_MAX_VARYING_FLOATS_ARB 0x8B4B
#define GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS 0x8B4C
#define GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS_ARB 0x8B4C
#define GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS 0x8B4D
#define GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS_ARB 0x8B4D
#define GL_OBJECT_TYPE_ARB 0x8B4E
#define GL_OBJECT_SUBTYPE_ARB 0x8B4F
#define GL_SHADER_TYPE 0x8B4F
#define GL_FLOAT_VEC2 0x8B50
#define GL_FLOAT_VEC2_ARB 0x8B50
#define GL_FLOAT_VEC3 0x8B51
#define GL_FLOAT_VEC3_ARB 0x8B51
#define GL_FLOAT_VEC4 0x8B52
#define GL_FLOAT_VEC4_ARB 0x8B52
#define GL_INT_VEC2 0x8B53
#define GL_INT_VEC2_ARB 0x8B53
#define GL_INT_VEC3 0x8B54
#define GL_INT_VEC3_ARB 0x8B54
#define GL_INT_VEC4 0x8B55
#define GL_INT_VEC4_ARB 0x8B55
#define GL_BOOL 0x8B56
#define GL_BOOL_ARB 0x8B56
#define GL_BOOL_VEC2 0x8B57
#define GL_BOOL_VEC2_ARB 0x8B57
#define GL_BOOL_VEC3 0x8B58
#define GL_BOOL_VEC3_ARB 0x8B58
#define GL_BOOL_VEC4 0x8B59
#define GL_BOOL_VEC4_ARB 0x8B59
#define GL_FLOAT_MAT2 0x8B5A
#define GL_FLOAT_MAT2_ARB 0x8B5A
#define GL_FLOAT_MAT3 0x8B5B
#define GL_FLOAT_MAT3_ARB 0x8B5B
#define GL_FLOAT_MAT4 0x8B5C
#define GL_FLOAT_MAT4_ARB 0x8B5C

```



```

#define GL_SAMPLER_1D 0x8B5D
#define GL_SAMPLER_1D_ARB 0x8B5D
#define GL_SAMPLER_2D 0x8B5E
#define GL_SAMPLER_2D_ARB 0x8B5E
#define GL_SAMPLER_3D 0x8B5F
#define GL_SAMPLER_3D_ARB 0x8B5F
#define GL_SAMPLER_CUBE 0x8B60
#define GL_SAMPLER_CUBE_ARB 0x8B60
#define GL_SAMPLER_1D_SHADOW 0x8B61
#define GL_SAMPLER_1D_SHADOW_ARB 0x8B61
#define GL_SAMPLER_2D_SHADOW 0x8B62
#define GL_SAMPLER_2D_SHADOW_ARB 0x8B62
#define GL_SAMPLER_2D_RECT_ARB 0x8B63
#define GL_SAMPLER_2D_RECT_SHADOW_ARB 0x8B64
#define GL_FLOAT_MAT2x3 0x8B65
#define GL_FLOAT_MAT2x4 0x8B66
#define GL_FLOAT_MAT3x2 0x8B67
#define GL_FLOAT_MAT3x4 0x8B68
#define GL_FLOAT_MAT4x2 0x8B69
#define GL_FLOAT_MAT4x3 0x8B6A
#define GL_DELETE_STATUS 0x8B80
#define GL_OBJECT_DELETE_STATUS_ARB 0x8B80
#define GL_COMPILE_STATUS 0x8B81
#define GL_OBJECT_COMPILE_STATUS_ARB 0x8B81
#define GL_LINK_STATUS 0x8B82
#define GL_OBJECT_LINK_STATUS_ARB 0x8B82
#define GL_OBJECT_VALIDATE_STATUS_ARB 0x8B83
#define GL_VALIDATE_STATUS 0x8B83
#define GL_INFO_LOG_LENGTH 0x8B84
#define GL_OBJECT_INFO_LOG_LENGTH_ARB 0x8B84
#define GL_ATTACHED_SHADERS 0x8B85
#define GL_OBJECT_ATTACHED_OBJECTS_ARB 0x8B85
#define GL_ACTIVE_UNIFORMS 0x8B86
#define GL_OBJECT_ACTIVE_UNIFORMS_ARB 0x8B86
#define GL_ACTIVE_UNIFORM_MAX_LENGTH 0x8B87
#define GL_OBJECT_ACTIVE_UNIFORM_MAX_LENGTH_ARB 0x8B87
#define GL_OBJECT_SHADER_SOURCE_LENGTH_ARB 0x8B88
#define GL_SHADER_SOURCE_LENGTH 0x8B88
#define GL_ACTIVE_ATTRIBUTES 0x8B89
#define GL_OBJECT_ACTIVE_ATTRIBUTES_ARB 0x8B89
#define GL_ACTIVE_ATTRIBUTE_MAX_LENGTH 0x8B8A
#define GL_OBJECT_ACTIVE_ATTRIBUTE_MAX_LENGTH_ARB 0x8B8A
#define GL_FRAGMENT_SHADER_DERIVATIVE_HINT 0x8B8B
#define GL_FRAGMENT_SHADER_DERIVATIVE_HINT_ARB 0x8B8B
#define GL_SHADING_LANGUAGE_VERSION 0x8B8C
#define GL_SHADING_LANGUAGE_VERSION_ARB 0x8B8C
#define GL_CURRENT_PROGRAM 0x8B8D
#define GL_IMPLEMENTATION_COLOR_READ_TYPE_OES 0x8B9A
#define GL_IMPLEMENTATION_COLOR_READ_FORMAT_OES 0x8B9B
#define GL_TEXTURE_RED_TYPE_ARB 0x8C10
#define GL_TEXTURE_GREEN_TYPE_ARB 0x8C11
#define GL_TEXTURE_BLUE_TYPE_ARB 0x8C12
#define GL_TEXTURE_ALPHA_TYPE_ARB 0x8C13
#define GL_TEXTURE_LUMINANCE_TYPE_ARB 0x8C14
#define GL_TEXTURE_INTENSITY_TYPE_ARB 0x8C15
#define GL_TEXTURE_DEPTH_TYPE_ARB 0x8C16
#define GL_UNSIGNED_NORMALIZED_ARB 0x8C17
#define GL_SRGB 0x8C40
#define GL_SRGB8 0x8C41
#define GL_SRGB_ALPHA 0x8C42
#define GL_SRGB8_ALPHA8 0x8C43
#define GL_SLUMINANCE_ALPHA 0x8C44
#define GL_SLUMINANCE8_ALPHA8 0x8C45
#define GL_SLUMINANCE 0x8C46
#define GL_SLUMINANCE8 0x8C47
#define GL_COMPRESSED_SRGB 0x8C48

```

```

#define GL_COMPRESSED_SRGB_ALPHA          0x8C49
#define GL_COMPRESSED_SLUMINANCE         0x8C4A
#define GL_COMPRESSED_SLUMINANCE_ALPHA  0x8C4B
#define GL_POINT_SPRITE_COORD_ORIGIN     0x8CA0
#define GL_LOWER_LEFT                     0x8CA1
#define GL_UPPER_LEFT                     0x8CA2
#define GL_STENCIL_BACK_REF               0x8CA3
#define GL_STENCIL_BACK_VALUE_MASK       0x8CA4
#define GL_STENCIL_BACK_WRITEMASK        0x8CA5
#define GL_FRAMEBUFFER_BINDING_EXT       0x8CA6
#define GL_RENDERBUFFER_BINDING_EXT      0x8CA7
#define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE_EXT 0x8CD0
#define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_NAME_EXT 0x8CD1
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL_EXT 0x8CD2
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE_EXT 0x8CD3
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_3D_ZOFFSET_EXT 0x8CD4
#define GL_FRAMEBUFFER_COMPLETE_EXT      0x8CD5
#define GL_FRAMEBUFFER_INCOMPLETE_ATTACHMENT_EXT 0x8CD6
#define GL_FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT_EXT 0x8CD7
#define GL_FRAMEBUFFER_INCOMPLETE_DIMENSIONS_EXT 0x8CD9
#define GL_FRAMEBUFFER_INCOMPLETE_FORMATS_EXT 0x8CDA
#define GL_FRAMEBUFFER_INCOMPLETE_DRAW_BUFFER_EXT 0x8CDB
#define GL_FRAMEBUFFER_INCOMPLETE_READ_BUFFER_EXT 0x8CDC
#define GL_FRAMEBUFFER_UNSUPPORTED_EXT   0x8CDD
#define GL_MAX_COLOR_ATTACHMENTS_EXT     0x8CDF
#define GL_COLOR_ATTACHMENT0_EXT         0x8CE0
#define GL_COLOR_ATTACHMENT1_EXT         0x8CE1
#define GL_COLOR_ATTACHMENT2_EXT         0x8CE2
#define GL_COLOR_ATTACHMENT3_EXT         0x8CE3
#define GL_COLOR_ATTACHMENT4_EXT         0x8CE4
#define GL_COLOR_ATTACHMENT5_EXT         0x8CE5
#define GL_COLOR_ATTACHMENT6_EXT         0x8CE6
#define GL_COLOR_ATTACHMENT7_EXT         0x8CE7
#define GL_COLOR_ATTACHMENT8_EXT         0x8CE8
#define GL_COLOR_ATTACHMENT9_EXT         0x8CE9
#define GL_COLOR_ATTACHMENT10_EXT        0x8CEA
#define GL_COLOR_ATTACHMENT11_EXT        0x8CEB
#define GL_COLOR_ATTACHMENT12_EXT        0x8CEC
#define GL_COLOR_ATTACHMENT13_EXT        0x8CED
#define GL_COLOR_ATTACHMENT14_EXT        0x8CEE
#define GL_COLOR_ATTACHMENT15_EXT        0x8CEF
#define GL_DEPTH_ATTACHMENT_EXT          0x8D00
#define GL_STENCIL_ATTACHMENT_EXT         0x8D20
#define GL_FRAMEBUFFER_EXT                0x8D40
#define GL_RENDERBUFFER_EXT               0x8D41
#define GL_RENDERBUFFER_WIDTH_EXT         0x8D42
#define GL_RENDERBUFFER_HEIGHT_EXT        0x8D43
#define GL_RENDERBUFFER_INTERNAL_FORMAT_EXT 0x8D44
#define GL_STENCIL_INDEX1_EXT             0x8D46
#define GL_STENCIL_INDEX4_EXT             0x8D47
#define GL_STENCIL_INDEX8_EXT             0x8D48
#define GL_STENCIL_INDEX16_EXT            0x8D49
#define GL_RENDERBUFFER_RED_SIZE_EXT      0x8D50
#define GL_RENDERBUFFER_GREEN_SIZE_EXT    0x8D51
#define GL_RENDERBUFFER_BLUE_SIZE_EXT     0x8D52
#define GL_RENDERBUFFER_ALPHA_SIZE_EXT    0x8D53
#define GL_RENDERBUFFER_DEPTH_SIZE_EXT    0x8D54
#define GL_RENDERBUFFER_STENCIL_SIZE_EXT  0x8D55
#define GL_3DFX_multisample               1
#define GL_3DFX_tbuffer                   1
#define GL_3DFX_texture_compression_FXT1 1
#define GL_ARB_color_buffer_float          1
#define GL_ARB_depth_texture              1

```



```

#define GL_ARB_draw_buffers 1
#define GL_ARB_fragment_program 1
#define GL_ARB_fragment_program_shadow 1
#define GL_ARB_fragment_shader 1
#define GL_ARB_half_float_pixel 1
#define GL_ARB_matrix_palette 1
#define GL_ARB_multisample 1
#define GL_ARB_occlusion_query 1
#define GL_ARB_pixel_buffer_object 1
#define GL_ARB_point_parameters 1
#define GL_ARB_point_sprite 1
#define GL_ARB_shader_objects 1
#define GL_ARB_shading_language_100 1
#define GL_ARB_shadow 1
#define GL_ARB_shadow_ambient 1
#define GL_ARB_texture_border_clamp 1
#define GL_ARB_texture_compression 1
#define GL_ARB_texture_cube_map 1
#define GL_ARB_texture_env_add 1
#define GL_ARB_texture_env_combine 1
#define GL_ARB_texture_env_crossbar 1
#define GL_ARB_texture_env_dot3 1
#define GL_ARB_texture_float 1
#define GL_ARB_texture_mirrored_repeat 1
#define GL_ARB_texture_non_power_of_two 1
#define GL_ARB_texture_rectangle 1
#define GL_ARB_transpose_matrix 1
#define GL_ARB_vertex_blend 1
#define GL_ARB_vertex_buffer_object 1
#define GL_ARB_vertex_shader 1
#define GL_ARB_window_pos 1
#define GL_ATI_draw_buffers 1
#define GL_ATI_element_array 1
#define GL_ATI_envmap_bumpmap 1
#define GL_ATI_fragment_shader 1
#define GL_ATI_map_object_buffer 1
#define GL_ATI_pixel_format_float 1
#define GL_ATI_pn_triangles 1
#define GL_ATI_separate_stencil 1
#define GL_ATI_text_fragment_shader 1
#define GL_ATI_texture_env_combine3 1
#define GL_ATI_texture_float 1
#define GL_ATI_texture_mirror_once 1
#define GL_ATI_vertex_array_object 1
#define GL_ATI_vertex_attrib_array_object 1
#define GL_ATI_vertex_streams 1
#define GL_EXT_422_pixels 1
#define GL_EXT_abgr 1
#define GL_EXT_bgra 1
#define GL_EXT_blend_color 1
#define GL_EXT_blend_equation_separate 1
#define GL_EXT_blend_func_separate 1
#define GL_EXT_blend_logic_op 1
#define GL_EXT_blend_minmax 1
#define GL_EXT_blend_subtract 1
#define GL_EXT_clip_volume_hint 1
#define GL_EXT_cmyka 1
#define GL_EXT_color_subtable 1
#define GL_EXT_compiled_vertex_array 1
#define GL_EXT_convolution 1
#define GL_EXT_coordinate_frame 1
#define GL_EXT_copy_texture 1
#define GL_EXT_cull_vertex 1
#define GL_EXT_depth_bounds_test 1
#define GL_EXT_draw_range_elements 1
#define GL_EXT_fog_coord 1

```

```

#define GL_EXT_histogram 1
#define GL_EXT_index_array_formats 1
#define GL_EXT_index_func 1
#define GL_EXT_index_material 1
#define GL_EXT_index_texture 1
#define GL_EXT_light_texture 1
#define GL_EXT_misc_attribute 1
#define GL_EXT_multi_draw_arrays 1
#define GL_EXT_multisample 1
#define GL_EXT_packed_pixels 1
#define GL_EXT_paletted_texture 1
#define GL_EXT_pixel_buffer_object 1
#define GL_EXT_pixel_transform 1
#define GL_EXT_pixel_transform_color_table 1
#define GL_EXT_point_parameters 1
#define GL_EXT_polygon_offset 1
#define GL_EXT_rescale_normal 1
#define GL_EXT_secondary_color 1
#define GL_EXT_separate_specular_color 1
#define GL_EXT_shadow_funcs 1
#define GL_EXT_shared_texture_palette 1
#define GL_EXT_stencil_two_side 1
#define GL_EXT_stencil_wrap 1
#define GL_EXT_subtexture 1
#define GL_EXT_texture 1
#define GL_EXT_texture3D 1
#define GL_EXT_texture_env_add 1
#define GL_EXT_texture_env_combine 1
#define GL_EXT_texture_env_dot3 1
#define GL_EXT_texture_filter_anisotropic 1
#define GL_EXT_texture_lod_bias 1
#define GL_EXT_texture_mirror_clamp 1
#define GL_EXT_texture_object 1
#define GL_EXT_texture_perturb_normal 1
#define GL_EXT_vertex_array 1
#define GL_EXT_vertex_shader 1
#define GL_EXT_vertex_weighting 1
#define GL_GREMEDY_string_marker 1
#define GL_HP_convolution_border_modes 1
#define GL_HP_image_transform 1
#define GL_HP_occlusion_test 1
#define GL_HP_texture_lighting 1
#define GL_IBM_cull_vertex 1
#define GL_IBM_multimode_draw_arrays 1
#define GL_IBM_rasterpos_clip 1
#define GL_IBM_vertex_array_lists 1
#define GL_INGR_blend_func_separate 1
#define GL_INGR_color_clamp 1
#define GL_INGR_interlace_read 1
#define GL_INTEL_parallel_arrays 1
#define GL_OES_read_format 1
#define GL_PGI_misc_hints 1
#define GL_PGI_vertex_hints 1
#define GL_REND_screen_coordinates 1
#define GL_S3_s3tc 1
#define GL_SUNX_constant_data 1
#define GL_SUN_convolution_border_modes 1
#define GL_SUN_global_alpha 1
#define GL_SUN_mesh_array 1
#define GL_SUN_slice_accum 1
#define GL_SUN_triangle_list 1
#define GL_SUN_vertex 1
#define GL_VERSION_1_4 1
#define GL_VERSION_1_5 1
#define GL_VERSION_2_0 1
#define GL_VERSION_2_1 1

```

```

#define GL_WIN_phong_shading      1
#define GL_WIN_specular_fog      1
#define GL_CULL_VERTEX_IBM        103050
#define GL_VERTEX_ARRAY_LIST_IBM  103070
#define GL_NORMAL_ARRAY_LIST_IBM  103071
#define GL_COLOR_ARRAY_LIST_IBM   103072
#define GL_INDEX_ARRAY_LIST_IBM   103073
#define GL_TEXTURE_COORD_ARRAY_LIST_IBM 103074
#define GL_EDGE_FLAG_ARRAY_LIST_IBM 103075
#define GL_FOG_COORDINATE_ARRAY_LIST_IBM 103076
#define GL_SECONDARY_COLOR_ARRAY_LIST_IBM 103077
#define GL_VERTEX_ARRAY_LIST_STRIDE_IBM 103080
#define GL_NORMAL_ARRAY_LIST_STRIDE_IBM 103081
#define GL_COLOR_ARRAY_LIST_STRIDE_IBM 103082
#define GL_INDEX_ARRAY_LIST_STRIDE_IBM 103083
#define GL_TEXTURE_COORD_ARRAY_LIST_STRIDE_IBM 103084
#define GL_EDGE_FLAG_ARRAY_LIST_STRIDE_IBM 103085
#define GL_FOG_COORDINATE_ARRAY_LIST_STRIDE_IBM 103086
#define GL_SECONDARY_COLOR_ARRAY_LIST_STRIDE_IBM 103087
#define GL_MODELVIEW0_EXT          GL_MODELVIEW
#define GL_MODELVIEW0_MATRIX_EXT   GL_MODELVIEW_MATRIX
#define GL_MODELVIEW0_STACK_DEPTH_EXT GL_MODELVIEW_STACK_DEPTH

typedef unsigned int GLhandleARB;
typedef ptrdiff_t GLintptr;
typedef char GLchar;
typedef char GLcharARB;
typedef ptrdiff_t GLsizeiptr;
typedef ptrdiff_t GLsizeiptrARB;
typedef long int GLintptrARB;
extern void glAttachShader(GLuint, GLuint);
extern void glBeginQuery(GLenum, GLuint);
extern void glBindAttribLocation(GLuint, GLuint, const GLchar *);
extern void glBindBuffer(GLenum, GLuint);
extern void glBlendEquationSeparate(GLenum, GLenum);
extern void glBlendFuncSeparate(GLenum, GLenum, GLenum, GLenum);
extern void glBufferData(GLenum, GLsizeiptr, const GLvoid *,
GLenum);
extern void glBufferSubData(GLenum, GLintptr, GLsizeiptr, const
GLvoid *);
extern void glCompileShader(GLuint);
extern GLuint glCreateProgram(void);
extern GLuint glCreateShader(GLenum);
extern void glDeleteBuffers(GLsizei, const GLuint *);
extern void glDeleteProgram(GLuint);
extern void glDeleteQueries(GLsizei, const GLuint *);
extern void glDeleteShader(GLuint);
extern void glDetachShader(GLuint, GLuint);
extern void glDisableVertexAttribArray(GLuint);
extern void glDrawBuffers(GLsizei, const GLenum *);
extern void glEnableVertexAttribArray(GLuint);
extern void glFogCoordPointer(GLenum, GLsizei, const GLvoid *);
extern void glFogCoordd(GLdouble);
extern void glFogCoorddv(const GLdouble *);
extern void glFogCoordf(GLfloat);
extern void glFogCoordfv(const GLfloat *);
extern void glGenBuffers(GLsizei, GLuint *);
extern void glGenQueries(GLsizei, GLuint *);
extern void glGetActiveAttrib(GLuint, GLuint, GLsizei, GLsizei *,
GLint *,
                        GLenum *, GLchar *);
extern void glGetActiveUniform(GLuint, GLuint, GLsizei, GLsizei *,
GLint *,
                        GLenum *, GLchar *);
extern void glGetAttachedShaders(GLuint, GLsizei, GLsizei *, GLuint
*);

```

```

extern GLint glGetAttribLocation(GLuint, const GLchar *);
extern void glGetBufferParameteriv(GGLenum, GGLenum, GLint *);
extern void glGetBufferPointerv(GGLenum, GGLenum, GLvoid * *);
extern void glGetBufferSubData(GGLenum, GLintptr, GLsizeiptr, GLvoid *);
extern void glGetProgramInfoLog(GLuint, GLsizei, GLsizei *, GLchar *);
extern void glGetProgramiv(GLuint, GGLenum, GLint *);
extern void glGetQueryObjectiv(GLuint, GGLenum, GLint *);
extern void glGetQueryObjectuiv(GLuint, GGLenum, GLuint *);
extern void glGetQueryiv(GGLenum, GGLenum, GLint *);
extern void glGetShaderInfoLog(GLuint, GLsizei, GLsizei *, GLchar *);
extern void glGetShaderSource(GLuint, GLsizei, GLsizei *, GLchar *);
extern void glGetShaderiv(GLuint, GGLenum, GLint *);
extern GLint glGetUniformLocation(GLuint, const GLchar *);
extern void glGetUniformfv(GLuint, GLint, GLfloat *);
extern void glGetUniformiv(GLuint, GLint, GLint *);
extern void glGetVertexAttribPointerv(GLuint, GGLenum, GLvoid * *);
extern void glGetVertexAttribdv(GLuint, GGLenum, GLdouble *);
extern void glGetVertexAttribfv(GLuint, GGLenum, GLfloat *);
extern void glGetVertexAttribiv(GLuint, GGLenum, GLint *);
extern GLboolean glIsBuffer(GLuint);
extern GLboolean glIsProgram(GLuint);
extern GLboolean glIsQuery(GLuint);
extern GLboolean glIsShader(GLuint);
extern void glLinkProgram(GLuint);
extern GLvoid *glMapBuffer(GGLenum, GGLenum);
extern void glMultiDrawArrays(GGLenum, const GLint *, const GLsizei *,
                             GLsizei);
extern void glMultiDrawElements(GGLenum, const GLsizei *, GGLenum,
                                const GLvoid * *, GLsizei);
extern void glPointParameterf(GGLenum, GLfloat);
extern void glPointParameteriv(GGLenum, const GLfloat *);
extern void glPointParameteri(GGLenum, GLint);
extern void glPointParameteriv(GGLenum, const GLint *);
extern void glSecondaryColor3b(GLbyte, GLbyte, GLbyte);
extern void glSecondaryColor3bv(const GLbyte *);
extern void glSecondaryColor3d(GLdouble, GLdouble, GLdouble);
extern void glSecondaryColor3dv(const GLdouble *);
extern void glSecondaryColor3f(GLfloat, GLfloat, GLfloat);
extern void glSecondaryColor3fv(const GLfloat *);
extern void glSecondaryColor3i(GLint, GLint, GLint);
extern void glSecondaryColor3iv(const GLint *);
extern void glSecondaryColor3s(GLshort, GLshort, GLshort);
extern void glSecondaryColor3sv(const GLshort *);
extern void glSecondaryColor3ub(GLubyte, GLubyte, GLubyte);
extern void glSecondaryColor3ubv(const GLubyte *);
extern void glSecondaryColor3ui(GLuint, GLuint, GLuint);
extern void glSecondaryColor3uiv(const GLuint *);
extern void glSecondaryColor3us(GLushort, GLushort, GLushort);
extern void glSecondaryColor3usv(const GLushort *);
extern void glSecondaryColorPointer(GLint, GGLenum, GLsizei,
                                    const GLvoid *);
extern void glShaderSource(GLuint, GLsizei, const GLchar * *,
                           const GLint *);
extern void glStencilFuncSeparate(GGLenum, GGLenum, GLint, GLuint);
extern void glStencilMaskSeparate(GGLenum, GLuint);
extern void glStencilOpSeparate(GGLenum, GGLenum, GGLenum);
extern void glUniform1f(GLint, GLfloat);
extern void glUniform1fv(GLint, GLsizei, const GLfloat *);
extern void glUniform1i(GLint, GLint);
extern void glUniform1iv(GLint, GLsizei, const GLint *);
extern void glUniform2f(GLint, GLfloat, GLfloat);

```

```

extern void glUniform2fv(GLint, GLsizei, const GLfloat *);
extern void glUniform2i(GLint, GLint, GLint);
extern void glUniform2iv(GLint, GLsizei, const GLint *);
extern void glUniform3f(GLint, GLfloat, GLfloat, GLfloat);
extern void glUniform3fv(GLint, GLsizei, const GLfloat *);
extern void glUniform3i(GLint, GLint, GLint, GLint);
extern void glUniform3iv(GLint, GLsizei, const GLint *);
extern void glUniform4f(GLint, GLfloat, GLfloat, GLfloat, GLfloat);
extern void glUniform4fv(GLint, GLsizei, const GLfloat *);
extern void glUniform4i(GLint, GLint, GLint, GLint, GLint);
extern void glUniform4iv(GLint, GLsizei, const GLint *);
extern void glUniformMatrix2fv(GLint, GLsizei, GLboolean, const
GLfloat *);
extern void glUniformMatrix2x3fv(GLint, GLsizei, GLboolean,
const GLfloat *);
extern void glUniformMatrix2x4fv(GLint, GLsizei, GLboolean,
const GLfloat *);
extern void glUniformMatrix3fv(GLint, GLsizei, GLboolean, const
GLfloat *);
extern void glUniformMatrix3x2fv(GLint, GLsizei, GLboolean,
const GLfloat *);
extern void glUniformMatrix3x4fv(GLint, GLsizei, GLboolean,
const GLfloat *);
extern void glUniformMatrix4fv(GLint, GLsizei, GLboolean, const
GLfloat *);
extern void glUniformMatrix4x2fv(GLint, GLsizei, GLboolean,
const GLfloat *);
extern void glUniformMatrix4x3fv(GLint, GLsizei, GLboolean,
const GLfloat *);
extern GLboolean glUnmapBuffer(GLenum);
extern void glUseProgram(GLuint);
extern void glValidateProgram(GLuint);
extern void glVertexAttrib1d(GLuint, GLdouble);
extern void glVertexAttrib1dv(GLuint, const GLdouble *);
extern void glVertexAttrib1f(GLuint, GLfloat);
extern void glVertexAttrib1fv(GLuint, const GLfloat *);
extern void glVertexAttrib1s(GLuint, GLshort);
extern void glVertexAttrib1sv(GLuint, const GLshort *);
extern void glVertexAttrib2d(GLuint, GLdouble, GLdouble);
extern void glVertexAttrib2dv(GLuint, const GLdouble *);
extern void glVertexAttrib2f(GLuint, GLfloat, GLfloat);
extern void glVertexAttrib2fv(GLuint, const GLfloat *);
extern void glVertexAttrib2s(GLuint, GLshort, GLshort);
extern void glVertexAttrib2sv(GLuint, const GLshort *);
extern void glVertexAttrib3d(GLuint, GLdouble, GLdouble, GLdouble);
extern void glVertexAttrib3dv(GLuint, const GLdouble *);
extern void glVertexAttrib3f(GLuint, GLfloat, GLfloat, GLfloat);
extern void glVertexAttrib3fv(GLuint, const GLfloat *);
extern void glVertexAttrib3s(GLuint, GLshort, GLshort, GLshort);
extern void glVertexAttrib3sv(GLuint, const GLshort *);
extern void glVertexAttrib4Nbv(GLuint, const GLbyte *);
extern void glVertexAttrib4Niv(GLuint, const GLint *);
extern void glVertexAttrib4Nsv(GLuint, const GLshort *);
extern void glVertexAttrib4Nub(GLuint, GLubyte, GLubyte, GLubyte,
GLubyte);
extern void glVertexAttrib4Nubv(GLuint, const GLubyte *);
extern void glVertexAttrib4Nuiv(GLuint, const GLuint *);
extern void glVertexAttrib4Nusv(GLuint, const GLushort *);
extern void glVertexAttrib4bv(GLuint, const GLbyte *);
extern void glVertexAttrib4d(GLuint, GLdouble, GLdouble, GLdouble,
GLdouble);
extern void glVertexAttrib4dv(GLuint, const GLdouble *);
extern void glVertexAttrib4f(GLuint, GLfloat, GLfloat, GLfloat,
GLfloat);
extern void glVertexAttrib4fv(GLuint, const GLfloat *);
extern void glVertexAttrib4iv(GLuint, const GLint *);

```

```

extern void glVertexAttrib4s(GLuint, GLshort, GLshort, GLshort,
GLshort);
extern void glVertexAttrib4sv(GLuint, const GLshort *);
extern void glVertexAttrib4ubv(GLuint, const GLubyte *);
extern void glVertexAttrib4uiv(GLuint, const GLuint *);
extern void glVertexAttrib4usv(GLuint, const GLushort *);
extern void glVertexAttribPointer(GLuint, GLint, GLenum, GLboolean,
GLsizei, const GLvoid *);

extern void glWindowPos2d(GLdouble, GLdouble);
extern void glWindowPos2dv(const GLdouble *);
extern void glWindowPos2f(GLfloat, GLfloat);
extern void glWindowPos2fv(const GLfloat *);
extern void glWindowPos2i(GLint, GLint);
extern void glWindowPos2iv(const GLint *);
extern void glWindowPos2s(GLshort, GLshort);
extern void glWindowPos2sv(const GLshort *);
extern void glWindowPos3d(GLdouble, GLdouble, GLdouble);
extern void glWindowPos3dv(const GLdouble *);
extern void glWindowPos3f(GLfloat, GLfloat, GLfloat);
extern void glWindowPos3fv(const GLfloat *);
extern void glWindowPos3i(GLint, GLint, GLint);
extern void glWindowPos3iv(const GLint *);
extern void glWindowPos3s(GLshort, GLshort, GLshort);
extern void glWindowPos3sv(const GLshort *);

```

7.2.3 GL/glx.h

```

#define GLX_EXTENSION_NAME      "GLX"
#define GLX_FRONT_LEFT_BUFFER_BIT 0x00000001
#define GLX_RGBA_BIT 0x00000001
#define GLX_WINDOW_BIT 0x00000001
#define GLX_COLOR_INDEX_BIT 0x00000002
#define GLX_FRONT_RIGHT_BUFFER_BIT 0x00000002
#define GLX_PIXMAP_BIT 0x00000002
#define GLX_BACK_LEFT_BUFFER_BIT 0x00000004
#define GLX_PBUFFER_BIT 0x00000004
#define GLX_BACK_RIGHT_BUFFER_BIT 0x00000008
#define GLX_AUX_BUFFERS_BIT 0x00000010
#define GLX_DEPTH_BUFFER_BIT 0x00000020
#define GLX_STENCIL_BUFFER_BIT 0x00000040
#define GLX_ACCUM_BUFFER_BIT 0x00000080
#define GLX_PBUFFER_CLOBBER_MASK 0x08000000
#define GLX_SAMPLE_BUFFERS 0x186a0
#define GLX_SAMPLES 0x186a1
#define GLX_CONFIG_CAVEAT 0x20
#define GLX_X_VISUAL_TYPE 0x22
#define GLX_TRANSPARENT_TYPE 0x23
#define GLX_TRANSPARENT_INDEX_VALUE 0x24
#define GLX_TRANSPARENT_RED_VALUE 0x25
#define GLX_TRANSPARENT_GREEN_VALUE 0x26
#define GLX_TRANSPARENT_BLUE_VALUE 0x27
#define GLX_TRANSPARENT_ALPHA_VALUE 0x28
#define GLX_NONE 0x8000
#define GLX_SLOW_CONFIG 0x8001
#define GLX_TRUE_COLOR 0x8002
#define GLX_DIRECT_COLOR 0x8003
#define GLX_PSEUDO_COLOR 0x8004
#define GLX_STATIC_COLOR 0x8005
#define GLX_GRAY_SCALE 0x8006
#define GLX_STATIC_GRAY 0x8007
#define GLX_TRANSPARENT_RGB 0x8008
#define GLX_TRANSPARENT_INDEX 0x8009
#define GLX_VISUAL_ID 0x800B
#define GLX_SCREEN 0x800C
#define GLX_NON_CONFORMANT_CONFIG 0x800D

```

```

#define GLX_DRAWABLE_TYPE          0x8010
#define GLX_RENDER_TYPE            0x8011
#define GLX_X_RENDERABLE          0x8012
#define GLX_FBCONFIG_ID           0x8013
#define GLX_RGBA_TYPE             0x8014
#define GLX_COLOR_INDEX_TYPE      0x8015
#define GLX_MAX_PBUFFER_WIDTH     0x8016
#define GLX_MAX_PBUFFER_HEIGHT   0x8017
#define GLX_MAX_PBUFFER_PIXELS   0x8018
#define GLX_PRESERVED_CONTENTS    0x801B
#define GLX_LARGEST_PBUFFER       0x801C
#define GLX_WIDTH                 0x801D
#define GLX_HEIGHT                0x801E
#define GLX_EVENT_MASK            0x801F
#define GLX_DAMAGED               0x8020
#define GLX_SAVED                 0x8021
#define GLX_WINDOW                0x8022
#define GLX_PBUFFER               0x8023
#define GLX_PBUFFER_HEIGHT        0x8040
#define GLX_PBUFFER_WIDTH         0x8041
#define GLX_DONT_CARE             0xFFFFFFFF
#define GLX_ARB_get_proc_address  1
#define GLX_ARB_render_texture    1
#define GLX_BAD_SCREEN            1
#define GLX_EXT_texture_from_pixmap 1
#define GLX_USE_GL                1
#define GLX_VENDOR                1
#define GLX_VERSION_1_1          1
#define GLX_VERSION_1_2          1
#define GLX_VERSION_1_3          1
#define GLX_VERSION_1_4          1
#define GLX_BLUE_SIZE            10
#define GLX_ALPHA_SIZE           11
#define GLX_DEPTH_SIZE           12
#define GLX_STENCIL_SIZE         13
#define GLX_ACCUM_RED_SIZE       14
#define GLX_ACCUM_GREEN_SIZE     15
#define GLX_ACCUM_BLUE_SIZE      16
#define GLX_ACCUM_ALPHA_SIZE     17
#define GLX_BAD_ATTRIBUTE        2
#define GLX_BUFFER_SIZE          2
#define GLX_VERSION              2
#define GLX_EXTENSIONS           3
#define GLX_LEVEL                3
#define GLX_NO_EXTENSION         3
#define GLX_GLXEXT_VERSION       32
#define GLX_BAD_VISUAL           4
#define GLX_RGBA                 4
#define GLX_BAD_CONTEXT          5
#define GLX_DOUBLEBUFFER         5
#define GLX_BAD_VALUE            6
#define GLX_STEREO               6
#define GLX_AUX_BUFFERS          7
#define GLX_BAD_ENUM             7
#define GLX_RED_SIZE             8
#define GLX_GREEN_SIZE           9

typedef struct __GLXcontextRec *GLXContext;
typedef struct __GLXFBConfigRec *GLXFBConfig;
typedef XID GLXDrawable;
typedef union __GLXEvent {
    GLXPbufferClobberEvent glxpbufferclobber;
    GLXBufferSwapComplete glxbufferswapcomplete;
    long int pad[24];
} GLXEvent;
typedef XID GLXContextID;

```



```

typedef XID GLXPixmap;
typedef struct {
    int event_type;
    int draw_type;
    unsigned long int serial;
    int send_event;
    Display *display;
    GLXDrawable drawable;
    unsigned int buffer_mask;
    unsigned int aux_buffer;
    int x;
    int y;
    int width;
    int height;
    int count;
} GLXPbufferClobberEvent;
typedef XID GLXPbuffer;
typedef XID GLXWindow;
typedef XID GLXFBConfigID;
typedef void (*__GLXextFuncPtr) (void);
typedef struct {
    int type;
    unsigned long int serial;
    int send_event;
    Display *display;
    GLXDrawable drawable;
    int event_type;
    int64_t ust;
    int64_t msc;
    int64_t sbc;
} GLXBufferSwapComplete;
extern GLXFBConfig *glXChooseFBConfig(Display * dpy, int screen,
                                     const int *attribList, int *nitems);
extern XVisualInfo *glXChooseVisual(Display * dpy, int screen,
                                    int *attribList);
extern void glXCopyContext(Display * dpy, struct __GLXcontextRec
*src,
                           struct __GLXcontextRec *dst,
                           long unsigned int mask);
extern GLXContext glXCreateContext(Display * dpy, XVisualInfo * vis,
                                   struct __GLXcontextRec *shareList,
                                   int direct);
extern GLXPixmap glXCreateGLXPixmap(Display * dpy, XVisualInfo *
vis,
                                   Pixmap pixmap);
extern GLXContext glXCreateNewContext(Display * dpy,
                                       struct __GLXFBConfigRec *config,
                                       int renderType,
                                       struct __GLXcontextRec *shareList,
                                       int direct);
extern GLXPbuffer glXCreatePbuffer(Display * dpy,
                                    struct __GLXFBConfigRec *config,
                                    const int *attribList);
extern GLXPixmap glXCreatePixmap(Display * dpy,
                                  struct __GLXFBConfigRec *config,
                                  Pixmap pixmap, const int *attribList);
extern GLXWindow glXCreateWindow(Display * dpy,
                                  struct __GLXFBConfigRec *config,
                                  Window win, const int *attribList);
extern void glXDestroyContext(Display * dpy, struct __GLXcontextRec
*ctx);
extern void glXDestroyGLXPixmap(Display * dpy, GLXPixmap pix);
extern void glXDestroyPbuffer(Display * dpy, GLXPbuffer pbuf);
extern void glXDestroyPixmap(Display * dpy, GLXPixmap pixmap);
extern void glXDestroyWindow(Display * dpy, GLXWindow window);
extern const char *glXGetClientString(Display * dpy, int name);

```



```

extern int glXGetConfig(Display * dpy, XVisualInfo * vis, int attrib,
                        int *value);
extern GLXContext glXGetCurrentContext(void);
extern Display *glXGetCurrentDisplay(void);
extern GLXDrawable glXGetCurrentDrawable(void);
extern GLXDrawable glXGetCurrentReadDrawable(void);
extern int glXGetFBConfigAttrib(Display * dpy, GLXFBConfig config,
                                int attribute, int *value);
extern GLXFBConfig *glXGetFBConfigs(Display * dpy, int screen,
                                     int *nelements);
extern void (*glXGetProcAddress(const GLubyte * procName)) (void);
extern void glXGetSelectedEvent(Display * dpy, GLXDrawable drawable,
                                unsigned long int *mask);
extern XVisualInfo *glXGetVisualFromFBConfig(Display * dpy,
                                              GLXFBConfig config);
extern int glXIsDirect(Display * dpy, struct __GLXcontextRec *ctx);
extern int glXMakeContextCurrent(Display * dpy, GLXDrawable draw,
                                GLXDrawable read, GLXContext ctx);
extern int glXMakeCurrent(Display * dpy, GLXDrawable drawable,
                           struct __GLXcontextRec *ctx);
extern int glXQueryContext(Display * dpy, GLXContext ctx, int
                           attribute,
                           int *value);
extern void glXQueryDrawable(Display * dpy, GLXDrawable draw,
                              int attribute, unsigned int *value);
extern int glXQueryExtension(Display * dpy, int *errorBase,
                              int *eventBase);
extern const char *glXQueryExtensionsString(Display * dpy, int
                                              screen);
extern const char *glXQueryServerString(Display * dpy, int screen,
                                          int name);
extern int glXQueryVersion(Display * dpy, int *major, int *minor);
extern void glXSelectEvent(Display * dpy, GLXDrawable drawable,
                           long unsigned int mask);
extern void glXSwapBuffers(Display * dpy, GLXDrawable drawable);
extern void glXUseXFont(Font font, int first, int count, int
                        listBase);
extern void glXWaitGL(void);
extern void glXWaitX(void);

```

7.2.4 GL/glxext.h

```

#define GLX_VISUAL_CAVEAT_EXT 0x20
#define GLX_X_VISUAL_TYPE_EXT 0x22
#define GLX_TRANSPARENT_TYPE_EXT 0x23
#define GLX_TRANSPARENT_INDEX_VALUE_EXT 0x24
#define GLX_TRANSPARENT_RED_VALUE_EXT 0x25
#define GLX_TRANSPARENT_GREEN_VALUE_EXT 0x26
#define GLX_TRANSPARENT_BLUE_VALUE_EXT 0x27
#define GLX_TRANSPARENT_ALPHA_VALUE_EXT 0x28
#define GLX_NONE_EXT 0x8000
#define GLX_SLOW_VISUAL_EXT 0x8001
#define GLX_TRUE_COLOR_EXT 0x8002
#define GLX_DIRECT_COLOR_EXT 0x8003
#define GLX_PSEUDO_COLOR_EXT 0x8004
#define GLX_STATIC_COLOR_EXT 0x8005
#define GLX_GRAY_SCALE_EXT 0x8006
#define GLX_STATIC_GRAY_EXT 0x8007
#define GLX_TRANSPARENT_RGB_EXT 0x8008
#define GLX_TRANSPARENT_INDEX_EXT 0x8009
#define GLX_SHARE_CONTEXT_EXT 0x800A
#define GLX_VISUAL_ID_EXT 0x800B
#define GLX_SCREEN_EXT 0x800C
#define GLX_NON_CONFORMANT_VISUAL_EXT 0x800D
#define GLX_SAMPLE_BUFFERS_3DFX 0x8050

```

```

#define GLX_SAMPLES_3DFX          0x8051
#define GLX_ARB_multisample      1
#define GLX_EXT_import_context   1
#define GLX_EXT_visual_info      1
#define GLX_EXT_visual_rating    1
#define GLX_SUN_get_transparent_index 1
#define GLX_SAMPLE_BUFFERS_ARB   100000
#define GLX_SAMPLES_ARB          100001

extern void glXFreeContextEXT(Display *, GLXContext);
extern GLXContextID glXGetContextIDEXT(GLXContext);
extern __GLXextFuncPtr glXGetProcAddressARB(const GLubyte *
procName);
extern GLXContext glXImportContextEXT(Display *, GLXContextID);
extern int glXQueryContextInfoEXT(Display *, GLXContext, int, int
*);

```

7.3 Interfaces for libGLU

Table 7-4 defines the library name and shared object name for the libGLU library

Table 7-4 libGLU Definition

Library:	libGLU
SONAME:	libGLU.so.1

The behavior of the interfaces in this library is specified by the following specifications:

[GLU] OpenGL Utilities

7.3.1 GL Utilities

7.3.1.1 Interfaces for GL Utilities

An LSB conforming implementation shall provide the generic functions for GL Utilities specified in Table 7-5, with the full mandatory functionality as described in the referenced underlying specification.

Table 7-5 libGLU - GL Utilities Function Interfaces

gluBeginCurve [GLU]	gluBeginPolygon [GLU]	gluBeginSurface [GLU]	gluBeginTrim [GLU]
gluBuild1DMip mapLevels [GLU]	gluBuild1DMip maps [GLU]	gluBuild2DMip mapLevels [GLU]	gluBuild2DMip maps [GLU]
gluBuild3DMip mapLevels [GLU]	gluBuild3DMip maps [GLU]	gluCheckExtensi on [GLU]	gluCylinder [GLU]
gluDeleteNurbsR enderer [GLU]	gluDeleteQuadri c [GLU]	gluDeleteTess [GLU]	gluDisk [GLU]
gluEndCurve [GLU]	gluEndPolygon [GLU]	gluEndSurface [GLU]	gluEndTrim [GLU]
gluErrorString [GLU]	gluGetNurbsPro perty [GLU]	gluGetString [GLU]	gluGetTessPrope rty [GLU]

gluLoadSamplingMatrices [GLU]	gluLookAt [GLU]	gluNewNurbsRenderer [GLU]	gluNewQuadric [GLU]
gluNewTess [GLU]	gluNextContour [GLU]	gluNurbsCallback [GLU]	gluNurbsCallbackData [GLU]
gluNurbsCallbackDataEXT [GLU]	gluNurbsCurve [GLU]	gluNurbsProperty [GLU]	gluNurbsSurface [GLU]
gluOrtho2D [GLU]	gluPartialDisk [GLU]	gluPerspective [GLU]	gluPickMatrix [GLU]
gluProject [GLU]	gluPwlCurve [GLU]	gluQuadricCallback [GLU]	gluQuadricDrawStyle [GLU]
gluQuadricNormals [GLU]	gluQuadricOrientation [GLU]	gluQuadricTexture [GLU]	gluScaleImage [GLU]
gluSphere [GLU]	gluTessBeginContour [GLU]	gluTessBeginPolygon [GLU]	gluTessCallback [GLU]
gluTessEndContour [GLU]	gluTessEndPolygon [GLU]	gluTessNormal [GLU]	gluTessProperty [GLU]
gluTessVertex [GLU]	gluUnProject [GLU]	gluUnProject4 [GLU]	

7.4 Data Definitions for libGLU

This section defines global identifiers and their values that are associated with interfaces contained in libGLU. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

7.4.1 GL/glu.h

```
#define GLU_FALSE          0
#define GLU_EXT_nurbs_tessellator    1
#define GLU_EXT_object_space_tess    1
#define GLU_TRUE           1
#define GLU_VERSION_1_1    1
#define GLU_VERSION_1_2    1
#define GLU_VERSION_1_3    1
#define GLU_TESS_MAX_COORD    1.0e150
#define GLU_SMOOTH            100000
#define GLU_FLAT              100001
#define GLU_NONE              100002
#define GLU_POINT             100010
```

```

#define GLU_LINE 100011
#define GLU_FILL 100012
#define GLU_SILHOUETTE 100013
#define GLU_OUTSIDE 100020
#define GLU_INSIDE 100021
#define GLU_BEGIN 100100
#define GLU_TESS_BEGIN 100100
#define GLU_TESS_VERTEX 100101
#define GLU_VERTEX 100101
#define GLU_END 100102
#define GLU_TESS_END 100102
#define GLU_ERROR 100103
#define GLU_NURBS_ERROR 100103
#define GLU_TESS_ERROR 100103
#define GLU_EDGE_FLAG 100104
#define GLU_TESS_EDGE_FLAG 100104
#define GLU_TESS_COMBINE 100105
#define GLU_TESS_BEGIN_DATA 100106
#define GLU_TESS_VERTEX_DATA 100107
#define GLU_TESS_END_DATA 100108
#define GLU_TESS_ERROR_DATA 100109
#define GLU_TESS_EDGE_FLAG_DATA 100110
#define GLU_TESS_COMBINE_DATA 100111
#define GLU_CW 100120
#define GLU_CCW 100121
#define GLU_INTERIOR 100122
#define GLU_EXTERIOR 100123
#define GLU_UNKNOWN 100124
#define GLU_TESS_WINDING_ODD 100130
#define GLU_TESS_WINDING_NONZERO 100131
#define GLU_TESS_WINDING_POSITIVE 100132
#define GLU_TESS_WINDING_NEGATIVE 100133
#define GLU_TESS_WINDING_ABS_GEO_TWO 100134
#define GLU_TESS_WINDING_RULE 100140
#define GLU_TESS_BOUNDARY_ONLY 100141
#define GLU_TESS_TOLERANCE 100142
#define GLU_TESS_ERROR1 100151
#define GLU_TESS_MISSING_BEGIN_POLYGON 100151
#define GLU_TESS_ERROR2 100152
#define GLU_TESS_MISSING_BEGIN_CONTOUR 100152
#define GLU_TESS_ERROR3 100153
#define GLU_TESS_MISSING_END_POLYGON 100153
#define GLU_TESS_ERROR4 100154
#define GLU_TESS_MISSING_END_CONTOUR 100154
#define GLU_TESS_COORD_TOO_LARGE 100155
#define GLU_TESS_ERROR5 100155
#define GLU_TESS_ERROR6 100156
#define GLU_TESS_NEED_COMBINE_CALLBACK 100156
#define GLU_TESS_ERROR7 100157
#define GLU_TESS_ERROR8 100158
#define GLU_NURBS_MODE 100160
#define GLU_NURBS_MODE_EXT 100160
#define GLU_NURBS_TESSELLATOR 100161
#define GLU_NURBS_TESSELLATOR_EXT 100161
#define GLU_NURBS_RENDERER 100162
#define GLU_NURBS_RENDERER_EXT 100162
#define GLU_NURBS_BEGIN 100164
#define GLU_NURBS_BEGIN_EXT 100164
#define GLU_NURBS_VERTEX 100165
#define GLU_NURBS_VERTEX_EXT 100165
#define GLU_NURBS_NORMAL 100166
#define GLU_NURBS_NORMAL_EXT 100166
#define GLU_NURBS_COLOR 100167
#define GLU_NURBS_COLOR_EXT 100167
#define GLU_NURBS_TEXTURE_COORD 100168
#define GLU_NURBS_TEX_COORD_EXT 100168

```

```

#define GLU_NURBS_END 100169
#define GLU_NURBS_END_EXT 100169
#define GLU_NURBS_BEGIN_DATA 100170
#define GLU_NURBS_BEGIN_DATA_EXT 100170
#define GLU_NURBS_VERTEX_DATA 100171
#define GLU_NURBS_VERTEX_DATA_EXT 100171
#define GLU_NURBS_NORMAL_DATA 100172
#define GLU_NURBS_NORMAL_DATA_EXT 100172
#define GLU_NURBS_COLOR_DATA 100173
#define GLU_NURBS_COLOR_DATA_EXT 100173
#define GLU_NURBS_TEXTURE_COORD_DATA 100174
#define GLU_NURBS_TEX_COORD_DATA_EXT 100174
#define GLU_NURBS_END_DATA 100175
#define GLU_NURBS_END_DATA_EXT 100175
#define GLU_AUTO_LOAD_MATRIX 100200
#define GLU_CULLING 100201
#define GLU_PARAMETRIC_TOLERANCE 100202
#define GLU_SAMPLING_TOLERANCE 100203
#define GLU_DISPLAY_MODE 100204
#define GLU_SAMPLING_METHOD 100205
#define GLU_U_STEP 100206
#define GLU_V_STEP 100207
#define GLU_OBJECT_PARAMETRIC_ERROR 100208
#define GLU_OBJECT_PARAMETRIC_ERROR_EXT 100208
#define GLU_OBJECT_PATH_LENGTH 100209
#define GLU_OBJECT_PATH_LENGTH_EXT 100209
#define GLU_MAP1_TRIM_2 100210
#define GLU_MAP1_TRIM_3 100211
#define GLU_PATH_LENGTH 100215
#define GLU_PARAMETRIC_ERROR 100216
#define GLU_DOMAIN_DISTANCE 100217
#define GLU_OUTLINE_POLYGON 100240
#define GLU_OUTLINE_PATCH 100241
#define GLU_NURBS_ERROR1 100251
#define GLU_NURBS_ERROR2 100252
#define GLU_NURBS_ERROR3 100253
#define GLU_NURBS_ERROR4 100254
#define GLU_NURBS_ERROR5 100255
#define GLU_NURBS_ERROR6 100256
#define GLU_NURBS_ERROR7 100257
#define GLU_NURBS_ERROR8 100258
#define GLU_NURBS_ERROR9 100259
#define GLU_NURBS_ERROR10 100260
#define GLU_NURBS_ERROR11 100261
#define GLU_NURBS_ERROR12 100262
#define GLU_NURBS_ERROR13 100263
#define GLU_NURBS_ERROR14 100264
#define GLU_NURBS_ERROR15 100265
#define GLU_NURBS_ERROR16 100266
#define GLU_NURBS_ERROR17 100267
#define GLU_NURBS_ERROR18 100268
#define GLU_NURBS_ERROR19 100269
#define GLU_NURBS_ERROR20 100270
#define GLU_NURBS_ERROR21 100271
#define GLU_NURBS_ERROR22 100272
#define GLU_NURBS_ERROR23 100273
#define GLU_NURBS_ERROR24 100274
#define GLU_NURBS_ERROR25 100275
#define GLU_NURBS_ERROR26 100276
#define GLU_NURBS_ERROR27 100277
#define GLU_NURBS_ERROR28 100278
#define GLU_NURBS_ERROR29 100279
#define GLU_NURBS_ERROR30 100280
#define GLU_NURBS_ERROR31 100281
#define GLU_NURBS_ERROR32 100282
#define GLU_NURBS_ERROR33 100283

```

```

#define GLU_NURBS_ERROR34      100284
#define GLU_NURBS_ERROR35      100285
#define GLU_NURBS_ERROR36      100286
#define GLU_NURBS_ERROR37      100287
#define GLU_VERSION            100800
#define GLU_EXTENSIONS          100801
#define GLU_INVALID_ENUM        100900
#define GLU_INVALID_VALUE        100901
#define GLU_OUT_OF_MEMORY        100902
#define GLU_INCOMPATIBLE_GL_VERSION 100903
#define GLU_INVALID_OPERATION    100904

typedef void *_GLUfuncptr;
typedef struct GLUtesselator GLUtesselatorObj;
typedef struct GLUtesselator GLUtriangulatorObj;
typedef struct GLUquadric GLUquadricObj;
typedef struct GLUnurbs GLUnurbsObj;
typedef struct GLUnurbs GLUnurbs;
typedef struct GLUquadric GLUquadric;
typedef struct GLUtesselator GLUtesselator;
extern void gluBeginCurve(GLUnurbs * nurb);
extern void gluBeginPolygon(GLUtesselator * tess);
extern void gluBeginSurface(GLUnurbs * nurb);
extern void gluBeginTrim(GLUnurbs * nurb);
extern GLint gluBuild1DMipmapLevels(GLenum target, GLint
internalFormat,
                                GLsizei width, GLenum format,
                                GLenum type, GLint level, GLint base,
                                GLint max, const void *data);
extern GLint gluBuild1DMipmaps(GLenum target, GLint internalFormat,
                                GLsizei width, GLenum format, GLenum type,
                                const void *data);
extern GLint gluBuild2DMipmapLevels(GLenum target, GLint
internalFormat,
                                GLsizei width, GLsizei height,
                                GLenum format, GLenum type,
                                GLint level, GLint base, GLint max,
                                const void *data);
extern GLint gluBuild2DMipmaps(GLenum target, GLint internalFormat,
                                GLsizei width, GLsizei height,
                                GLenum format, GLenum type,
                                const void *data);
extern GLint gluBuild3DMipmapLevels(GLenum target, GLint
internalFormat,
                                GLsizei width, GLsizei height,
                                GLsizei depth, GLenum format,
                                GLenum type, GLint level, GLint base,
                                GLint max, const void *data);
extern GLint gluBuild3DMipmaps(GLenum target, GLint internalFormat,
                                GLsizei width, GLsizei height,
                                GLsizei depth, GLenum format, GLenum type,
                                const void *data);
extern GLboolean gluCheckExtension(const GLubyte * extName,
                                const GLubyte * extString);
extern void gluCylinder(GLUquadric * quad, GLdouble base, GLdouble
top,
                                GLdouble height, GLint slices, GLint stacks);
extern void gluDeleteNurbsRenderer(GLUnurbs * nurb);
extern void gluDeleteQuadric(GLUquadric * quad);
extern void gluDeleteTess(GLUtesselator * tess);
extern void gluDisk(GLUquadric * quad, GLdouble inner, GLdouble
outer,
                                GLint slices, GLint loops);
extern void gluEndCurve(GLUnurbs * nurb);
extern void gluEndPolygon(GLUtesselator * tess);
extern void gluEndSurface(GLUnurbs * nurb);

```

```

extern void gluEndTrim(GLUnurbs * nurb);
extern const GLubyte *gluErrorString(GLenum error);
extern void gluGetNurbsProperty(GLUnurbs * nurb, GLenum property,
                                GLfloat * data);
extern const GLubyte *gluGetString(GLenum name);
extern void gluGetTessProperty(GLUtesselator * tess, GLenum which,
                                GLdouble * data);
extern void gluLoadSamplingMatrices(GLUnurbs * nurb, const GLfloat
* model,
                                const GLfloat * perspective,
                                const GLint * view);
extern void gluLookAt(GLdouble eyeX, GLdouble eyeY, GLdouble eyeZ,
                      GLdouble centerX, GLdouble centerY, GLdouble
centerZ,
                      GLdouble upX, GLdouble upY, GLdouble upZ);
extern GLUnurbs *gluNewNurbsRenderer(void);
extern GLUquadric *gluNewQuadric(void);
extern GLUtesselator *gluNewTess(void);
extern void gluNextContour(GLUtesselator * tess, GLenum type);
extern void gluNurbsCallback(GLUnurbs * nurb, GLenum which,
                             _GLUfuncptr CallBackFunc);
extern void gluNurbsCallbackData(GLUnurbs * nurb, GLvoid *
userData);
extern void gluNurbsCallbackDataEXT(GLUnurbs * nurb, GLvoid *
userData);
extern void gluNurbsCurve(GLUnurbs * nurb, GLint knotCount,
                          GLfloat * knots, GLint stride, GLfloat *
control,
                          GLint order, GLenum type);
extern void gluNurbsProperty(GLUnurbs * nurb, GLenum property,
                              GLfloat value);
extern void gluNurbsSurface(GLUnurbs * nurb, GLint sKnotCount,
                            GLfloat * sKnots, GLint tKnotCount,
                            GLfloat * tKnots, GLint sStride, GLint
tStride,
                            GLfloat * control, GLint sOrder, GLint
tOrder,
                            GLenum type);
extern void gluOrtho2D(GLdouble left, GLdouble right, GLdouble
bottom,
                      GLdouble top);
extern void gluPartialDisk(GLUquadric * quad, GLdouble inner,
                          GLdouble outer, GLint slices, GLint loops,
                          GLdouble start, GLdouble sweep);
extern void gluPerspective(GLdouble fovy, GLdouble aspect, GLdouble
zNear,
                          GLdouble zFar);
extern void gluPickMatrix(GLdouble x, GLdouble y, GLdouble delX,
                          GLdouble delY, GLint * viewport);
extern GLint gluProject(GLdouble objX, GLdouble objY, GLdouble objZ,
                       const GLdouble * model, const GLdouble * proj,
                       const GLint * view, GLdouble * winX,
                       GLdouble * winY, GLdouble * winZ);
extern void gluPwlCurve(GLUnurbs * nurb, GLint count, GLfloat *
data,
                       GLint stride, GLenum type);
extern void gluQuadricCallback(GLUquadric * quad, GLenum which,
                              _GLUfuncptr CallBackFunc);
extern void gluQuadricDrawStyle(GLUquadric * quad, GLenum draw);
extern void gluQuadricNormals(GLUquadric * quad, GLenum normal);
extern void gluQuadricOrientation(GLUquadric * quad, GLenum
orientation);
extern void gluQuadricTexture(GLUquadric * quad, GLboolean texture);
extern GLint gluScaleImage(GLenum format, GLsizei wIn, GLsizei hIn,
                           GLenum typeIn, const void *dataIn, GLsizei
wOut,

```

```

        GLsizei hOut, GLenum typeOut, GLvoid *
dataOut);
extern void gluSphere(GLUquadric * quad, GLdouble radius, GLint
slices,
        GLint stacks);
extern void gluTessBeginContour(GLUtesselator * tess);
extern void gluTessBeginPolygon(GLUtesselator * tess, GLvoid *
data);
extern void gluTessCallback(GLUtesselator * tess, GLenum which,
        _GLUfuncptr CallBackFunc);
extern void gluTessEndContour(GLUtesselator * tess);
extern void gluTessEndPolygon(GLUtesselator * tess);
extern void gluTessNormal(GLUtesselator * tess, GLdouble valueX,
        GLdouble valueY, GLdouble valueZ);
extern void gluTessProperty(GLUtesselator * tess, GLenum which,
        GLdouble data);
extern void gluTessVertex(GLUtesselator * tess, GLdouble * location,
        GLvoid * data);
extern GLint gluUnProject(GLdouble winX, GLdouble winY, GLdouble
winZ,
        const GLdouble * model, const GLdouble * proj,
        const GLint * view, GLdouble * objX,
        GLdouble * objY, GLdouble * objZ);
extern GLint gluUnProject4(GLdouble winX, GLdouble winY, GLdouble
winZ,
        GLdouble clipW, const GLdouble * model,
        const GLdouble * proj, const GLint * view,
        GLdouble nearVal, GLdouble farVal,
        GLdouble * objX, GLdouble * objY,
        GLdouble * objZ, GLdouble * objW);

```


IV PNG12 library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

8 Libraries

8.1 Interfaces for libpng12

Table 8-1 defines the library name and shared object name for the libpng12 library

Table 8-1 libpng12 Definition

Library:	libpng12
SONAME:	libpng12.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

8.1.1 PNG Reference library

8.1.1.1 Interfaces for PNG Reference library

An LSB conforming implementation shall provide the generic functions for PNG Reference library specified in Table 8-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 8-2 libpng12 - PNG Reference library Function Interfaces

png_access_version_number(PNG12_0) [LSB]	png_check_sig(PNG12_0) [LSB]	png_convert_from_struct_tm(PNG12_0) [LSB]
png_convert_from_time_t(PNG12_0) [LSB]	png_create_info_struct(PNG12_0) [LSB]	png_create_read_struct(PNG12_0) [LSB]
png_create_read_struct_2(PNG12_0) [LSB]	png_create_write_struct(PNG12_0) [LSB]	png_create_write_struct_2(PNG12_0) [LSB]
png_data_freer(PNG12_0) [LSB]	png_destroy_info_struct(PNG12_0) [LSB]	png_destroy_read_struct(PNG12_0) [LSB]
png_destroy_write_struct(PNG12_0) [LSB]	png_error(PNG12_0) [LSB]	png_free(PNG12_0) [LSB]
png_free_data(PNG12_0) [LSB]	png_get_IHDR(PNG12_0) [LSB]	png_get_PLTE(PNG12_0) [LSB]
png_get_bKGD(PNG12_0) [LSB]	png_get_bit_depth(PNG12_0) [LSB]	png_get_cHRM(PNG12_0) [LSB]
png_get_channels(PNG12_0) [LSB]	png_get_color_type(PNG12_0) [LSB]	png_get_error_ptr(PNG12_0) [LSB]
png_get_gAMA(PNG12_0) [LSB]	png_get_hIST(PNG12_0) [LSB]	png_get_header_ver(PNG12_0) [LSB]
png_get_iCCP(PNG12_0) [LSB]	png_get_image_height(PNG12_0) [LSB]	png_get_image_width(PNG12_0) [LSB]
png_get_interlace_type(PNG12_0) [LSB]	png_get_io_ptr(PNG12_0) [LSB]	png_get_libpng_ver(PNG12_0) [LSB]

png_get_oFFs(PNG12_0) [LSB]	png_get_pHYs(PNG12_0) [LSB]	png_get_progressive_ptr(PNG12_0) [LSB]
png_get_rowbytes(PNG12_0) [LSB]	png_get_rows(PNG12_0) [LSB]	png_get_sBIT(PNG12_0) [LSB]
png_get_sRGB(PNG12_0) [LSB]	png_get_tIME(PNG12_0) [LSB]	png_get_tRNS(PNG12_0) [LSB]
png_get_text(PNG12_0) [LSB]	png_get_unknown_chunks(PNG12_0) [LSB]	png_get_user_chunk_ptr(PNG12_0) [LSB]
png_get_valid(PNG12_0) [LSB]	png_get_x_offset_pixels(PNG12_0) [LSB]	png_get_x_pixels_per_meter(PNG12_0) [LSB]
png_get_y_offset_pixels(PNG12_0) [LSB]	png_get_y_pixels_per_meter(PNG12_0) [LSB]	png_info_init_3(PNG12_0) [LSB]
png_init_io(PNG12_0) [LSB]	png_malloc(PNG12_0) [LSB]	png_permit_mng_features(PNG12_0) [LSB]
png_process_data(PNG12_0) [LSB]	png_progressive_combine_row(PNG12_0) [LSB]	png_read_end(PNG12_0) [LSB]
png_read_image(PNG12_0) [LSB]	png_read_info(PNG12_0) [LSB]	png_read_png(PNG12_0) [LSB]
png_read_row(PNG12_0) [LSB]	png_read_rows(PNG12_0) [LSB]	png_read_update_info(PNG12_0) [LSB]
png_set_IHDR(PNG12_0) [LSB]	png_set_PLTE(PNG12_0) [LSB]	png_set_bKGD(PNG12_0) [LSB]
png_set_background(PNG12_0) [LSB]	png_set_bgr(PNG12_0) [LSB]	png_set_cHRM(PNG12_0) [LSB]
png_set_compression_buffer_size(PNG12_0) [LSB]	png_set_compression_level(PNG12_0) [LSB]	png_set_compression_mem_level(PNG12_0) [LSB]
png_set_compression_method(PNG12_0) [LSB]	png_set_compression_strategy(PNG12_0) [LSB]	png_set_compression_window_bits(PNG12_0) [LSB]
png_set_dither(PNG12_0) [LSB]	png_set_error_fn(PNG12_0) [LSB]	png_set_expand(PNG12_0) [LSB]
png_set_filler(PNG12_0) [LSB]	png_set_filter(PNG12_0) [LSB]	png_set_gAMA(PNG12_0) [LSB]
png_set_gamma(PNG12_0) [LSB]	png_set_gray_1_2_4_to_8(PNG12_0) [LSB]	png_set_gray_to_rgb(PNG12_0) [LSB]
png_set_hIST(PNG12_0) [LSB]	png_set_iCCP(PNG12_0) [LSB]	png_set_interlace_handling(PNG12_0) [LSB]
png_set_invert_alpha(PNG12_0) [LSB]	png_set_invert_mono(PNG12_0) [LSB]	png_set_keep_unknown_chunks(PNG12_0) [LSB]

png_set_mem_fn(PNG12_0) [LSB]	png_set_offs(PNG12_0) [LSB]	png_set_pHYs(PNG12_0) [LSB]
png_set_packing(PNG12_0) [LSB]	png_set_packswap(PNG12_0) [LSB]	png_set_palette_to_rgb(PNG12_0) [LSB]
png_set_progressive_read_fn(PNG12_0) [LSB]	png_set_read_fn(PNG12_0) [LSB]	png_set_read_user_chunk_fn(PNG12_0) [LSB]
png_set_read_user_transform_fn(PNG12_0) [LSB]	png_set_rgb_to_gray(PNG12_0) [LSB]	png_set_rows(PNG12_0) [LSB]
png_set_sBIT(PNG12_0) [LSB]	png_set_sRGB(PNG12_0) [LSB]	png_set_sRGB_gAMA_and_chRM(PNG12_0) [LSB]
png_set_shift(PNG12_0) [LSB]	png_set_sig_bytes(PNG12_0) [LSB]	png_set_strip_16(PNG12_0) [LSB]
png_set_strip_alpha(PNG12_0) [LSB]	png_set_swap(PNG12_0) [LSB]	png_set_swap_alpha(PNG12_0) [LSB]
png_set_tIME(PNG12_0) [LSB]	png_set_tRNS(PNG12_0) [LSB]	png_set_tRNS_to_alpha(PNG12_0) [LSB]
png_set_text(PNG12_0) [LSB]	png_set_unknown_chunk_location(PNG12_0) [LSB]	png_set_unknown_chunks(PNG12_0) [LSB]
png_set_write_fn(PNG12_0) [LSB]	png_set_write_status_fn(PNG12_0) [LSB]	png_set_write_user_transform_fn(PNG12_0) [LSB]
png_sig_cmp(PNG12_0) [LSB]	png_start_read_image(PNG12_0) [LSB]	png_warning(PNG12_0) [LSB]
png_write_chunk(PNG12_0) [LSB]	png_write_end(PNG12_0) [LSB]	png_write_flush(PNG12_0) [LSB]
png_write_image(PNG12_0) [LSB]	png_write_info(PNG12_0) [LSB]	png_write_png(PNG12_0) [LSB]
png_write_row(PNG12_0) [LSB]	png_write_rows(PNG12_0) [LSB]	

An LSB conforming implementation shall provide the generic data interfaces for PNG Reference library specified in Table 8-3, with the full mandatory functionality as described in the referenced underlying specification.

Table 8-3 libpng12 - PNG Reference library Data Interfaces

png_libpng_ver(PNG12_0) [LSB]		

8.2 Data Definitions for libpng12

This section defines global identifiers and their values that are associated with interfaces contained in libpng12. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the

reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

8.2.1 libpng12/png.h

```
#define FARDATA
#define PNGAPI
#define PNG_1_2_X
#define PNG_EASY_ACCESS_SUPPORTED
#define PNG_ERROR_NUMBERS_SUPPORTED
#define PNG_FIXED_POINT_SUPPORTED
#define PNG_FLOATING_POINT_SUPPORTED
#define PNG_FREE_ME_SUPPORTED
#define PNG_HANDLE_AS_UNKNOWN_SUPPORTED
#define PNG_IMPEXP
#define PNG_INFO_IMAGE_SUPPORTED
#define PNG_MMX_CODE_SUPPORTED
#define PNG_MNG_FEATURES_SUPPORTED
#define PNG_NO_READ_iTXt
#define PNG_NO_WRITE_iTXt
#define PNG_PROGRESSIVE_READ_SUPPORTED
#define PNG_READ_16_TO_8_SUPPORTED
#define PNG_READ_ANCILLARY_CHUNKS_SUPPORTED
#define PNG_READ_BACKGROUND_SUPPORTED
#define PNG_READ_BGR_SUPPORTED
#define PNG_READ_COMPOSITE_NODIV_SUPPORTED
#define PNG_READ_DITHER_SUPPORTED
#define PNG_READ_EMPTY_PLTE_SUPPORTED
#define PNG_READ_EXPAND_SUPPORTED
#define PNG_READ_FILLER_SUPPORTED
#define PNG_READ_GAMMA_SUPPORTED
#define PNG_READ_GRAY_TO_RGB_SUPPORTED
#define PNG_READ_INTERLACING_SUPPORTED
#define PNG_READ_INVERT_ALPHA_SUPPORTED
#define PNG_READ_INVERT_SUPPORTED
#define PNG_READ_OPT_PLTE_SUPPORTED
#define PNG_READ_PACKSWAP_SUPPORTED
#define PNG_READ_PACK_SUPPORTED
#define PNG_READ_RGB_TO_GRAY_SUPPORTED
#define PNG_READ_SHIFT_SUPPORTED
#define PNG_READ_STRIP_ALPHA_SUPPORTED
#define PNG_READ_SUPPORTED
#define PNG_READ_SWAP_ALPHA_SUPPORTED
#define PNG_READ_SWAP_SUPPORTED
#define PNG_READ_TEXT_SUPPORTED
#define PNG_READ_TRANSFORMS_SUPPORTED
#define PNG_READ_UNKNOWN_CHUNKS_SUPPORTED
#define PNG_READ_USER_CHUNKS_SUPPORTED
#define PNG_READ_USER_TRANSFORM_SUPPORTED
#define PNG_READ_bKGD_SUPPORTED
#define PNG_READ_cHRM_SUPPORTED
#define PNG_READ_gAMA_SUPPORTED
```

```

#define PNG_READ_hIST_SUPPORTED
#define PNG_READ_icCP_SUPPORTED
#define PNG_READ_oFFs_SUPPORTED
#define PNG_READ_pCAL_SUPPORTED
#define PNG_READ_pHYs_SUPPORTED
#define PNG_READ_sBIT_SUPPORTED
#define PNG_READ_sCAL_SUPPORTED
#define PNG_READ_sPLT_SUPPORTED
#define PNG_READ_sRGB_SUPPORTED
#define PNG_READ_tEXt_SUPPORTED
#define PNG_READ_tIME_SUPPORTED
#define PNG_READ_tRNS_SUPPORTED
#define PNG_READ_zTXt_SUPPORTED
#define PNG_SETJMP_SUPPORTED
#define PNG_SET_USER_LIMITS_SUPPORTED
#define PNG_TEXT_SUPPORTED
#define PNG_TIME_RFC1123_SUPPORTED
#define PNG_UNKNOWN_CHUNKS_SUPPORTED
#define PNG_USER_CHUNKS_SUPPORTED
#define PNG_USER_MEM_SUPPORTED
#define PNG_USER_TRANSFORM_PTR_SUPPORTED
#define PNG_USE_GLOBAL_ARRAYS
#define PNG_WRITE_ANCILLARY_CHUNKS_SUPPORTED
#define PNG_WRITE_BGR_SUPPORTED
#define PNG_WRITE_EMPTY_PLTE_SUPPORTED
#define PNG_WRITE_FILLER_SUPPORTED
#define PNG_WRITE_FLUSH_SUPPORTED
#define PNG_WRITE_INTERLACING_SUPPORTED
#define PNG_WRITE_INVERT_ALPHA_SUPPORTED
#define PNG_WRITE_INVERT_SUPPORTED
#define PNG_WRITE_PACKSWAP_SUPPORTED
#define PNG_WRITE_PACK_SUPPORTED
#define PNG_WRITE_SHIFT_SUPPORTED
#define PNG_WRITE_SUPPORTED
#define PNG_WRITE_SWAP_ALPHA_SUPPORTED
#define PNG_WRITE_SWAP_SUPPORTED
#define PNG_WRITE_TEXT_SUPPORTED
#define PNG_WRITE_TRANSFORMS_SUPPORTED
#define PNG_WRITE_UNKNOWN_CHUNKS_SUPPORTED
#define PNG_WRITE_USER_TRANSFORM_SUPPORTED
#define PNG_WRITE_WEIGHTED_FILTER_SUPPORTED
#define PNG_WRITE_bKGD_SUPPORTED
#define PNG_WRITE_cHRM_SUPPORTED
#define PNG_WRITE_gAMA_SUPPORTED
#define PNG_WRITE_hIST_SUPPORTED
#define PNG_WRITE_icCP_SUPPORTED
#define PNG_WRITE_oFFs_SUPPORTED
#define PNG_WRITE_pCAL_SUPPORTED
#define PNG_WRITE_pHYs_SUPPORTED
#define PNG_WRITE_sBIT_SUPPORTED
#define PNG_WRITE_sCAL_SUPPORTED
#define PNG_WRITE_sPLT_SUPPORTED
#define PNG_WRITE_sRGB_SUPPORTED
#define PNG_WRITE_tEXt_SUPPORTED
#define PNG_WRITE_tIME_SUPPORTED
#define PNG_WRITE_tRNS_SUPPORTED
#define PNG_WRITE_zTXt_SUPPORTED
#define PNG_bKGD_SUPPORTED
#define PNG_cHRM_SUPPORTED
#define PNG_gAMA_SUPPORTED
#define PNG_hIST_SUPPORTED
#define PNG_icCP_SUPPORTED
#define PNG_oFFs_SUPPORTED
#define PNG_pCAL_SUPPORTED
#define PNG_pHYs_SUPPORTED
#define PNG_sBIT_SUPPORTED

```

```

#define PNG_SCAL_SUPPORTED
#define PNG_sPLT_SUPPORTED
#define PNG_sRGB_SUPPORTED
#define PNG_tEXt_SUPPORTED
#define PNG_tIME_SUPPORTED
#define PNG_tRNS_SUPPORTED
#define PNG_zTXt_SUPPORTED
#define PNG_MMX_READ_FLAGS \
    ( PNG_ASM_FLAG_MMX_READ_COMBINE_ROW |
    PNG_ASM_FLAG_MMX_READ_INTERLACE \
    | PNG_ASM_FLAG_MMX_READ_FILTER_SUB |
    PNG_ASM_FLAG_MMX_READ_FILTER_UP | \
    PNG_ASM_FLAG_MMX_READ_FILTER_AVG |
    PNG_ASM_FLAG_MMX_READ_FILTER_PAETH \
    )
#define PNG_MMX_FLAGS \
    ( PNG_ASM_FLAG_MMX_SUPPORT_COMPILED |
    PNG_ASM_FLAG_MMX_SUPPORT_IN_CPU \
    | PNG_MMX_READ_FLAGS | PNG_MMX_WRITE_FLAGS )
#define PNG_ALL_FILTERS \
    (PNG_FILTER_NONE | PNG_FILTER_SUB | PNG_FILTER_UP |
    PNG_FILTER_AVG | \
    PNG_FILTER_PAETH)
#define png_info_init(info_ptr) \
    png_info_init_3(&info_ptr, png_sizeof(png_info));
#define png_composite(composite,fg,alpha,bg) \
    { png_uint_16 temp = (png_uint_16)((png_uint_16)(fg) * \
    (png_uint_16)(alpha) + (png_uint_16)(bg)*(png_uint_16)(255
- \
    (png_uint_16)(alpha)) + (png_uint_16)128); (composite) = \
    (png_byte)((temp + (temp >> 8)) >> 8); }
#define png_composite_16(composite,fg,alpha,bg) \
    { png_uint_32 temp = (png_uint_32)((png_uint_32)(fg) * \
    (png_uint_32)(alpha) + (png_uint_32)(bg)*(png_uint_32)65535L - \
    (png_uint_32)(alpha)) + (png_uint_32)32768L); (composite) = \
    (png_uint_16)((temp + (temp >> 16)) >> 16); }
#define PNG_HEADER_VERSION_STRING " libpng version 1.2.8 -
December 3, 2004 (header)\n"
#define PNG_LIBPNG_VER_STRING "1.2.8"
#define PNG_MMX_WRITE_FLAGS ( 0 )
#define png_jmpbuf(png_ptr) ((png_ptr)->jmpbuf)
#define PNG_SIZE_MAX ((png_size_t)(-1))
#define PNG_UINT_32_MAX ((png_uint_32)(-1))
#define PNG_UINT_31_MAX ((png_uint_32)0x7fffffffL)
#define int_p_NULL (int *)NULL
#define png_bytep_NULL (png_bytep)NULL
#define png_bytepp_NULL (png_bytepp)NULL
#define PNG_COLOR_TYPE_GRAY_ALPHA (PNG_COLOR_MASK_ALPHA)
#define PNG_COLOR_TYPE_RGB_ALPHA (PNG_COLOR_MASK_COLOR |
PNG_COLOR_MASK_ALPHA)
#define PNG_COLOR_TYPE_PALETTE (PNG_COLOR_MASK_COLOR |
PNG_COLOR_MASK_PALETTE)
#define PNG_COLOR_TYPE_RGB (PNG_COLOR_MASK_COLOR)
#define png_doublep_NULL (png_doublep)NULL
#define png_error_ptr_NULL (png_error_ptr)NULL
#define png_flush_ptr_NULL (png_flush_ptr)NULL
#define png_infopp_NULL (png_infopp)NULL
#define png_rw_ptr_NULL (png_rw_ptr)NULL
#define png_structp_NULL (png_structp)NULL
#define png_uint_16p_NULL (png_uint_16p)NULL
#define png_voidp_NULL (png_voidp)NULL
#define CVT_PTR(ptr) (ptr)
#define CVT_PTR_NOCHECK(ptr) (ptr)
#define PNG_TEXT_COMPRESSION_NONE -1

```

```

#define PNG_TEXT_COMPRESSION_zTXt_WR -2
#define PNG_TEXT_COMPRESSION_NONE_WR -3
#define PNG_BACKGROUND_GAMMA_UNKNOWN 0
#define PNG_COLOR_TYPE_GRAY 0
#define PNG_COMPRESSION_TYPE_BASE 0
#define PNG_CRC_DEFAULT 0
#define PNG_EQUATION_LINEAR 0
#define PNG_FILLER_BEFORE 0
#define PNG_FILTER_HEURISTIC_DEFAULT 0
#define PNG_FILTER_TYPE_BASE 0
#define PNG_FILTER_VALUE_NONE 0
#define PNG_HANDLE_CHUNK_AS_DEFAULT 0
#define PNG_INTERLACE_NONE 0
#define PNG_LIBPNG_VER_BUILD 0
#define PNG_LIBPNG_VER_SONUM 0
#define PNG_OFFSET_PIXEL 0
#define PNG_RESOLUTION_UNKNOWN 0
#define PNG_SCALE_UNKNOWN 0
#define PNG_TEXT_COMPRESSION_zTXt 0
#define PNG_sRGB_INTENT_PERCEPTUAL 0
#define PNG_NO_FILTERS 0x00
#define PNG_TRANSFORM_IDENTITY 0x0000
#define PNG_INFO_gAMA 0x0001
#define PNG_TRANSFORM_STRIP_16 0x0001
#define PNG_INFO_sBIT 0x0002
#define PNG_TRANSFORM_STRIP_ALPHA 0x0002
#define PNG_INFO_chRM 0x0004
#define PNG_TRANSFORM_PACKING 0x0004
#define PNG_FREE_HIST 0x0008
#define PNG_INFO_PLTE 0x0008
#define PNG_TRANSFORM_PACKSWAP 0x0008
#define PNG_FREE_ICCP 0x0010
#define PNG_INFO_tRNS 0x0010
#define PNG_TRANSFORM_EXPAND 0x0010
#define PNG_FREE_SPLT 0x0020
#define PNG_INFO_bKGD 0x0020
#define PNG_TRANSFORM_INVERT_MONO 0x0020
#define PNG_FREE_ROWS 0x0040
#define PNG_INFO_hIST 0x0040
#define PNG_TRANSFORM_SHIFT 0x0040
#define PNG_FREE_pCAL 0x0080
#define PNG_INFO_pHYs 0x0080
#define PNG_TRANSFORM_BGR 0x0080
#define PNG_ASM_FLAG_MMX_SUPPORT_COMPILED 0x01
#define PNG_FLAG_MNG_EMPTY_PLTE 0x01
#define PNG_FREE_SCAL 0x0100
#define PNG_INFO_oFFs 0x0100
#define PNG_TRANSFORM_SWAP_ALPHA 0x0100
#define PNG_ASM_FLAG_MMX_SUPPORT_IN_CPU 0x02
#define PNG_FREE_UNKN 0x0200
#define PNG_INFO_tIME 0x0200
#define PNG_TRANSFORM_SWAP_ENDIAN 0x0200
#define PNG_ASM_FLAG_MMX_READ_COMBINE_ROW 0x04
#define PNG_FLAG_MNG_FILTER_64 0x04
#define PNG_FREE_LIST 0x0400
#define PNG_INFO_pCAL 0x0400
#define PNG_TRANSFORM_INVERT_ALPHA 0x0400
#define PNG_ALL_MNG_FEATURES 0x05
#define PNG_ASM_FLAG_MMX_READ_INTERLACE 0x08
#define PNG_FILTER_NONE 0x08
#define PNG_INFO_sRGB 0x0800
#define PNG_TRANSFORM_STRIP_FILLER 0x0800
#define PNG_ASM_FLAG_MMX_READ_FILTER_SUB 0x10
#define PNG_FILTER_SUB 0x10
#define PNG_FREE_PLTE 0x1000
#define PNG_INFO_iCCP 0x1000

```



```

#define PNG_ASM_FLAG_MMX_READ_FILTER_UP 0x20
#define PNG_FILTER_UP 0x20
#define PNG_FREE_TRNS 0x2000
#define PNG_INFO_sPLT 0x2000
#define PNG_ASM_FLAG_MMX_READ_FILTER_AVG 0x40
#define PNG_FILTER_AVG 0x40
#define PNG_FREE_TEXT 0x4000
#define PNG_INFO_sCAL 0x4000
#define PNG_FREE_MUL 0x4220
#define PNG_FREE_ALL 0x7fff
#define PNG_ASM_FLAG_MMX_READ_FILTER_PAETH 0x80
#define PNG_FILTER_PAETH 0x80
#define PNG_ASM_FLAGS_INITIALIZED 0x80000000
#define PNG_INFO_IDAT 0x8000L
#define PNG_BACKGROUND_GAMMA_SCREEN 1
#define PNG_COLOR_MASK_PALETTE 1
#define PNG_CRC_ERROR_QUIT 1
#define PNG_DESTROY_WILL_FREE_DATA 1
#define PNG_EQUATION_BASE_E 1
#define PNG_FILLER_AFTER 1
#define PNG_FILTER_HEURISTIC_UNWEIGHTED 1
#define PNG_FILTER_VALUE_SUB 1
#define PNG_HANDLE_CHUNK_NEVER 1
#define PNG_INTERLACE_ADAM7 1
#define PNG_ITXT_COMPRESSION_NONE 1
#define PNG_LIBPNG_BUILD_ALPHA 1
#define PNG_LIBPNG_VER_MAJOR 1
#define PNG_OFFSET_MICROMETER 1
#define PNG_RESOLUTION_METER 1
#define PNG_SCALE_METER 1
#define PNG_SELECT_READ 1
#define PNG_SET_WILL_FREE_DATA 1
#define PNG_sRGB_INTENT_RELATIVE 1
#define PNG_USER_HEIGHT_MAX 1000000L
#define PNG_USER_WIDTH_MAX 1000000L
#define PNG_LIBPNG_VER 10208
#define PNG_LIBPNG_VER_DLLNUM 13
#define PNG_LIBPNG_BUILD_PRIVATE 16
#define PNG_BACKGROUND_GAMMA_FILE 2
#define PNG_COLOR_MASK_COLOR 2
#define PNG_CRC_WARN_DISCARD 2
#define PNG_EQUATION_ARBITRARY 2
#define PNG_FILTER_HEURISTIC_WEIGHTED 2
#define PNG_FILTER_VALUE_UP 2
#define PNG_HANDLE_CHUNK_IF_SAFE 2
#define PNG_INTERLACE_LAST 2
#define PNG_ITXT_COMPRESSION_zTXt 2
#define PNG_LIBPNG_BUILD_BETA 2
#define PNG_LIBPNG_VER_MINOR 2
#define PNG_OFFSET_LAST 2
#define PNG_RESOLUTION_LAST 2
#define PNG_SCALE_RADIAN 2
#define PNG_SELECT_WRITE 2
#define PNG_USER_WILL_FREE_DATA 2
#define PNG_sRGB_INTENT_SATURATION 2
#define PNG_MAX_PALETTE_LENGTH 256
#define PNG_BACKGROUND_GAMMA_UNIQUE 3
#define PNG_CRC_WARN_USE 3
#define PNG_EQUATION_HYPERBOLIC 3
#define PNG_FILTER_HEURISTIC_LAST 3
#define PNG_FILTER_VALUE_AVG 3
#define PNG_HANDLE_CHUNK_ALWAYS 3
#define PNG_LIBPNG_BUILD_RC 3
#define PNG_SCALE_LAST 3
#define PNG_TEXT_COMPRESSION_LAST 3
#define PNG_sRGB_INTENT_ABSOLUTE 3

```

```

#define PNG_LIBPNG_BUILD_SPECIAL          32
#define PNG_COLOR_MASK_ALPHA            4
#define PNG_CRC_QUIET_USE                4
#define PNG_EQUATION_LAST                4
#define PNG_FILTER_VALUE_PAETH           4
#define PNG_LIBPNG_BUILD_STABLE           4
#define PNG_sRGB_INTENT_LAST              4
#define PNG_CRC_NO_CHANGE                 5
#define PNG_FILTER_VALUE_LAST             5
#define PNG_INTRAPIXEL_DIFFERENCING       64
#define PNG_LIBPNG_BUILD_RELEASE_STATUS_MASK 7
#define PNG_KEYWORD_MAX_LENGTH            79
#define PNG_LIBPNG_BUILD_PATCH            8
#define PNG_LIBPNG_VER_RELEASE             8
#define PNG_ZBUF_SIZE                      8192
#define PNG_ABORT()                       abort()
#define PNG_CONST                         const
#define PNG_EXPORT_VAR(type)              extern PNG_IMPEXP type
#define png_memcmp                        memcmp
#define png_memcpy                        memcpy
#define png_memset                        memset
#define PNG_COLOR_TYPE_GA                 PNG_COLOR_TYPE_GRAY_ALPHA
#define PNG_COLOR_TYPE_RGBA               PNG_COLOR_TYPE_RGB_ALPHA
#define PNG_COMPRESSION_TYPE_DEFAULT       PNG_COMPRESSION_TYPE_BASE
#define PNG_FILTER_TYPE_DEFAULT            PNG_FILTER_TYPE_BASE
#define PNG_EXPORT(type,symbol) PNG_IMPEXP type PNGAPI symbol
#define PNG_LIBPNG_BUILD_TYPE              PNG_LIBPNG_BUILD_BASE_TYPE
#define PNG_LIBPNG_BUILD_BASE_TYPE         PNG_LIBPNG_BUILD_STABLE
#define PNG_MAX_UINT                      PNG_UINT_31_MAX
#define png_sizeof(x)                     sizeof (x)
#define png_strcpy                        strcpy
#define png_strlen                        strlen
#define png_strncpy                       strncpy

typedef png_unknown_chunk **png_unknown_chunkpp;
typedef struct png_sPLT_entry_struct {
    png_uint_16 red;
    png_uint_16 green;
    png_uint_16 blue;
    png_uint_16 alpha;
    png_uint_16 frequency;
} png_sPLT_entry;
typedef png_sPLT_entry *png_sPLT_entryp;
typedef png_sPLT_entry **png_sPLT_entrypp;
typedef struct png_sPLT_struct {
    png_charp name;
    png_byte depth;
    png_sPLT_entryp entries;
    png_int_32 nentries;
} png_sPLT_t;
typedef png_sPLT_t *png_sPLT_tp;
typedef png_sPLT_t **png_sPLT_tpp;
typedef struct png_unknown_chunk_t {
    png_byte name[5];
    png_byte *data;
    png_size_t size;
    png_byte location;
} png_unknown_chunk;
typedef png_unknown_chunk *png_unknown_chunkp;
typedef struct png_struct_def {
    struct __jmp_buf_tag jmpbuf[1];
} png_struct;
typedef png_struct *png_structp;
typedef struct png_info_struct {
    png_uint_32 width;
    png_uint_32 height;

```

```

png_uint_32 valid;
png_uint_32 rowbytes;
png_colorp palette;
png_uint_16 num_palette;
png_uint_16 num_trans;
png_byte bit_depth;
png_byte color_type;
png_byte compression_type;
png_byte filter_type;
png_byte interlace_type;
png_byte channels;
png_byte pixel_depth;
png_byte spare_byte;
png_byte signature[8];
float gamma;
png_byte srgb_intent;
int num_text;
int max_text;
png_textp text;
png_time mod_time;
png_color_8 sig_bit;
png_bytep trans;
png_color_16 trans_values;
png_color_16 background;
png_int_32 x_offset;
png_int_32 y_offset;
png_byte offset_unit_type;
png_uint_32 x_pixels_per_unit;
png_uint_32 y_pixels_per_unit;
png_byte phys_unit_type;
png_uint_16p hist;
float x_white;
float y_white;
float x_red;
float y_red;
float x_green;
float y_green;
float x_blue;
float y_blue;
png_charp pcal_purpose;
png_int_32 pcal_X0;
png_int_32 pcal_X1;
png_charp pcal_units;
png_charpp pcal_params;
png_byte pcal_type;
png_byte pcal_nparams;
png_uint_32 free_me;
png_unknown_chunkp unknown_chunks;
png_size_t unknown_chunks_num;
png_charp iccp_name;
png_charp iccp_profile;
png_uint_32 iccp_proflen;
png_byte iccp_compression;
png_sPLT_tp splt_palettes;
png_uint_32 splt_palettes_num;
png_byte scal_unit;
double scal_pixel_width;
double scal_pixel_height;
png_charp scal_s_width;
png_charp scal_s_height;
png_bytepp row_pointers;
png_fixed_point int_gamma;
png_fixed_point int_x_white;
png_fixed_point int_y_white;
png_fixed_point int_x_red;
png_fixed_point int_y_red;

```

```

    png_fixed_point int_x_green;
    png_fixed_point int_y_green;
    png_fixed_point int_x_blue;
    png_fixed_point int_y_blue;
} png_info;
typedef png_info *png_infop;
typedef void *png_voidp;
typedef void (*png_progressive_info_ptr) (png_structp, png_infop);
typedef unsigned char png_byte;
typedef png_byte *png_bytep;
typedef unsigned long int png_uint_32;
typedef void (*png_progressive_row_ptr) (png_structp, png_bytep,
                                         png_uint_32, int);
typedef void (*png_progressive_end_ptr) (png_structp, png_infop);
typedef struct png_color_8_struct {
    png_byte red;
    png_byte green;
    png_byte blue;
    png_byte gray;
    png_byte alpha;
} png_color_8;
typedef png_color_8 *png_color_8p;
typedef long int png_int_32;
typedef char *png_charp;
typedef size_t png_size_t;
typedef struct png_text_struct {
    int compression;
    png_charp key;
    png_charp text;
    png_size_t text_length;
} png_text;
typedef png_text *png_textp;
typedef png_byte **png_bytepp;
typedef unsigned short png_uint_16;
typedef struct png_color_16_struct {
    png_byte index;
    png_uint_16 red;
    png_uint_16 green;
    png_uint_16 blue;
    png_uint_16 gray;
} png_color_16;
typedef png_color_16 *png_color_16p;
typedef struct png_color_struct {
    png_byte red;
    png_byte green;
    png_byte blue;
} png_color;
typedef png_color *png_colorp;
typedef const char *png_const_charp;
typedef void (*png_error_ptr) (png_structp, png_const_charp);
typedef void (*png_rw_ptr) (png_structp, png_bytep, png_size_t);
typedef struct png_time_struct {
    png_uint_16 year;
    png_byte month;
    png_byte day;
    png_byte hour;
    png_byte minute;
    png_byte second;
} png_time;
typedef png_time *png_timep;
typedef png_uint_16 *png_uint_16p;
typedef void (*png_flush_ptr) (png_structp);
typedef char **png_charpp;
typedef png_struct **png_structpp;
typedef png_info **png_infopp;
typedef FILE *png_FILE_p;

```

```

typedef struct png_row_info_struct {
    png_uint_32 width;
    png_uint_32 rowbytes;
    png_byte color_type;
    png_byte bit_depth;
    png_byte channels;
    png_byte pixel_depth;
} png_row_info;
typedef png_row_info *png_row_infop;
typedef png_structp version_1_2_8;
typedef png_uint_32 *png_uint_32p;
typedef png_uint_16 **png_uint_16pp;
typedef png_int_32 png_fixed_point;
typedef double *png_doublep;
typedef charf *png_zcharp;
typedef png_fixed_point *png_fixed_point_p;
typedef png_int_32 *png_int_32p;
typedef z_stream *png_zstreamp;
typedef short png_int_16;
typedef png_int_16 *png_int_16p;
typedef png_int_16 **png_int_16pp;
typedef png_int_32 **png_int_32pp;
typedef png_uint_32 **png_uint_32pp;
typedef charf **png_zcharpp;
typedef char ***png_charppp;
typedef const char **png_const_charpp;
typedef double **png_doublepp;
typedef png_colorp *png_colorpp;
typedef png_color_16p *png_color_16pp;
typedef png_color_8p *png_color_8pp;
typedef png_fixed_point **png_fixed_point_pp;
typedef png_row_info **png_row_infopp;
typedef png_textp *png_textpp;
typedef png_timep *png_timepp;
typedef void (*png_user_transform_ptr) (png_structp, png_row_infop,
                                       png_bytep);
typedef void (*png_read_status_ptr) (png_structp, png_uint_32, int);
typedef void (*png_write_status_ptr) (png_structp, png_uint_32,
int);
typedef int (*png_user_chunk_ptr) (png_structp,
png_unknown_chunkp);
typedef png_voidp (*png_malloc_ptr) (png_structp, png_size_t);
typedef void (*png_free_ptr) (png_structp, png_voidp);
extern png_uint_32 png_access_version_number(void);
extern int png_check_sig(png_bytep sig, int num);
extern void png_convert_from_struct_tm(png_timep ptime, struct tm
*ttime);
extern void png_convert_from_time_t(png_timep ptime, time_t ttime);
extern png_infop png_create_info_struct(png_structp png_ptr);
extern png_structp png_create_read_struct(png_const_charp
user_png_ver,
                                       png_voidp error_ptr,
                                       png_error_ptr error_fn,
                                       png_error_ptr warn_fn);
extern png_structp png_create_read_struct_2(png_const_charp
user_png_ver,
                                       png_voidp error_ptr,
                                       png_error_ptr error_fn,
                                       png_error_ptr warn_fn,
                                       png_voidp mem_ptr,
                                       png_malloc_ptr malloc_fn,
                                       png_free_ptr free_fn);
extern png_structp png_create_write_struct(png_const_charp
user_png_ver,
                                       png_voidp error_ptr,
                                       png_error_ptr error_fn,

```

```

                                png_error_ptr warn_fn);
extern      png_structp      png_create_write_struct_2(png_const_charp
user_png_ver,

                                png_voidp error_ptr,
                                png_error_ptr error_fn,
                                png_error_ptr warn_fn,
                                png_voidp mem_ptr,
                                png_malloc_ptr malloc_fn,
                                png_free_ptr free_fn);
extern void png_data_freer(png_structp png_ptr, png_infop info_ptr,
                           int freer, png_uint_32 mask);
extern void png_destroy_info_struct(png_structp png_ptr,
                                   png_infopp info_ptr_ptr);
extern void png_destroy_read_struct(png_structpp png_ptr_ptr,
                                   png_infopp info_ptr_ptr,
                                   png_infopp end_info_ptr_ptr);
extern void png_destroy_write_struct(png_structpp png_ptr_ptr,
                                   png_infopp info_ptr_ptr);
extern      void      png_error(png_structp png_ptr, png_const_charp
error_message);
extern void png_free(png_structp png_ptr, png_voidp ptr);
extern void png_free_data(png_structp png_ptr, png_infop info_ptr,
                           png_uint_32 free_me, int num);
extern png_uint_32 png_get_IHDR(png_structp png_ptr, png_infop
info_ptr,
                                png_uint_32 * width, png_uint_32 *
height,
                                int *bit_depth, int *color_type,
                                int *interlace_method,
                                int *compression_method,
                                int *filter_method);
extern png_uint_32 png_get_PLTE(png_structp png_ptr, png_infop
info_ptr,
                                png_colorp * palette, int *num_palette);
extern png_uint_32 png_get_bKGD(png_structp png_ptr, png_infop
info_ptr,
                                png_color_16p * background);
extern png_byte png_get_bit_depth(png_structp png_ptr, png_infop
info_ptr);
extern png_uint_32 png_get_CHRM(png_structp png_ptr, png_infop
info_ptr,
                                double *white_x, double *white_y,
                                double *red_x, double *red_y,
                                double *green_x, double *green_y,
                                double *blue_x, double *blue_y);
extern png_byte png_get_channels(png_structp png_ptr, png_infop
info_ptr);
extern png_byte png_get_color_type(png_structp png_ptr,
                                   png_infop info_ptr);
extern png_voidp png_get_error_ptr(png_structp png_ptr);
extern png_uint_32 png_get_gAMA(png_structp png_ptr, png_infop
info_ptr,
                                double *file_gamma);
extern png_uint_32 png_get_hIST(png_structp png_ptr, png_infop
info_ptr,
                                png_uint_16p * hist);
extern png_charp png_get_header_ver(png_structp png_ptr);
extern png_uint_32 png_get_iCCP(png_structp png_ptr, png_infop
info_ptr,
                                png_charpp name, int *compression_type,
                                png_charpp profile, png_uint_32 *
proflen);
extern png_uint_32 png_get_image_height(png_structp png_ptr,
                                         png_infop info_ptr);
extern png_uint_32 png_get_image_width(png_structp png_ptr,
                                         png_infop info_ptr);

```

```

extern png_byte png_get_interlace_type(png_structp png_ptr,
                                       png_infop info_ptr);
extern png_voidp png_get_io_ptr(png_structp png_ptr);
extern png_charp png_get_libpng_ver(png_structp png_ptr);
extern png_uint_32 png_get_oFFs(png_structp png_ptr, png_infop
info_ptr,
                               png_int_32 * offset_x,
                               png_int_32 * offset_y, int *unit_type);
extern png_uint_32 png_get_pHYs(png_structp png_ptr, png_infop
info_ptr,
                               png_uint_32 * res_x, png_uint_32 * res_y,
                               int *unit_type);
extern png_voidp png_get_progressive_ptr(png_structp png_ptr);
extern png_uint_32 png_get_rowbytes(png_structp png_ptr,
                                     png_infop info_ptr);
extern png_bytepp png_get_rows(png_structp png_ptr, png_infop
info_ptr);
extern png_uint_32 png_get_sBIT(png_structp png_ptr, png_infop
info_ptr,
                               png_color_8p * sig_bit);
extern png_uint_32 png_get_sRGB(png_structp png_ptr, png_infop
info_ptr,
                               int *intent);
extern png_uint_32 png_get_tIME(png_structp png_ptr, png_infop
info_ptr,
                               png_timep * mod_time);
extern png_uint_32 png_get_tRNS(png_structp png_ptr, png_infop
info_ptr,
                               png_bytep * trans, int *num_trans,
                               png_color_16p * trans_values);
extern png_uint_32 png_get_text(png_structp png_ptr, png_infop
info_ptr,
                               png_textp * text_ptr, int *num_text);
extern png_uint_32 png_get_unknown_chunks(png_structp png_ptr,
                                           png_infop info_ptr,
                                           png_unknown_chunkpp entries);
extern png_voidp png_get_user_chunk_ptr(png_structp png_ptr);
extern png_uint_32 png_get_valid(png_structp png_ptr, png_infop
info_ptr,
                               png_uint_32 flag);
extern png_int_32 png_get_x_offset_pixels(png_structp png_ptr,
                                           png_infop info_ptr);
extern png_uint_32 png_get_x_pixels_per_meter(png_structp png_ptr,
                                                png_infop info_ptr);
extern png_int_32 png_get_y_offset_pixels(png_structp png_ptr,
                                           png_infop info_ptr);
extern png_uint_32 png_get_y_pixels_per_meter(png_structp png_ptr,
                                                png_infop info_ptr);
extern void png_info_init_3(png_infopp info_ptr,
                           png_size_t png_info_struct_size);
extern void png_init_io(png_structp png_ptr, png_FILE_p fp);
extern const char png_libpng_ver[];
extern png_voidp png_malloc(png_structp png_ptr, png_uint_32 size);
extern png_uint_32 png_permit_mng_features(png_structp png_ptr,
                                           png_uint_32
                                           mng_features_permitted);
extern void png_process_data(png_structp png_ptr, png_infop
info_ptr,
                           png_bytep buffer, png_size_t buffer_size);
extern void png_progressive_combine_row(png_structp png_ptr,
                                        png_bytep old_row,
                                        png_bytep new_row);
extern void png_read_end(png_structp png_ptr, png_infop info_ptr);
extern void png_read_image(png_structp png_ptr, png_bytepp image);
extern void png_read_info(png_structp png_ptr, png_infop info_ptr);
extern void png_read_png(png_structp png_ptr, png_infop info_ptr,

```

```

        int transforms, voidp params);
extern void png_read_row(png_structp png_ptr, png_bytep row,
        png_bytep display_row);
extern void png_read_rows(png_structp png_ptr, png_bytepp row,
        png_bytepp display_row, png_uint_32
num_rows);
extern void png_read_update_info(png_structp png_ptr, png_infop
info_ptr);
extern void png_set_IHDR(png_structp png_ptr, png_infop info_ptr,
        png_uint_32 width, png_uint_32 height,
        int bit_depth, int color_type,
        int interlace_method, int compression_method,
        int filter_method);
extern void png_set_PLTE(png_structp png_ptr, png_infop info_ptr,
        png_colorp palette, int num_palette);
extern void png_set_bKGD(png_structp png_ptr, png_infop info_ptr,
        png_color_16p background);
extern void png_set_background(png_structp png_ptr,
        png_color_16p background_color,
        int background_gamma_code, int
need_expand,
        double background_gamma);
extern void png_set_bgr(png_structp png_ptr);
extern void png_set_cHRM(png_structp png_ptr, png_infop info_ptr,
        double white_x, double white_y, double red_x,
        double red_y, double green_x, double green_y,
        double blue_x, double blue_y);
extern void png_set_compression_buffer_size(png_structp png_ptr,
        png_uint_32 size);
extern void png_set_compression_level(png_structp png_ptr, int
level);
extern void png_set_compression_mem_level(png_structp png_ptr,
        int mem_level);
extern void png_set_compression_method(png_structp png_ptr, int
method);
extern void png_set_compression_strategy(png_structp png_ptr,
        int strategy);
extern void png_set_compression_window_bits(png_structp png_ptr,
        int window_bits);
extern void png_set_dither(png_structp png_ptr, png_colorp palette,
        int num_palette, int maximum_colors,
        png_uint_16p histogram, int full_dither);
extern void png_set_error_fn(png_structp png_ptr, png_voidp
error_ptr,
        png_error_ptr error_fn,
        png_error_ptr warning_fn);
extern void png_set_expand(png_structp png_ptr);
extern void png_set_filler(png_structp png_ptr, png_uint_32 filler,
        int flags);
extern void png_set_filter(png_structp png_ptr, int method, int
filters);
extern void png_set_gAMA(png_structp png_ptr, png_infop info_ptr,
        double file_gamma);
extern void png_set_gamma(png_structp png_ptr, double screen_gamma,
        double default_file_gamma);
extern void png_set_gray_1_2_4_to_8(png_structp png_ptr);
extern void png_set_gray_to_rgb(png_structp png_ptr);
extern void png_set_hIST(png_structp png_ptr, png_infop info_ptr,
        png_uint_16p hist);
extern void png_set_iCCP(png_structp png_ptr, png_infop info_ptr,
        png_charp name, int compression_type,
        png_charp profile, png_uint_32 proflen);
extern int png_set_interlace_handling(png_structp png_ptr);
extern void png_set_invert_alpha(png_structp png_ptr);
extern void png_set_invert_mono(png_structp png_ptr);

```



```

extern void png_set_keep_unknown_chunks(png_structp png_ptr, int
keep,
                                png_bytep chunk_list,
                                int num_chunks);
extern void png_set_mem_fn(png_structp png_ptr, png_voidp mem_ptr,
                                png_malloc_ptr malloc_fn, png_free_ptr
free_fn);
extern void png_set_oFFs(png_structp png_ptr, png_infop info_ptr,
                                png_int_32 offset_x, png_int_32 offset_y,
                                int unit_type);
extern void png_set_pHYs(png_structp png_ptr, png_infop info_ptr,
                                png_uint_32 res_x, png_uint_32 res_y,
                                int unit_type);
extern void png_set_packing(png_structp png_ptr);
extern void png_set_packswap(png_structp png_ptr);
extern void png_set_palette_to_rgb(png_structp png_ptr);
extern void png_set_progressive_read_fn(png_structp png_ptr,
                                png_voidp progressive_ptr,
                                png_progressive_info_ptr info_fn,
                                png_progressive_row_ptr row_fn,
                                png_progressive_end_ptr end_fn);
extern void png_set_read_fn(png_structp png_ptr, png_voidp io_ptr,
                                png_rw_ptr read_data_fn);
extern void png_set_read_user_chunk_fn(png_structp png_ptr,
                                png_voidp user_chunk_ptr,
                                png_user_chunk_ptr
read_user_chunk_fn);
extern void png_set_read_user_transform_fn(png_structp png_ptr,
                                png_user_transform_ptr
read_user_transform_fn);
extern void png_set_rgb_to_gray(png_structp png_ptr, int
error_action,
                                double red, double green);
extern void png_set_rows(png_structp png_ptr, png_infop info_ptr,
                                png_bytepp row_pointers);
extern void png_set_sBIT(png_structp png_ptr, png_infop info_ptr,
                                png_color_8p sig_bit);
extern void png_set_sRGB(png_structp png_ptr, png_infop info_ptr,
                                int intent);
extern void png_set_sRGB_gAMA_and_cHRM(png_structp png_ptr,
                                png_infop info_ptr, int intent);
extern void png_set_shift(png_structp png_ptr, png_color_8p
true_bits);
extern void png_set_sig_bytes(png_structp png_ptr, int num_bytes);
extern void png_set_strip_16(png_structp png_ptr);
extern void png_set_strip_alpha(png_structp png_ptr);
extern void png_set_swap(png_structp png_ptr);
extern void png_set_swap_alpha(png_structp png_ptr);
extern void png_set_tIME(png_structp png_ptr, png_infop info_ptr,
                                png_timestp mod_time);
extern void png_set_tRNS(png_structp png_ptr, png_infop info_ptr,
                                png_bytep trans, int num_trans,
                                png_color_16p trans_values);
extern void png_set_tRNS_to_alpha(png_structp png_ptr);
extern void png_set_text(png_structp png_ptr, png_infop info_ptr,
                                png_textp text_ptr, int num_text);
extern void png_set_unknown_chunk_location(png_structp png_ptr,
                                png_infop info_ptr, int chunk,
                                int location);
extern void png_set_unknown_chunks(png_structp png_ptr, png_infop
info_ptr,
                                png_unknown_chunkp unknowns,
                                int num_unknowns);
extern void png_set_write_fn(png_structp png_ptr, png_voidp io_ptr,
                                png_rw_ptr write_data_fn,
                                png_flush_ptr output_flush_fn);

```

```

extern void png_set_write_status_fn(png_structp png_ptr,
                                   png_write_status_ptr write_row_fn);
extern void png_set_write_user_transform_fn(png_structp png_ptr,
                                           png_user_transform_ptr
                                           write_user_transform_fn);
extern int png_sig_cmp(png_bytep sig, png_size_t start,
                      png_size_t num_to_check);
extern void png_start_read_image(png_structp png_ptr);
extern void png_warning(png_structp png_ptr,
                      png_const_charp warning_message);
extern void png_write_chunk(png_structp png_ptr, png_bytep
                           chunk_name,
                           png_bytep data, png_size_t length);
extern void png_write_end(png_structp png_ptr, png_infop info_ptr);
extern void png_write_flush(png_structp png_ptr);
extern void png_write_image(png_structp png_ptr, png_bytepp image);
extern void png_write_info(png_structp png_ptr, png_infop info_ptr);
extern void png_write_png(png_structp png_ptr, png_infop info_ptr,
                          int transforms, voidp params);
extern void png_write_row(png_structp png_ptr, png_bytep row);
extern void png_write_rows(png_structp png_ptr, png_bytepp row,
                          png_uint_32 num_rows);

```

8.3 Interface Definitions for libpng12

The interfaces defined on the following pages are included in libpng12 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 8.1 shall behave as described in the referenced base document.

png_access_version_number

Name

png_access_version_number — return version of the run-time library

Synopsis

```

#include <png.h>
png_uint_32 png_access_version_number(void);

```

Description

png_access_version_number() returns version of the libpng12 library available at run-time.

Return Value

png_access_version_number() returns integer constructed from the major version, minor version with leading zero and leading number with leading zero. For example, the version number for version 1.2.8 is 10208.

png_check_sig**Name**

png_check_sig — INSERT PURPOSE HERE

Synopsis

```
#include <png.h>
int png_check_sig(png_bytep arg0, int arg1);
```

Description

INSERT TEXT HERE

Return Value

INSERT TEXT HERE

Errors

INSERT TEXT HERE

png_convert_from_struct_tm**Name**

png_convert_from_struct_tm — convert struct tm to png_time

Synopsis

```
#include <png.h>
void png_convert_from_struct_tm(png_timep ptime, struct tm * ttime);
```

Description

This interface shall convert from the struct tm time format to the png_time format.

The parameter *ttime* shall specify the struct tm value to convert to png_time format.

The output parameter *ptime* shall contain the converted value.

png_convert_from_time_t**Name**

png_convert_from_time_t — convert time_t to png_time

Synopsis

```
#include <png.h>
void png_convert_from_time_t(png_timep ptime, time_t ttime);
```

Description

This interface shall convert from the time_t time format to the png_time format.

The parameter *ttime* shall specify the time_t value to convert to png_time format.

The output parameter *ptime* shall contain the converted value.

png_create_info_struct

Name

png_create_info_struct — allocate and initialize a png_info structure

Synopsis

```
#include <png.h>
png_info* png_create_info_struct(png_structp png_ptr);
```

Description

png_create_info_struct() shall allocate and initialize a png_info structure.

Return Value

Returns the pointer to png_info structure. Returns NULL if it fails to create the structure.

Errors

png_create_info_struct() shall return NULL if the allocation fails. The application should check for the return value.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

png_create_read_struct

Name

`png_create_read_struct` — allocate and initialize a `png_struct` structure for reading PNG file

Synopsis

```
#include <png.h>
png_structp png_create_read_struct(png_const_charp user_png_ver,
png_voidp error_ptr, png_error_ptr error_fn, png_error_ptr warn_fn);
```

Description

`png_create_read_struct()` shall allocate and initialize a `png_struct` structure. The function shall return NULL if the allocation fails. The application should check for the return value. For handling errors and warnings, the application can pass desired error handling routines as arguments to `png_create_read_struct()`. Otherwise, the default error handling uses `stderr` and `longjmp`. The error handling routine must NOT return to the calling routine.

user_png_ver

version string of the library. Must be `PNG_LIBPNG_VER_STRING`

error_ptr

user defined struct for error functions.

error_fn

user defined function for printing errors and aborting.

warn_fn

user defined function for warnings.

Return Value

Returns the pointer to `png_struct` structure. Returns NULL if it fails to create the structure.

Errors

`png_create_read_struct()` shall return NULL if the allocation fails. The application should check for the return value.

png_create_read_struct_2**Name**

`png_create_read_struct_2` — register custom read function

Synopsis

```
#include <png.h>
png_structp png_create_read_struct_2(png_const_charp user_png_ver,
png_voidp error_ptr, png_error_ptr error_fn, png_error_ptr warn_fn,
png_voidp mem_ptr, png_malloc_ptr malloc_fn, png_free_ptr free_fn);
```

Description

This interface shall register a custom read function and allocate and initialize a `png_struct` structure. For handling errors and warnings, as well as allocating and deallocating memory, the application can pass functions as arguments. Otherwise, the default error handling functions `stderr` and `longjmp()` will be used. The error handling routine must not return to the calling routine.

The parameter `user_png_ver` shall specify the version string of the library, which must be `PNG_LIBPNG_VER_STRING`.

The parameter `error_ptr` shall specify a user-defined structure for error functions.

The parameter `error_fn` shall specify an optional user-defined function for printing errors and aborting.

The parameter `warn_fn` shall specify an optional user-defined function for warnings.

The parameter `mem_ptr` shall specify the memory to allocate.

The parameter `malloc_fn` shall specify an optional user-defined memory allocation function.

The parameter `free_fn` shall specify an optional user-defined memory deallocation function.

Application Usage (informative)

You should define `PNG_USER_MEM_SUPPORTED` before you call `png_create_read_struct2()`.

png_create_write_struct

Name

`png_create_write_struct` — allocate and initialize a `png_struct` structure for writing PNG file

Synopsis

```
#include <png.h>
png_structp png_create_write_struct(png_const_charp user_png_ver,
png_voidp error_ptr, png_error_ptr error_fn, png_error_ptr warn_fn);
```

Description

`png_create_write_struct()` shall allocate and initialize a `png_struct` structure. The function shall return NULL if the allocation fails. The application should check for the return value.

user_png_ver

version string of the library. Must be PNG_LIBPNG_VER_STRING

error_ptr

user defined struct for error functions.

error_fn

user defined function for printing errors and aborting.

warn_fn

user defined function for warnings.

Return Value

Returns the pointer to `png_struct` structure. Returns NULL if it fails to create the structure.

png_create_write_struct_2

Name

png_create_write_struct_2 — register custom write function

Synopsis

```
#include <png.h>
png_structp png_create_write_struct_2(png_const_charp user_png_ver,
png_voidp error_ptr, png_error_ptr error_fn, png_error_ptr warn_fn,
png_voidp mem_ptr, png_malloc_ptr malloc_fn, png_free_ptr free_fn);
```

Description

This interface shall register a custom write function and allocate and initialize a `png_struct` structure. For handling errors and warnings, as well as allocating and deallocating memory, the application can pass functions as arguments. Otherwise, the default error handling functions `stderr` and `longjmp()` will be used. The error handling routine must not return to the calling routine.

The parameter `user_png_ver` shall specify the version string of the library, which must be `PNG_LIBPNG_VER_STRING`.

The parameter `error_ptr` shall specify a user-defined structure for error functions.

The parameter `error_fn` shall specify an optional user-defined function for printing errors and aborting.

The parameter `warn_fn` shall specify an optional user-defined function for warnings.

The parameter `mem_ptr` shall specify the memory to allocate.

The parameter `malloc_fn` shall specify an optional user-defined memory allocation function.

The parameter `free_fn` shall specify an optional user-defined memory deallocation function.

Return Value

On success, returns a pointer to the `png_struct` that was created.

On failure, returns `NULL`.

png_data_freer

Name

`png_data_freer` — change the default behavior for freeing data

Synopsis

```
#include <png.h>
void png_data_freer(png_structp png_ptr, png_infop info_ptr, int freer,
png_uint_32 mask);
```

Description

This interface shall change the default behavior for freeing allocated data, from only freeing data allocated internally by libpng, to either freeing user-allocated data or to not freeing any data at all.

This interface can enable deallocation of user data allocated with `png_malloc()` or `png_zalloc()` and passed to libpng with one of the `png_set_*` interfaces. Users can call it both before and after passing the data. Calling it after reading PNG data but before passing the data controls whether the user or the interface is responsible for the existing data. Calling it after passing the data controls whether the user or the interface should deallocate the data.

If the user becomes responsible for data allocated by libpng, `png_free()` must be called to free it. If libpng becomes responsible for user-allocated data, the data must only have been allocated with `png_malloc()` or `png_zalloc()`.

The parameter `png_ptr` shall specify the PNG file.

The parameter `info_ptr` shall specify the PNG info structure.

The parameter `freer` shall specify one of these constants:
`PNG_DESTROY_WILL_FREE_DATA`, `PNG_SET_WILL_FREE_DATA`,
`PNG_USER_WILL_FREE_DATA`.

The parameter `mask` shall specify the data to free, as described under `png_free_data()`.

Application Usage (informative)

A `row_pointers` array allocated in a single block must not be freed with `png_set_rows()` or `png_read_destroy()`, because they would attempt to free the elements of the array as well.

Do not free `text_ptr` with libpng if some of its members were allocated separately, because this will actually only free `text_ptr.key`. If responsibility for freeing `text_ptr` moves from libpng to the user, the user must not free the members separately.

png_destroy_info_struct**Name**

`png_destroy_info_struct` — free memory in PNG info structure

Synopsis

```
#include <png.h>
void png_destroy_info_struct(png_structp png_ptr, png_infopp
info_ptr_ptr);
```

Description

This interface shall free any memory in a single PNG info structure.

The parameter `png_ptr` shall specify the PNG file.

The parameter `info_ptr_ptr` shall point to a pointer to the `png_info` struct to destroy.

png_destroy_read_struct**Name**

`png_destroy_read_struct` — free the memory associated with read `png_struct`

Synopsis

```
#include <png.h>
void png_destroy_read_struct(png_structpp png_ptr_ptr, png_infopp
info_ptr_ptr, png_infopp end_info_ptr_ptr);
```

Description

`png_destroy_read_struct()` frees the memory associated with the read `png_struct` struct that holds information from the given PNG file, the associated `png_info` struct for holding the image information and `png_info` struct for holding the information at end of the given PNG file.

png_destroy_write_struct**Name**

`png_destroy_write_struct` — free the memory associated with write `png_struct`

Synopsis

```
#include <png.h>
void png_destroy_write_struct(png_structpp png_ptr_ptr, png_infopp
info_ptr_ptr);
```

Description

`png_destroy_write_struct()` frees the memory associated with the write `png_struct` struct that holds information for writing the PNG file and the associated `png_info` struct for holding the image information.

png_error**Name**

`png_error` — default function to handle fatal errors

Synopsis

```
#include <png.h>
void png_error(png_structp png_ptr, png_const_charp error_message);
```

Description

`png_error()` is the default error handling function for fatal errors. The default error handling functionality may be changed by using `png_set_error_fn()` to replace the error function at run-time.

Return Value**Errors****png_free****Name**

`png_free` — free a pointer allocated by `png_malloc()`

Synopsis

```
#include <png.h>
void png_free(png_structp png_ptr, png_voidp ptr);
```

Description

`png_free()` shall free memory pointed to by "ptr" previously allocated by `png_malloc()`.

png_free_data**Name**

`png_free_data` — free internally allocated data

Synopsis

```
#include <png.h>
void    png_free_data(png_structp    png_ptr,    png_infop    info_ptr,
png_uint_32 free_me, int num);
```

Description

This interface shall free data that was internally allocated by libpng.

The parameter *png_ptr* shall specify the PNG file.

The parameter *info_ptr* shall specify the PNG info structure containing members to free that are pointing to memory allocated by libpng.

The parameter *mask* shall specify the data to free with a mask consisting of the logical OR of at least one of the following constants: `PNG_FREE_HIST`, `PNG_FREE_ICCP`, `PNG_FREE_PCAL`, `PNG_FREE_PLTE`, `PNG_FREE_ROWS`, `PNG_FREE_SCAL`, `PNG_FREE_SPLT`, `PNG_FREE_TEXT`, `PNG_FREE_TRNS`, `PNG_FREE_UNKN`. (To specify all constants, use `PNG_FREE_ALL`.)

The parameter *num* shall specify the sequence number of the item to free. The value -1 specifies all items.

png_get_IHDR

Name

png_get_IHDR — get PNG_IHDR chunk information from png_info structure

Synopsis

```
#include <png.h>
png_uint_32 png_get_IHDR(png_structp png_ptr, png_infop info_ptr,
png_uint_32 * width, png_uint_32 * height, int * bit_depth, int *
color_type, int * interlace_method, int * compression_method, int *
filter_method);
```

Description

png_get_IHDR() gets PNG_IHDR chunk type information from png_info structure.

width

holds the width of the image in pixels up to 2^{31} .

height

holds the height of the image in pixels up to 2^{31} .

bit_depth

holds the bit depth of one of the image channels. Valid values are 1, 2, 4, 8, 16 and also depend on the color_type

color_type

describes which color/alpha channels are present. Supported color types shall include:

PNG_COLOR_TYPE_GRAY (bit depths 1, 2, 4, 8, 16)
 PNG_COLOR_TYPE_GRAY_ALPHA (bit depths 8, 16)
 PNG_COLOR_TYPE_PALETTE (bit depths 1, 2, 4, 8)
 PNG_COLOR_TYPE_RGB (bit depths 8, 16)
 PNG_COLOR_TYPE_RGB_ALPHA (bit depths 8, 16)
 PNG_COLOR_MASK_PALETTE
 PNG_COLOR_MASK_COLOR
 PNG_COLOR_MASK_ALPHA

filter_method

holds the filter method. If this argument is NULL, the filter method will not be retrieved. Valid values after retrieval are

PNG_FILTER_TYPE_BASE
 PNG_INTRAPIXEL_DIFFERENCING

compression_method

holds the compression method. If this argument is NULL, the compression method will not be retrieved. Valid values after retrieval are PNG_COMPRESSION_TYPE_BASE

interlace_method

holds the interlace method. If this argument is NULL, the interlacing method will not be retrieved. Valid values after retrieval are

PNG_INTERLACE_NONE
PNG_INTERLACE_ADAM7

Return Value

On success, `png_get_HDR()` shall return 1. Otherwise, `png_get_IHDR()` shall return 0.

png_get_PLTE

Name

`png_get_PLTE` — get image palette information from `png_info` structure

Synopsis

```
#include <png.h>
png_uint_32 png_get_PLTE(png_structp png_ptr, png_info_ptr info_ptr,
png_colorp * palette, int * num_palette);
```

Description

`png_get_PLTE()` gets palette information from `png_info` structure. "palette" holds an array of color values with "num_palette" entries.

Return Value

On successful retrieval of palette information, `png_get_PLTE()` shall return `PNG_INFO_PLTE`. Otherwise, `png_get_PLTE()` shall return 0.

png_get_bKGD

Name

`png_get_bKGD` — get background color for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_bKGD(png_structp png_ptr, png_info_ptr info_ptr,
png_color_16p * background);
```

Description

`png_get_bKGD()` shall return the background color to "background" if the validity flag for background is set.

Return Value

On success, `png_get_bKGD()` shall return `PNG_INFO_bKGD`. Otherwise, `png_get_bKGD()` shall return 0.

png_get_bit_depth

Name

png_get_bit_depth — return image bit_depth

Synopsis

```
#include <png.h>
png_byte png_get_bit_depth(png_structp png_ptr, png_infop info_ptr);
```

Description

Returns the image bit_depth.

Return Value

Returns 0 if png_ptr or info_ptr is NULL, bit_depth otherwise.

png_get_cHRM

Name

png_get_cHRM — get CIE chromacities and referenced white point for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_cHRM(png_structp png_ptr, png_infop info_ptr,
double * white_x, double * white_y, double * red_x, double * red_y,
double * green_x, double * green_y, double * blue_x, double * blue_y);
```

Description

png_get_cHRM() shall return the CIE x,y chromaticities of the red, green and blue display primaries used in the image and the referenced white point from the cHRM chunk in the image.

Return Value

On success, png_get_cHRM() shall return PNG_INFO_cHRM. Otherwise, the function shall return 0.

png_get_channels

Name

png_get_channels — get number of color channels in image

Synopsis

```
#include <png.h>
png_byte png_get_channels(png_structp png_ptr, png_infop info_ptr);
```

Description

png_get_channels() shall return the number of data channels per pixel for the color type of the image. The number of channels shall range from 1-4 depending on the color type as given below.

- 1 - PNG_COLOR_TYPE_GRAY or PNG_COLOR_TYPE_PALETTE
- 2 - PNG_COLOR_TYPE_GRAY_ALPHA
- 3 - PNG_COLOR_TYPE_RGB
- 4 - PNG_COLOR_TYPE_RGB_ALPHA or PNG_COLOR_TYPE_RGB+filler
byte

Return Value

On success, `png_get_channels()` shall return the number of channels ranging from 1-4. Otherwise, `png_get_channels` shall return 0.

png_get_color_type

Name

`png_get_color_type` — return image color type

Synopsis

```
#include <png.h>
png_byte png_get_color_type(png_structp png_ptr, png_info_ptr info_ptr);
```

Description

Returns the image color type.

Return Value

Returns 0 if `png_ptr` or `info_ptr` is NULL, `color_type` otherwise.

png_get_error_ptr

Name

`png_get_error_ptr` — return `error_ptr` for user-defined functions

Synopsis

```
#include <png.h>
png_voidp png_get_error_ptr(png_structp png_ptr);
```

Description

`png_get_error_ptr()` returns the `error_ptr` associated with user-defined functions.

Return Value

Returns `error_ptr`

png_get_gAMA**Name**

png_get_gAMA — get the gamma value for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_gAMA(png_structp png_ptr, png_infop info_ptr,
double * file_gamma);
```

Description

Returns the gamma value of an image to "file_gamma" if the gAMA chunk information is valid for the image.

Return Value

PNG_INFO_gAMA - if png_ptr, info_ptr, file_gamma are not NULL and gAMA chunk information is valid 0 - otherwise.

png_get_hIST**Name**

png_get_hIST — get the histogram for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_hIST(png_structp png_ptr, png_infop info_ptr,
png_uint_16p * hist);
```

Description

Returns the histogram of an image to *hist if the hIST chunk information is valid for the image.

Return Value

PNG_INFO_hIST - if png_ptr, info_ptr, hist are not NULL and hIST chunk information is valid 0 - otherwise.

png_get_header_ver**Name**

png_get_header_ver — get version information for libpng header files

Synopsis

```
#include <png.h>
png_charp png_get_header_ver(png_structp png_ptr);
```

Description

This interface shall return the version of the header files used when building libpng as a short string in the format "1.0.0" through "99.99.99zz".

png_get_iCCP**Name**

png_get_iCCP — get the embedded ICC profile data for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_iCCP(png_structp png_ptr, png_infop info_ptr,
png_charpp name, int * compression_type, png_charpp profile,
png_uint_32 * proflen);
```

Description

png_get_iCCP() shall return the embedded ICC profile data in iCCP chunk. "name" shall contain the profile name, *compression_type shall contain the compression type, profile shall contain the International Color Consortium color profile data and *proflen shall contain the length of the profile data in bytes. *compression_type must always be set to PNG_COMPRESSION_TYPE_BASE.

Return Value

On success, png_get_iCCP() shall return PNG_INFO_iCCP. Otherwise, the function shall return 0.

png_get_image_height**Name**

png_get_image_height — return image height

Synopsis

```
#include <png.h>
png_uint_32 png_get_image_height(png_structp png_ptr, png_infop info_ptr);
```

Description

png_get_image_height() returns the image height in pixels.

Return Value

Returns 0 if png_ptr or info_ptr is NULL, image_height otherwise.

png_get_image_width**Name**

png_get_image_width — return image width

Synopsis

```
#include <png.h>
png_uint_32 png_get_image_width(png_structp png_ptr, png_infop
info_ptr);
```

Description

png_get_image_width() returns the image width in pixels.

Return Value

Returns 0 if png_ptr or info_ptr is NULL, image_width otherwise.

png_get_interlace_type**Name**

png_get_interlace_type — returns interlace method

Synopsis

```
#include <png.h>
png_byte png_get_interlace_type(png_structp png_ptr, png_infop
info_ptr);
```

Description

png_get_interlace_type() returns the interlace method used for the image.

Return Value

Valid values are PNG_INTERLACE_NONE, PNG_INTERLACE_ADAM7.
Returns 0 if png_ptr or info_ptr is NULL.

png_get_io_ptr**Name**

png_get_io_ptr — return pointer for user-defined I/O

Synopsis

```
#include <png.h>
png_voidp png_get_io_ptr(png_structp png_ptr);
```

Description

Returns the pointer associated with user-defined input-output functions.

png_get_libpng_ver**Name**

png_get_libpng_ver — get the library version string

Synopsis

```
#include <png.h>
png_charp png_get_libpng_ver(png_structp png_ptr);
```

Description

png_get_libpng_ver() shall return the library version as a short string in the format "1.0.0" through "99.99.99zz".

png_get_oFFs**Name**

png_get_oFFs — get screen offsets for the given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_oFFs(png_structp png_ptr, png_infop info_ptr,
png_int_32 * x_offset, png_int_32 * y_offset, int * unit_type);
```

Description

png_get_oFFs() shall read the positive offset from the left edge of the screen from info_ptr to x_offset and the positive offset from the top edge of the screen to y_offset. The unit type shall be returned in unit_type, which can take the following values

```
PNG_OFFSET_PIXEL
PNG_OFFSET_MICROMETER
```

Return Value

On success, png_get_oFFs() shall return PNG_INFO_oFFs. Otherwise, png_get_oFFs() shall return 0.

png_get_pHYs

Name

png_get_pHYs — get the physical resolution for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_pHYs(png_structp png_ptr, png_infop info_ptr,
png_uint_32 * res_x, png_uint_32 * res_y, int * unit_type);
```

Description

png_get_pHYs() shall return the physical pixel resolution of the image and the unit of resolution. Upon success, *res_x shall contain the horizontal resolution and *res_y shall contain the vertical resolution in pixels per unit. *unit_type will be set to PNG_RESOLUTION_METER if the resolution is expressed in pixels per meter. Otherwise *unit_type will be PNG_RESOLUTION_UNKNOWN.

Return Value

PNG_INFO_pHYs - on success 0 - otherwise.

png_get_progressive_ptr

Name

png_get_progressive_ptr — return pointer to user-defined push read functions

Synopsis

```
#include <png.h>
png_voidp png_get_progressive_ptr(png_structp png_ptr);
```

Description

Returns the pointer to user-defined structure containing information about the callback functions.

png_get_rowbytes

Name

png_get_rowbytes — Return number of bytes for a row

Synopsis

```
#include <png.h>
png_uint_32 png_get_rowbytes(png_structp png_ptr, png_infop info_ptr);
```

Description

png_get_rowbytes() returns the number of bytes needed to hold a transformed row of an image.

Return Value

Returns 0 if png_ptr or info_ptr is NULL, number of bytes otherwise.

png_get_rows**Name**

`png_get_rows` — retrieve image data from `png_info` structure

Synopsis

```
#include <png.h>
png_bytepp png_get_rows(png_structp png_ptr, png_infop info_ptr);
```

Description

`png_get_rows()` shall retrieve rows of image data from the `info_ptr` structure in an array of pointers to the pixel data for each row.

Return Value

On success, `png_get_rows()` shall return an array of pointers to the pixel data for each row of the image.

png_get_sBIT**Name**

`png_get_sBIT` — get number of significant bits for each color channel

Synopsis

```
#include <png.h>
png_uint_32 png_get_sBIT(png_structp png_ptr, png_infop info_ptr,
png_color_8p * sig_bit);
```

Description

`png_get_sBIT()` shall return the number of significant bit for each of the gray, red, blue and green color channels.

Return Value

On success, `png_get_sBIT()` shall return `PNG_INFO_sBIT`. Otherwise, `png_get_sBIT()` shall return 0.

png_get_sRGB**Name**

png_get_sRGB — get the rendering intent for given image

Synopsis

```
#include <png.h>
png_uint_32 png_get_sRGB(png_structp png_ptr, png_infop info_ptr, int
* srgb_intent);
```

Description

png_get_sRGB() shall return the rendering intent of an image to *srgb_intent if the sRGB chunk information is valid for the image.

Return Value

PNG_INFO_sRGB - if png_ptr, info_ptr, srgb_intent are not NULL and sRGB chunk information is valid 0 - otherwise.

png_get_tIME**Name**

png_get_tIME — get last modification time for the image

Synopsis

```
#include <png.h>
png_uint_32 png_get_tIME(png_structp png_ptr, png_infop info_ptr,
png_timep * mod_time);
```

Description

png_get_tIME() shall return the time of last modification of the image if the tIME information is valid for the image.

Return Value

PNG_INFO_tIME - if png_ptr, info_ptr, mod_time are not NULL and tIME information is valid 0 - otherwise.

png_get_tRNS

Name

png_get_tRNS — get transparency data for images

Synopsis

```
#include <png.h>
png_uint_32 png_get_tRNS(png_structp png_ptr, png_info_ptr info_ptr,
png_bytep * trans, int * num_trans, png_color_16p * trans_values);
```

Description

png_get_tRNS() shall obtain the transparency data for paletted images and image types that don't need a full alpha channel from info_ptr. For a paletted image, the function retrieves the transparency values stored in the same order as the palette colors, starting from index 0. For non-paletted images, the function retrieves the single color value which is treated as fully transparent. If the transparency information is valid, i.e. PNG_INFO_tRNS bit is set for info_ptr->valid: *trans shall be set to the transparency values for a paletted image. Values for the data shall be in range [0,255], ranging from fully transparent to fully opaque, respectively. *num_trans shall be set to the number of transparency values. *trans_values shall be set to the single color value specified for non-paletted images.

Return Value

Returns PNG_INFO_tRNS on successful return, 0 otherwise.

png_get_text

Name

png_get_text — get comments information from png_info structure

Synopsis

```
#include <png.h>
png_uint_32 png_get_text(png_structp png_ptr, png_info_ptr info_ptr,
png_textp * text_ptr, int * num_text);
```

Description

png_get_text() returns the text chunk information from the PNG stream in the array pointed to by text_ptr. It also returns the number of text chunks in num_text. text_ptr is an array of structure png_text whose members include:

compression

type of compression used on text. Valid values are:

PNG_TEXT_COMPRESSION_NONE
 PNG_TEXT_COMPRESSION_zTXt
 PNG_ITXT_COMPRESSION_NONE
 PNG_ITXT_COMPRESSION_zTXt

key

keyword for comment. Must contain 1-79 characters.

text

text comment for current keyword. May be empty.

text_length

length of text string after decompression. 0 for iTXt.

Return Value

Returns 0 if *png_ptr* or *info_ptr* is NULL, returns the number of text chunks otherwise.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

png_get_unknown_chunks

Name

`png_get_unknown_chunks` — retrieve the unknown chunks from a PNG file

Synopsis

```
#include <png.h>
png_uint_32 png_get_unknown_chunks(png_structp png_ptr, png_infop
info_ptr, png_unknown_chunkpp unknowns);
```

Description

This interface shall retrieve the unknown chunks from a PNG file.

The parameter `png_ptr` shall specify the PNG file.

The parameter `info_ptr` shall specify the PNG info structure.

The parameter `unknowns` shall specify an array of `png_unknown_chunk` structures containing the unknown chunks. The position of a structure in the array shall correspond to the order in which `png_set_unknown_chunks()` inserted its chunk into the PNG file, or in which it was read.

The `png_unknown_chunkpp` structure shall contain the following members.

name

the name of the chunk

data

the data of the chunk

size

the size of the chunk's data

location

the position of the chunk in the PNG file

Return Value

On success, this interface shall return the number of unknown chunks obtained.

On failure, this interface shall return 0.

png_get_user_chunk_ptr

Name

png_get_user_chunk_ptr — get pointer to user chunk data

Synopsis

```
#include <png.h>
png_voidp png_get_user_chunk_ptr(png_structp png_ptr);
```

Description

This interface shall return the pointer to the user chunk data associated with the specified PNG file.

The parameter *png_ptr* shall specify the PNG file.

png_get_valid

Name

png_get_valid — determine if given chunk data is valid

Synopsis

```
#include <png.h>
png_uint_32 png_get_valid(png_structp png_ptr, png_infop info_ptr,
png_uint_32 flag);
```

Description

png_get_valid() shall obtain the validity of chunk data specified by the bits set in "flag". The following bits may be set in flag: PNG_INFO_gAMA PNG_INFO_sBIT PNG_INFO_cHRM PNG_INFO_PLTE PNG_INFO_tRNS PNG_INFO_bKGD PNG_INFO_hIST PNG_INFO_pHYs PNG_INFO_oFFs PNG_INFO_tIME PNG_INFO_pCAL PNG_INFO_sRGB PNG_INFO_iCCP PNG_INFO_sPLT PNG_INFO_sCAL PNG_INFO_IDAT

Return Value

On success, png_get_valid() shall return "flag" with the chunk bits set. Otherwise, png_get_valid() shall return 0.

png_get_x_offset_pixels**Name**

`png_get_x_offset_pixels` — return x offset in pixels from oFFs chunk

Synopsis

```
#include <png.h>
png_int_32 png_get_x_offset_pixels(png_structp png_ptr, png_infop
info_ptr);
```

Description

`png_get_x_offset_pixels()` shall obtain x offset in pixels for the image from its PNG_oFFs chunk data stored in `info_ptr`, if the unit for offset is pixels.

Return Value

On success, `png_get_x_offset_pixels()` shall return x offset in pixels. Otherwise `png_get_x_offset_pixels()` shall return 0.

png_get_x_pixels_per_meter**Name**

`png_get_x_pixels_per_meter` — return horizontal pixel density per meter

Synopsis

```
#include <png.h>
png_uint_32 png_get_x_pixels_per_meter(png_structp png_ptr, png_infop
info_ptr);
```

Description

`png_get_x_pixels_per_meter()` shall obtain the horizontal pixel density in pixels per meter from its PNG_pHYs chunk data stored in `info_ptr`, if the unit for resolution is pixels per meter.

Return Value

On success, `png_get_x_pixels_per_meter()` shall return horizontal pixel density in pixels per meter. Otherwise `png_get_x_pixels_per_meter()` shall return 0.

png_get_y_offset_pixels**Name**

`png_get_y_offset_pixels` — return y offset in pixels from oFFs chunk

Synopsis

```
#include <png.h>
png_int_32 png_get_y_offset_pixels(png_structp png_ptr, png_infop
info_ptr);
```

Description

`png_get_y_offset_pixels()` shall obtain y offset in pixels for the image from its PNG_oFFs chunk data stored in `info_ptr`, if the unit for offset is pixels.

Return Value

On success, `png_get_y_offset_pixels()` shall return y offset in pixels. Otherwise `png_get_y_offset_pixels()` shall return 0.

png_get_y_pixels_per_meter**Name**

`png_get_y_pixels_per_meter` — return vertical pixel density per meter

Synopsis

```
#include <png.h>
png_uint_32 png_get_y_pixels_per_meter(png_structp png_ptr, png_infop
info_ptr);
```

Description

`png_get_y_pixels_per_meter()` shall obtain the vertical pixel density in pixels per meter from its PNG_pHYs chunk data stored in `info_ptr`, if the unit for resolution is pixels per meter.

Return Value

On success, `png_get_y_pixels_per_meter()` shall return vertical pixel density in pixels per meter. Otherwise `png_get_y_pixels_per_meter()` shall return 0.

png_info_init_3

Name

png_info_init_3 — initialize an info structure (DEPRECATED)

Synopsis

```
#include <png.h>
void      png_info_init_3(png_infopp      info_ptr,      png_size_t
png_info_struct_size);
```

Description

This interface shall initialize a PNG info structure. This interface is deprecated.

The parameter *info_ptr* shall specify the PNG info structure to initialize.

The parameter *png_info_struct_size* shall specify the size of the new structure.

png_init_io

Name

png_init_io — initialize input/output for the PNG file

Synopsis

```
#include <png.h>
void png_init_io(png_structp png_ptr, png_FILE_p fp);
```

Description

Initialize the default input/output functions for the PNG file to standard C streams. To replace the default read and write functions, use *png_set_read_fn()* and *png_set_write_fn()* respectively.

Return Value

Errors

png_libpng_ver

Name

png_libpng_ver — external variable holding version number of libpng

Synopsis

```
#include <png.h>
const char png_libpng_ver[18];
```

Description

The external variable *png_libpng_ver* shall contain a character string value indicating the version number of libpng. An LSB conforming implementation shall return a value at least as high as *PNG_LIBPNG_VER_STRING* as defined in this specification.

png_malloc

Name

png_malloc — allocate memory

Synopsis

```
#include <png.h>
png_voidp png_malloc(png_structp png_ptr, png_uint_32 size);
```

Description

png_malloc() shall return a pointer to allocated memory of the specified size.

Return Value

Pointer to the block of memory allocated.

Errors

Invokes error handling function if the system is out of memory and sets PNG_FLAG_MALLOC_NULL_MEM_OK in member flags of png_struct.

png_permit_mng_features

Name

png_permit_mng_features — enable MNG extensions for PNG image wrapped in MNG datastream

Synopsis

```
#include <png.h>
png_uint_32 png_permit_mng_features(png_structp png_ptr, png_uint_32
mng_features_permitted);
```

Description

This interface shall enable some MNG extensions for a PNG image wrapped in a MNG datastream.

The parameter *png_ptr* shall specify the PNG image.

The parameter *mng_features_permitted* shall specify the logical OR of the features to be enabled, possibly including PNG_ALL_MNG_FEATURES, PNG_FLAG_MNG_EMPTY_PLTE, and PNG_FLAG_MNG_FILTER_64.

Return Value

This interface shall return the logical AND of the parameter *mng_features_permitted* with the set of MNG features supported by the local version of libpng.

Application Usage (informational)

This interface may not read or write a standalone PNG file; the PNG datastream must be embedded in a MNG datastream with an 8-byte MNG signature and MEND and MHDR chunks.

png_process_data**Name**

png_process_data — read PNG file progressively

Synopsis

```
#include <png.h>
void png_process_data(png_structp png_ptr, png_infop info_ptr,
png_bytep buffer, png_size_t length);
```

Description

png_process_data() shall process data progressively from the PNG stream using callback functions set within png_set_progressive_read_fn(). The data is passed in "buffer" and length of data to be processed is specified by "length".

png_progressive_combine_row**Name**

png_progressive_combine_row — combines current row data with processed row

Synopsis

```
#include <png.h>
void png_progressive_combine_row(png_structp png_ptr, png_bytep old_row,
png_bytep new_row);
```

Description

For non-NULL rows of interlaced images during progressive read, png_progressive_combine_row() shall combine the data for the current row with the previously processed row data. png_progressive_combine_row() shall return for NULL rows of interlaced images and memcpy rows for non-interlaced images.

png_read_end**Name**

png_read_end — read the end of PNG file

Synopsis

```
#include <png.h>
void png_read_end(png_structp png_ptr, png_infop info_ptr);
```

Description

png_read_end() reads the end of a PNG file after reading the image data, including any comments or time information at the end of the file. The function shall not read past the end of the file.

png_read_image

Name

png_read_image — read the entire image into memory

Synopsis

```
#include <png.h>
void png_read_image(png_structp png_ptr, png_bytepp image);
```

Description

png_read_image() reads the entire image into memory at once. For each pass of an interlaced image, use png_read_rows() instead.

png_read_info

Name

png_read_info — read the PNG image information

Synopsis

```
#include <png.h>
void png_read_info(png_structp png_ptr, png_infop info_ptr);
```

Description

Reads the information before the actual image data from the PNG file. The function allows reading a file that already has the PNG signature bytes read from the stream.

png_read_png

Name

png_read_png — read the entire PNG file

Synopsis

```
#include <png.h>
void png_read_png(png_structp png_ptr, png_infop info_ptr, int
transforms, png_voidp params);
```

Description

png_read_png() shall provide the high-level read operation. The function shall read the entire image into memory. The integer "transforms" shall contain the logical OR of a set of the following transformation flags:

PNG_TRANSFORM_IDENTITY

No transformation

PNG_TRANSFORM_STRIP_16

Strip 16-bit samples to 8 bits

PNG_TRANSFORM_STRIP_ALPHA

Discard the alpha channel

PNG_TRANSFORM_PACKING

Expand 1, 2 and 4-bit samples to bytes

PNG_TRANSFORM_PACKSWAP

Change order of packed pixels to LSB first

PNG_TRANSFORM_EXPAND

Expand paletted images to RGB, grayscale to 8-bit images and tRNS chunks to alpha channels

PNG_TRANSFORM_INVERT_MONO

Invert monochrome images

PNG_TRANSFORM_SHIFT

Normalize pixels to the sBIT depth

PNG_TRANSFORM_BGR

Flip RGB to BGR, RGBA to BGRA

PNG_TRANSFORM_SWAP_ALPHA

Flip RGBA to ARGB or GA to AG

PNG_TRANSFORM_INVERT_ALPHA

Change alpha from opacity to transparency

PNG_TRANSFORM_SWAP_ENDIAN

Byte-swap 16-bit samples

"params" is unused and must be set to NULL.

png_read_row

Name

png_read_row — read a row of image data

Synopsis

```
#include <png.h>
void png_read_row(png_structp png_ptr, png_bytep row, png_bytep
display_row);
```

Description

png_read_row() reads a row of actual image data. "row" holds the image pixels as they are processed. If the image is displayed after each pass, "display_row" is used to display a blurred progressive image. "display_row" can be NULL if the progressive image is not required.

png_read_rows

Name

png_read_rows — read multiple rows of image data

Synopsis

```
#include <png.h>
void png_read_rows(png_structp png_ptr, png_bytepp row, png_bytepp
display_row, png_uint_32 num_rows);
```

Description

Read "num_rows" rows of image data starting from "row". If the image is interlaced, the rows must contain the contents of the rows from the previous pass. If the image is displayed after each pass, "display_row" is used to display a blurred progressive image. "display_row" can be NULL if the progressive image is not required.

png_read_update_info

Name

png_read_update_info — update png_info structure

Synopsis

```
#include <png.h>
void png_read_update_info(png_structp png_ptr, png_infop info_ptr);
```

Description

png_read_update_info() updates the structure pointed to by info_ptr to reflect any transformations that have been requested. For example, rowbytes will be updated to handle expansion of an interlaced image with png_read_update_info().

png_set_IHDR

Name

png_set_IHDR — set the PNG_IHDR chunk information

Synopsis

```
#include <png.h>
void png_set_IHDR(png_structp png_ptr, png_infop info_ptr, png_uint_32
width, png_uint_32 height, int bit_depth, int color_type, int
interlace_type, int compression_type, int filter_type);
```

Description

png_set_IHDR() shall set image header information in info_ptr. width is the image width in pixels. height is the image height in pixels. bit_depth is the bit depth of the image. Valid values shall include 1, 2, 4, 8, 16 and shall also depend on the color type. color_type is the type of image. Supported color types shall include: PNG_COLOR_TYPE_GRAY (bit depths 1, 2, 4, 8, 16) PNG_COLOR_TYPE_GRAY_ALPHA (bit depths 8, 16) PNG_COLOR_TYPE_PALETTE (bit depths 1, 2, 4, 8) PNG_COLOR_TYPE_RGB (bit depths 8, 16) PNG_COLOR_TYPE_RGB_ALPHA (bit depths 8, 16) PNG_COLOR_MASK_PALETTE PNG_COLOR_MASK_COLOR PNG_COLOR_MASK_ALPHA interlace_type is the image interlace method. Supported values shall include: PNG_INTERLACE_NONE or PNG_INTERLACE_ADAM7 compression_type is the method used for image compression. The value must be PNG_COMPRESSION_TYPE_DEFAULT. filter_type is the method used for image filtering. The value must be PNG_FILTER_TYPE_DEFAULT.

Errors

png_set_IHDR() shall invoke error function if any of the arguments has an invalid value.

png_set_PLTE

Name

png_set_PLTE — set color values for the palette

Synopsis

```
#include <png.h>
void png_set_PLTE(png_structp png_ptr, png_infop info_ptr, png_colorp
palette, int num_palette);
```

Description

png_set_PLTE() shall set the array of color values used as palette for image to "palette". The palette shall include "num_palette" entries.

png_set_bKGD

Name

png_set_bKGD — set the background color for given image

Synopsis

```
#include <png.h>
void png_set_bKGD(png_structp png_ptr, png_infop info_ptr,
png_color_16p background);
```

Description

png_set_bKGD() shall set the background color of an image to "background" and sets bKGD chunk information to valid for the image.

png_set_background

Name

png_set_background — set the background for given image

Synopsis

```
#include <png.h>
void png_set_background(png_structp png_ptr, png_color_16p
background_color, int background_gamma_code, int need_expand, double
background_gamma);
```

Description

png_set_background() shall set the background of an image with alpha channel or simple transparency with the specified background color. If background_gamma_code is set to PNG_BACKGROUND_GAMMA_SCREEN, it indicates that the supplied background color is in the gamma space of the display, else if it is set to PNG_BACKGROUND_GAMMA_FILE, the color is in the gamma space of the file. If the background color is supplied at the original bit-depth for a grayscale image that is expanded to truecolor or to a higher bit-depth, need_expand must be set to 1, but if the background color is supplied at the expanded bit-depth, need_expand must be set to 0. Similarly for paletted images, if background color is supplied as a palette index, need_expand must be set to 1, else if background color is supplied as an RGB triplet, need_expand must be set to 0.

png_set_bgr

Name

png_set_bgr — set pixel order to blue, green, red

Synopsis

```
#include <png.h>
void png_set_bgr(png_structp png_ptr);
```

Description

png_set_bgr() shall set the pixel order to blue, green, red.

png_set_cHRM**Name**

`png_set_cHRM` — set CIE chromacities and referenced white point for given image

Synopsis

```
#include <png.h>
void png_set_cHRM(png_structp png_ptr, png_infop info_ptr, double
white_x, double white_y, double red_x, double red_y, double green_x,
double green_y, double blue_x, double blue_y);
```

Description

`png_set_cHRM()` shall set the CIE x,y chromaticities of the red, green and blue display primaries for the image and the referenced white point. The values must range from 0 to 21474.83 both inclusive.

Errors

`png_set_cHRM()` shall report a non-fatal error and exit if any of the chromacity values lies outside the range 0 to 21474.83.

png_set_compression_buffer_size**Name**

`png_set_compression_buffer_size` — set the size of the compression buffer

Synopsis

```
#include <png.h>
void png_set_compression_buffer_size(png_structp png_ptr, png_uint_32
size);
```

Description

This interface shall set the size of the libz compression buffer `zbuf` for the specified PNG file.

The parameter `png_ptr` shall specify the PNG file for which to set the size of the compression buffer.

The parameter `size` shall specify the size to which to set the compression buffer, in bytes.

png_set_compression_level

Name

png_set_compression_level — set image compression level

Synopsis

```
#include <png.h>
void png_set_compression_level(png_structp png_ptr, int level);
```

Description

png_set_compression_level() shall set the compression level to "level". The valid values for "level" range from [0,9], corresponding directly to compression levels for zlib. The value 0 implies no compression and 9 implies maximal compression. Note: Tests have shown that zlib compression levels 3-6 usually perform as well as level 9 for PNG images, and do considerably fewer calculations.

png_set_compression_mem_level

Name

png_set_compression_mem_level — set how much memory to use for the internal state during PNG compression

Synopsis

```
#include <png.h>
void png_set_compression_mem_level(png_structp png_ptr, int mem_level);
```

Description

This interface shall set how much memory to use for the internal state during PNG compression.

The parameter *png_ptr* shall specify the PNG file to compress.

The parameter *mem_level* corresponds directly to the *memLevel* parameter of the `libz deflateInit2_()` interface. This parameter shall specify how much memory to use for the internal state. The value of *mem_level* must be between 1 and `MAX_MEM_LEVEL`. Smaller values use less memory but are slower, while higher values use more memory to gain compression speed.

png_set_compression_method**Name**

`png_set_compression_method` — set PNG compression algorithm

Synopsis

```
#include <png.h>
void png_set_compression_method(png_structp png_ptr, int method);
```

Description

This interface shall set the PNG compression algorithm to use.

The parameter *png_ptr* shall specify the PNG file to compress.

The parameter *method* corresponds directly to the *method* parameter of the `libz deflateInit2_()` interface. An LSB-conforming implementation shall support the `Z_DEFLATED` method, and may support other implementation defined methods.

png_set_compression_strategy**Name**

`png_set_compression_strategy` — set PNG compression strategy

Synopsis

```
#include <png.h>
void png_set_compression_strategy(png_structp png_ptr, int strategy);
```

Description

This interface shall set the PNG compression strategy.

The parameter *png_ptr* shall specify the PNG file to compress.

The parameter *strategy* corresponds directly to the *strategy* parameter of the `libz deflateInit2_()` interface. This parameter shall specify the PNG compression strategy to use: one of `Z_DEFAULT_STRATEGY`, `Z_FILTERED`, and `Z_HUFFMAN_ONLY`.

png_set_compression_window_bits

Name

`png_set_compression_window_bits` — set PNG compression window size

Synopsis

```
#include <png.h>
void png_set_compression_window_bits(png_structp png_ptr, int
window_bits);
```

Description

This interface shall set the PNG compression window size.

The parameter *window_bits* corresponds directly to the *windowBits* parameter of the libz `deflateInit2_()` interface. The value of this parameter equals the base 2 logarithm of the window size to use, and must be a value between 8 and 15.

png_set_dither

Name

`png_set_dither` — turn on dithering to 8-bit

Synopsis

```
#include <png.h>
void png_set_dither(png_structp png_ptr, png_colorp palette, int
num_palette, int maximum_colors, png_uint_16p histogram, int
full_dither);
```

Description

`png_set_dither()` shall set transformation to dither file to 8-bit. For the given palette with *num_palette* number of colors, the number of colors in the palette shall be reduced to fit in "*maximum_colors*" if the palette is larger than *maximum_colors*. If supplied, the histogram is used to get better results of the reduced palette. For reducing the number of colors in palette, *full_dither* must be set to 0. If *full_dither* is set to 1, dithering cube shall be setup for RGB images to reduce the RGB file to a paletted file.

png_set_error_fn**Name**

`png_set_error_fn` — set user defined functions for error handling

Synopsis

```
#include <png.h>
void png_set_error_fn(png_structp png_ptr, png_voidp error_ptr,
png_error_ptr error_fn, png_error_ptr warning_fn);
```

Description

`png_set_error_fn()` shall replace the default error handling and warning functions with user defined function `error_fn` for handling fatal errors and function `warning_fn` for handling non-fatal errors. The replacement functions must do a `longjmp` to the last `setjmp` location if `setjmp/longjmp` method of error handling is used. If `error_fn` or `warning_fn` is `NULL`, the default functions for error handling shall be used.

png_set_expand**Name**

`png_set_expand` — set expansion transformation

Synopsis

```
#include <png.h>
void png_set_expand(png_structp png_ptr);
```

Description

`png_set_expand()` shall set transformation in `png_ptr` such that paletted images are expanded to RGB, grayscale images of bit-depth less than 8 are expanded to 8-bit images and tRNS chunks are expanded to alpha channels.

png_set_filler**Name**

`png_set_filler` — add a filler byte to given image

Synopsis

```
#include <png.h>
void png_set_filler(png_structp png_ptr, png_uint_32 filler, int
flags);
```

Description

`png_set_filler()` shall set transformations in `png_ptr` such that a filler byte is added when an 8-bit grayscale image or 24-bit RGB image is read and a filler byte is deleted when an 8-bit grayscale image or 24-bit RGB image is written.

png_set_filter

Name

png_set_filter — set filtering method

Synopsis

```
#include <png.h>
void png_set_filter(png_structp png_ptr, int method, int filters);
```

Description

png_set_filter() shall set the filtering method used for scan-line filtering. The only valid value for "method" is 0. "filters" is a bitmap for which the following bits may be set. PNG_NO_FILTERS PNG_FILTER_NONE PNG_FILTER_SUB PNG_FILTER_UP PNG_FILTER_AVG PNG_FILTER_PAETH PNG_ALL_FILTERS

png_set_gAMA

Name

png_set_gAMA — set the gamma value for given image

Synopsis

```
#include <png.h>
void png_set_gAMA(png_structp png_ptr, png_info_ptr info_ptr, double
file_gamma);
```

Description

Sets the gamma value of an image to "file_gamma" and sets gAMA chunk information to valid for the image.

Errors

png_set_gAMA() shall generate warning if file_gamma > 21474.83 or file_gamma = 0

png_set_gamma

Name

png_set_gamma — transform the image from file gamma to screen gamma

Synopsis

```
#include <png.h>
void png_set_gamma(png_structp png_ptr, double screen_gamma, double
file_gamma);
```

Description

png_set_gamma() shall set the transformation for gamma correction of the PNG file based on the screen gamma i.e. the display exponent. The gamma transformation may be turned off later if no semitransparent entries are present in the tRNS array for palette images.

png_set_gray_1_2_4_to_8**Name**

png_set_gray_1_2_4_to_8 — set expansion transformation

Synopsis

```
#include <png.h>
void png_set_gray_1_2_4_to_8(png_structp png_ptr);
```

Description

png_set_gray_1_2_4_to_8() shall set transformation in *png_ptr* such that grayscale images of bit-depth less than 8 are expanded to 8-bit images. png_set_png_set_gray_1_2_4_to_8() is actually an alias for png_set_expand().

png_set_gray_to_rgb**Name**

png_set_gray_to_rgb — expand the grayscale image to 24-bit RGB

Synopsis

```
#include <png.h>
void png_set_gray_to_rgb(png_structp png_ptr);
```

Description

png_set_gray_to_rgb() shall set transformations such that the grayscale image is converted to 24-bit RGB.

png_set_hIST**Name**

png_set_hIST — set the histogram of color palette

Synopsis

```
#include <png.h>
void png_set_hIST(png_structp png_ptr, png_infop info_ptr,
png_uint_16p hist);
```

Description

png_set_hIST() shall set the histogram of palette to "hist".

png_set_iCCP

Name

png_set_iCCP — set ICC component

Synopsis

```
#include <png.h>
void png_set_iCCP(png_structp png_ptr, png_infop info_ptr, png_charp
name, int compression_type, png_charp profile, png_uint_32 proflen);
```

Description

png_set_iCCP() shall set the ICC component information to info_ptr. The arguments used to describe the ICC profile information have been described below:

name

ICC profile name

compression_type

compression type used must be 0

profile

profile data

proflen

length of profile data

png_set_interlace_handling

Name

png_set_interlace_handling — get the number of passes for image interlacing

Synopsis

```
#include <png.h>
int png_set_interlace_handling(png_structp png_ptr);
```

Description

png_set_interlace_handling() shall set the scheme to interlacing for writing an image and return the number of sub-images required to write the image.

Return Value

png_set_interlace_handling() shall return 7 if the image is interlaced, otherwise png_set_interlace_handling() shall return 1.

png_set_invert_alpha

Name

png_set_invert_alpha — invert the level of opacity of a PNG file

Synopsis

```
#include <png.h>
void png_set_invert_alpha(png_structp png_ptr);
```

Description

This interface shall invert the level of opacity (alpha) of a PNG file.

The parameter *png_ptr* shall specify the PNG file for which to invert the opacity.

png_set_invert_mono

Name

png_set_invert_mono — reverse values for monochromaticity

Synopsis

```
#include <png.h>
void png_set_invert_mono(png_structp png_ptr);
```

Description

png_set_invert_mono() shall set monochromaticity value 0 to white and value 1 to black.

png_set_keep_unknown_chunks

Name

`png_set_keep_unknown_chunks` — specify list of chunks and how to handle them

Synopsis

```
#include <png.h>
void png_set_keep_unknown_chunks(png_structp png_ptr, int keep,
png_bytep chunk_list, int num_chunks);
```

Description

This interface shall specify a list of chunks in the input PNG stream and how to handle them. Any unspecified chunks shall be handled in the default way. The IEND and IHDR chunks must not be specified.

The parameter `png_ptr` shall specify the PNG file.

The parameter `keep` shall specify how the unknown chunks are to be handled (see below).

The parameter `chunk_list` shall specify the list of chunks that shall be affected. The value passed must be a string of bytes with five bytes per chunk, or NULL or \0 if the value of `num_chunks` is 0.

The parameter `num_chunks` shall specify the number of chunks to be affected. If the value is 0, all unknown chunks shall be affected.

The possible values of `keep` are as follows.

- 0
handle unknown chunks in the default way
- 1
do not keep unknown chunks
- 2
keep unknown chunks only if they are safe to copy
- 3
keep unknown chunks even if they are unsafe to copy

Application Usage (informative)

The normal behavior of libpng is that known chunks are processed and unknown chunks are discarded. This interface reads both known and unknown chunks, handling them as specified by the user.

Unknown chunks specified to this interface are saved unchanged in a list of `png_unknown_chunk` structures. If a known chunk is specified in the list of unknown chunks, it will be handled per the `keep` parameter. If a chunk is specified in successive calls to this interface, the final call takes precedence.

png_set_mem_fn

Name

png_set_mem_fn — install custom memory allocation functions

Synopsis

```
#include <png.h>
void png_set_mem_fn(png_structp png_ptr, png_voidp mem_ptr,
png_malloc_ptr malloc_fn, png_free_ptr free_fn);
```

Description

This interface shall install custom memory allocation functions.

The parameter *png_ptr* shall specify the PNG file.

The parameter *mem_ptr* shall specify a struct provided by the user for memory functions.

The parameter *malloc_fn* shall specify the function used to allocate memory. If the value of this parameter is `NULL`, then the capability to allocate memory with the libpng ABI shall be disabled.

The parameter *free_fn* shall specify the function used to free memory. If the value of this parameter is `NULL`, then the capability to free memory with the libpng ABI shall be disabled.

png_set_oFFs

Name

png_set_oFFs — set screen offsets for given image

Synopsis

```
#include <png.h>
void png_set_oFFs(png_structp png_ptr, png_infop info_ptr, png_int_32
offset_x, png_int_32 offset_y, int unit_type);
```

Description

`png_set_oFFs()` shall set the positive offset from the left edge of the screen to `offset_x` and the positive offset from the left edge of the screen to `offset_y`. The `unit_type` must be `PNG_OFFSET_PIXEL` if the offset is defined in pixels or `PNG_OFFSET_MICROMETER` if the offset is defined in microns.

png_set_pHYs**Name**

png_set_pHYs — set physical resolution

Synopsis

```
#include <png.h>
void png_set_pHYs(png_structp png_ptr, png_infop info_ptr, png_uint_32
res_x, png_uint_32 res_y, int unit_type);
```

Description

png_set_pHYs() sets the physical resolution for the image in pixels per unit. The physical resolution in x direction is set to res_x and that in y direction is set to res_y. unit_type must be set to PNG_RESOLUTION_METER is the unit for resolution is pixels per unit, otherwise unit_type must be set to PNG_RESOLUTION_UNKNOWN.

png_set_packing**Name**

png_set_packing — expand image to 1 pixel per byte for bit-depths 1,2 and 4

Synopsis

```
#include <png.h>
void png_set_packing(png_structp png_ptr);
```

Description

png_set_packing() shall expand image to 1 pixel per byte for bit-depths 1, 2 and 4 without changing the order of the pixels. If png_set_packing() is not called, PNG files pack pixels of bit depths 1, 2 and 4 into bytes as small as possible, for example, 8 pixels per byte for 1-bit files.

png_set_packswap**Name**

png_set_packswap — swap the order of pixels for packed-pixel image

Synopsis

```
#include <png.h>
void png_set_packswap(png_structp png_ptr);
```

Description

png_set_swap() shall change the pixel packing order for each byte for packed-pixel images with bit-depths 1, 2 or 4.

png_set_palette_to_rgb**Name**

png_set_palette_to_rgb — set expansion transformation

Synopsis

```
#include <png.h>
void png_set_palette_to_rgb(png_structp png_ptr);
```

Description

png_set_palette_to_rgb() shall set transformation in *png_ptr* such that paletted images are expanded to RGB. png_set_palette_to_rgb() is actually an alias for png_set_expand().

png_set_progressive_read_fn**Name**

png_set_progressive_read_fn — set progressive read callback functions

Synopsis

```
#include <png.h>
void png_set_progressive_read_fn(png_structp png_ptr, png_voidp
user_ptr, png_progressive_info_ptr info_callback,
png_progressive_row_ptr row_callback, png_progressive_end_ptr
end_callback);
```

Description

png_set_progressive_read_fn() shall provide function callbacks for which shall be called for processing image data by png_process_data(). "info_callback" shall be called to process header information, "row_callback" shall be called when each row is completed and "end_callback" shall be called to process end of image information. png_set_progressive_read_fn() must be called even if all callback functions are NULL. The user-defined structure pointed to by "user_ptr" may be retrieved from inside the callbacks using function get_progressive_ptr().

png_set_read_fn**Name**

`png_set_read_fn` — set user-defined function for reading a PNG stream

Synopsis

```
#include <png.h>
void png_set_read_fn(png_structp png_ptr, png_voidp io_ptr, png_rw_ptr
read_data_fn);
```

Description

`png_set_read_fn()` sets the `read_data_fn` as the input function for reading PNG files instead of using standard C I/O stream functions. `png_ptr` - pointer to input data structure `png_struct` `io_ptr` - pointer to user-defined structure containing information about the input functions. This value may be NULL. `read_data_fn` - pointer to new input function that shall take the following arguments: - a pointer to a `png_struct` - a pointer to a structure where input data can be stored - 32-bit unsigned int to indicate number of bytes to read The input function should invoke `png_error()` to handle any fatal errors and `png_warning()` to handle non-fatal errors.

png_set_read_user_chunk_fn**Name**

`png_set_read_user_chunk_fn` — install custom callback function to handle unknown chunks in the input stream

Synopsis

```
#include <png.h>
void png_set_read_user_chunk_fn(png_structp png_ptr, png_voidp
user_chunk_ptr, png_user_chunk_ptr read_user_chunk_fn);
```

Description

This interface shall install a custom callback function to handle unknown chunks in the input stream.

The parameter `png_ptr` shall specify the PNG file.

The parameter `user_chunk_ptr` shall specify a user pointer obtainable with `png_get_user_chunk_ptr()`.

The parameter `read_user_chunk_fn` shall specify the custom callback function.

png_set_read_user_transform_fn**Name**

`png_set_read_user_transform_fn` — install a custom input transformation callback function

Synopsis

```
#include <png.h>
void png_set_read_user_transform_fn(png_structp png_ptr,
png_user_transform_ptr read_user_transform_fn);
```

Description

This interface shall install a custom input transformation callback function.

The parameter `png_ptr` shall specify the PNG file to be transformed.

The parameter `read_user_transform_fn` shall specify the custom callback function.

png_set_rgb_to_gray**Name**

`png_set_rgb_to_gray` — reduce 24-bit RGB to grayscale image

Synopsis

```
#include <png.h>
void png_set_rgb_to_gray(png_structp png_ptr);
```

Description

`png_set_rgb_to_gray()` shall set transformations such that the 24-bit RGB image is converted to grayscale.

png_set_rows**Name**

`png_set_rows` — put image data in `png_info` structure

Synopsis

```
#include <png.h>
void png_set_rows(png_structp png_ptr, png_infop info_ptr, png_bytepp
row_pointers);
```

Description

`png_set_rows()` shall put rows of image data into the `info_ptr` structure, where `row_pointers` is an array of pointers to the pixel data for each row.

png_set_sBIT

Name

png_set_sBIT — set number of significant bits for each channel

Synopsis

```
#include <png.h>
void png_set_sBIT(png_structp png_ptr, png_infop info_ptr,
png_color_8p sig_bit);
```

Description

png_set_sBIT shall set the number of significant bits for each of gray, red, green and blue channels, whichever are appropriate for the given color type.

png_set_sRGB

Name

png_set_sRGB — set the rendering intent for given image

Synopsis

```
#include <png.h>
void png_set_sRGB(png_structp png_ptr, png_infop info_ptr, int
srgb_intent);
```

Description

png_set_sRGB() shall set the rendering intent of an image as specified by *srgb_intent* and shall set the sRGB chunk information to valid for the image. The presence of sRGB chunk implies that the pixel data is in the sRGB color space. *srgb_intent* can take one of the following values

```
PNG_sRGB_INTENT_SATURATION
PNG_sRGB_INTENT_PERCEPTUAL
PNG_sRGB_INTENT_ABSOLUTE
PNG_sRGB_INTENT_RELATIVE
```

png_set_sRGB_gAMA_and_cHRM**Name**

`png_set_sRGB_gAMA_and_cHRM` — set rendering intent, gamma values, and CIE chromaticities of a PNG file

Synopsis

```
#include <png.h>
void png_set_sRGB_gAMA_and_cHRM(png_structp png_ptr, png_info_ptr info_ptr,
int srgb_intent);
```

Description

This interface shall set the rendering intent, gamma values, and CIE chromaticities of a PNG file.

The parameter *png_ptr* shall specify the PNG file.

The parameter *info_ptr* shall specify the PNG info structure.

The parameter *srgb_intent* shall specify the rendering intent. Because the sRGB chunk is present, the pixel data uses the sRGB color space. This interface shall also write gAMA and cHRM chunks with values consistent with sRGB.

png_set_shift**Name**

`png_set_shift` — shift pixel values to valid bit-depth

Synopsis

```
#include <png.h>
void png_set_shift(png_structp png_ptr, png_color_8p true_bits);
```

Description

If image data in a row buffer is stored in a bit depth other than those supported by PNG, `png_set_shift()` shall scale the values to a valid bit-depth defined by PNG format. For example, 3-bit data in range 0-7 is scaled to 4-bit PNG.

png_set_sig_bytes**Name**

`png_set_sig_bytes` — number of bytes read from PNG file

Synopsis

```
#include <png.h>
void png_set_sig_bytes(png_structp png_ptr, int num_bytes);
```

Description

`png_set_sig_bytes()` shall store the number of bytes of the PNG file signature that have been read from the PNG stream.

Errors

`png_set_sig_bytes()` shall invoke error function if `num_bytes > 8`.

png_set_strip_16**Name**

png_set_strip_16 — strip 16 bit PNG file to 8 bit depth

Synopsis

```
#include <png.h>
void png_set_strip_16(png_structp png_ptr);
```

Description

png_set_strip_16() shall strip the pixels of a PNG stream with 16 bits per channel to 8 bits per channel.

png_set_strip_alpha**Name**

png_set_strip_alpha — remove alpha channel on the given image

Synopsis

```
#include <png.h>
void png_set_strip_alpha(png_structp png_ptr);
```

Description

png_set_strip_alpha() shall set transformation on the image to remove the alpha channel.

png_set_swap**Name**

png_set_swap — swap byte-order for 16 bit depth files

Synopsis

```
#include <png.h>
void png_set_swap(png_structp png_ptr);
```

Description

PNG files store 16-bit pixels in network byte order (big-endian, ie most significant bytes first). png_set_swap() shall switch the byte-order to little-endian (ie, least significant bits first).

png_set_swap_alpha**Name**

png_set_swap_alpha — swap image data from RGBA to ARGB format

Synopsis

```
#include <png.h>
void png_set_swap_alpha(png_structp png_ptr);
```

Description

png_set_swap_alpha() shall swap data for an image with an alpha channel from RGBA format to ARGB format.

png_set_tIME**Name**

png_set_tIME — set last modification time for the image

Synopsis

```
#include <png.h>
void png_set_tIME(png_structp png_ptr, png_infop info_ptr, png_timep
mod_time);
```

Description

png_set_sBIT shall set the time of last modification of the image in info_ptr as specified by mod_time.

png_set_tRNS**Name**

png_set_tRNS — set transparency values for images

Synopsis

```
#include <png.h>
void png_set_tRNS(png_structp png_ptr, png_infop info_ptr, png_bytep
trans, int num_trans, png_color_16p trans_values);
```

Description

png_set_tRNS() shall set the transparency data for paletted images and image types that don't need a full alpha channel. For a paletted image, png_set_tRNS() shall set the array of transparency values for the palette colors to "trans". The number of transparency entries is given by "num_trans". For non-paletted images, png_set_tRNS() shall set the single color value or graylevel to "trans_values"

png_set_tRNS_to_alpha

Name

png_set_tRNS_to_alpha — set expansion transformation

Synopsis

```
#include <png.h>
void png_set_tRNS_to_alpha(png_structp png_ptr);
```

Description

png_set_tRNS_to_alpha() shall set transformation in *png_ptr* such that tRNS chunks are expanded to alpha channels. png_set_tRNS_to_alpha() is actually an alias for png_set_expand().

png_set_text

Name

png_set_text — stores information for image comments

Synopsis

```
#include <png.h>
void png_set_text(png_structp png_ptr, png_info_ptr info_ptr, png_textp
text_ptr, int num_text);
```

Description

png_set_text() shall store information for image comments given in *text_ptr* to *info_ptr*. *text_ptr* is an array of size *num_text* of png_text structures whose member fields include:

compression

type of compression used on text. Valid values are:

```
PNG_TEXT_COMPRESSION_NONE
PNG_TEXT_COMPRESSION_zTXt
PNG_TEXT_COMPRESSION_NONE
PNG_ITXT_COMPRESSION_zTXt
```

key

keyword for comment. Must contain 1-79 characters.

text

text comment for current keyword. May be empty.

text_length

length of text string after decompression. 0 for iTXt.

png_set_unknown_chunk_location

Name

`png_set_unknown_chunk_location` — set the location of an unknown chunk in a PNG file

Synopsis

```
#include <png.h>
void png_set_unknown_chunk_location(png_structp png_ptr, png_infop
info_ptr, int chunk, int location);
```

Description

This interface shall set the location of an unknown chunk in a PNG file.

The parameter *png_ptr* shall specify the PNG file.

The parameter *info_ptr* shall specify the PNG info structure.

The parameter *chunk* shall specify the number of the chunk in the array of unknown chunks.

The parameter *location* shall specify the new location of the chunk within the PNG file.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

png_set_unknown_chunks

Name

png_set_unknown_chunks — insert unknown chunks into a PNG file

Synopsis

```
#include <png.h>
void png_set_unknown_chunks(png_structp png_ptr, png_info_ptr info_ptr,
    png_unknown_chunkp unknowns, int num_unknowns);
```

Description

This interface shall insert unknown chunks into a PNG file.

The parameter *png_ptr* shall specify the PNG file.

The parameter *info_ptr* shall specify the PNG info structure.

The parameter *unknowns* shall specify an array of *png_unknown_chunk* structures containing the unknown chunks, as described under *png_get_unknown_chunks()*. The *location* member of a *png_unknown_chunk* structure can take several special values (see below).

The parameter *num_unknowns* shall specify the number of unknown chunks.

The special values for the *location* members of the *png_unknown_chunk* structures are as follows.

0

do not write the chunk

PNG_HAVE_IHDR

insert chunk before PLTE

PNG_HAVE_PLTE

insert chunk before IDAT

PNG_AFTER_IDAT

insert chunk after IDAT

Notes

The *location* member of the *png_unknown_chunk* structure is set automatically depending on how much of the PNG file has been written. Its value can be changed after calling this interface. The chunk is placed within a location according to its position in the array of structures, as described under *png_get_unknown_chunks()*.

png_set_write_fn

Name

png_set_write_fn — set user-defined function for writing a PNG stream

Synopsis

```
#include <png.h>
void png_set_write_fn(png_structp png_ptr, png_voidp io_ptr,
png_rw_ptr write_data_fn, png_flush_ptr output_flush_fn);
```

Description

png_set_write_fn() sets the write_data_fn as the output function for writing PNG files instead of using standard C I/O stream functions. png_ptr - pointer to output data structure png_struct io_ptr - pointer to user-defined structure containing information about the output functions. This value may be NULL. write_data_fn - pointer to new output function that shall take the following arguments: - a pointer to a png_struct - a pointer to a structure where output data can be stored - 32-bit unsigned int to indicate number of bytes to write The output function should invoke png_error() to handle any fatal errors and png_warning() to handle non-fatal errors. flush_data_fn - pointer to a new flush function that shall take a pointer to a png_struct as argument. This function shall flush any remaining data in buffers used by the output function. If the output function does not buffer output, a function prototype must still be supplied.

png_set_write_status_fn

Name

png_set_write_status_fn — install custom callback function to be called after row is written

Synopsis

```
#include <png.h>
void png_set_write_status_fn(png_structp png_ptr,
png_write_status_ptr write_row_fn);
```

Description

This interface shall install a custom callback function to be called after a row has been written.

The parameter *png_ptr* shall specify the PNG file to be transformed.

The parameter *write_row_fn* shall specify the custom callback function.

png_set_write_user_transform_fn

Name

png_set_write_user_transform_fn — install a custom output transformation callback function

Synopsis

```
#include <png.h>
void      png_set_write_user_transform_fn(png_structp      png_ptr,
png_user_transform_ptr write_user_transform_fn);
```

Description

This interface shall install a custom output transformation callback function.

The parameter *png_ptr* shall specify the PNG file to be transformed.

The parameter *write_user_transform_fn* shall specify the custom callback function.

png_sig_cmp

Name

png_sig_cmp — match the PNG signature

Synopsis

```
#include <png.h>
int      png_sig_cmp(png_bytep      sig,      png_size_t      start,      png_size_t
num_to_check);
```

Description

png_sig_cmp() checks whether the given number of bytes match the PNG signature starting from the start position. The function shall return non-zero if num_to_check == 0 or start > 7.

Return Value

Zero - the given number of bytes starting from start position match the respective bytes of the PNG signature. Non-zero - the given number of bytes starting from start position do not match the respective bytes of the PNG signature or num_to_check == 0 or start > 7.

png_start_read_image

Name

png_start_read_image — start reading a PNG file

Synopsis

```
#include <png.h>
void png_start_read_image(png_structp png_ptr);
```

Description

This interface shall update the palette with the previously specified transformations, and then start reading the specified PNG file.

The parameter *png_ptr* shall specify the PNG file to read.

png_warning

Name

png_warning — default function to handle non-fatal errors

Synopsis

```
#include <png.h>
void png_warning(png_structp png_ptr, png_const_charp warning_message);
```

Description

png_warning() is the default function for handling non-fatal errors. The default function to handle warnings may be changed by using png_set_error_fn() to replace the warning function at run-time.

png_write_chunk

Name

png_write_chunk — write a PNG chunk

Synopsis

```
#include <png.h>
void png_write_chunk(png_structp png_ptr, png_bytep chunk_name,
png_bytep data, png_size_t length);
```

Description

png_write_chunk() writes the start of a PNG chunk, the chunk data and the end of the chunk all at once.

png_write_end**Name**

`png_write_end` — write the end of a PNG file

Synopsis

```
#include <png.h>
void png_write_end(png_structp png_ptr, png_infop info_ptr);
```

Description

`png_write_end()` writes the end of a PNG file to which the image data has already been written. The user may write time information or comments at the end of the PNG file.

png_write_flush**Name**

`png_write_flush` — flush the current output buffers

Synopsis

```
#include <png.h>
void png_write_flush(png_structp png_ptr);
```

Description

`png_write_flush()` shall the current output buffers for any pending data.

png_write_image**Name**

`png_write_image` — write the given image data

Synopsis

```
#include <png.h>
void png_write_image(png_structp png_ptr, png_bytepp image);
```

Description

Write the rows of given image data. If the image is not interlaced, the image shall be written in a single pass.

png_write_info

Name

png_write_info — write PNG information to file

Synopsis

```
#include <png.h>
void png_write_info(png_structp png_ptr, png_infop info_ptr);
```

Description

png_write_info() writes the PNG information in info_ptr to file.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

png_write_png

Name

png_write_png — write the entire PNG file

Synopsis

```
#include <png.h>
void png_write_png(png_structp png_ptr, png_infop info_ptr, int
transforms, png_voidp params);
```

Description

png_write_png() shall provide the high-level write operation. The function shall write the PNG stream if the entire image information is available in png_ptr. The integer "transforms" shall contain the logical OR of a set of the following transformation flags:

PNG_TRANSFORM_IDENTITY

No transformation

PNG_TRANSFORM_PACKING

Expand 1, 2 and 4-bit samples to bytes

PNG_TRANSFORM_PACKSWAP

Change order of packed pixels to LSB first

PNG_TRANSFORM_INVERT_MONO

Invert monochrome images

PNG_TRANSFORM_SHIFT

Normalize pixels to the sBIT depth

PNG_TRANSFORM_BGR

Flip RGB to BGR, RGBA to BGRA

PNG_TRANSFORM_SWAP_ALPHA

Flip RGBA to ARGB or GA to AG

PNG_TRANSFORM_INVERT_ALPHA

Change alpha from opacity to transparency

PNG_TRANSFORM_SWAP_ENDIAN

Byte-swap 16-bit samples

PNG_TRANSFORM_STRIP_FILLER

Strip off filler bytes

"params" is unused and must be set to NULL.

png_write_row

Name

png_write_row — write a row of image data

Synopsis

```
#include <png.h>
void png_write_row(png_structp png_ptr, png_bytep row);
```

Description

Process and write a row of image data. The header information must have been written before the image data can be written.

png_write_rows

Name

png_write_rows — write multiple rows of image data

Synopsis

```
#include <png.h>
void png_write_rows(png_structp png_ptr, png_bytepp row, png_uint_32
num_rows);
```

Description

Process and write "num_rows" rows of image data starting from "row".

V JPEG library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

9 Libraries

9.1 Interfaces for libjpeg

Table 9-1 defines the library name and shared object name for the libjpeg library

Table 9-1 libjpeg Definition

Library:	libjpeg
SONAME:	libjpeg.so.62

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

9.1.1 JPEG Reference library

9.1.1.1 Interfaces for JPEG Reference library

An LSB conforming implementation shall provide the generic functions for JPEG Reference library specified in Table 9-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 9-2 libjpeg - JPEG Reference library Function Interfaces

jpeg_CreateCompress(LIBJPEG_6.2) [LSB]	jpeg_CreateDecompress(LIBJPEG_6.2) [LSB]	jpeg_abort(LIBJPEG_6.2) [LSB]
jpeg_abort_compress(LIBJPEG_6.2) [LSB]	jpeg_abort_decompress(LIBJPEG_6.2) [LSB]	jpeg_add_quant_table(LIBJPEG_6.2) [LSB]
jpeg_alloc_huff_table(LIBJPEG_6.2) [LSB]	jpeg_alloc_quant_table(LIBJPEG_6.2) [LSB]	jpeg_calc_output_dimensions(LIBJPEG_6.2) [LSB]
jpeg_consume_input(LIBJPEG_6.2) [LSB]	jpeg_copy_critical_parameters(LIBJPEG_6.2) [LSB]	jpeg_default_colorspace(LIBJPEG_6.2) [LSB]
jpeg_destroy(LIBJPEG_6.2) [LSB]	jpeg_destroy_compress(LIBJPEG_6.2) [LSB]	jpeg_destroy_decompress(LIBJPEG_6.2) [LSB]
jpeg_finish_compress(LIBJPEG_6.2) [LSB]	jpeg_finish_decompress(LIBJPEG_6.2) [LSB]	jpeg_finish_output(LIBJPEG_6.2) [LSB]
jpeg_has_multiple_scans(LIBJPEG_6.2) [LSB]	jpeg_input_complete(LIBJPEG_6.2) [LSB]	jpeg_new_colormap(LIBJPEG_6.2) [LSB]
jpeg_quality_scaling(LIBJPEG_6.2) [LSB]	jpeg_read_coefficients(LIBJPEG_6.2) [LSB]	jpeg_read_header(LIBJPEG_6.2) [LSB]
jpeg_read_raw_data(LIBJPEG_6.2) [LSB]	jpeg_read_scanlines(LIBJPEG_6.2) [LSB]	jpeg_resync_to_restart(LIBJPEG_6.2) [LSB]
jpeg_save_markers(LIBJPEG_6.2) [LSB]	jpeg_set_colorspace(LIBJPEG_6.2) [LSB]	jpeg_set_defaults(LIBJPEG_6.2) [LSB]
jpeg_set_linear_quality(LIBJPEG_6.2) [LSB]	jpeg_set_marker_processor(LIBJPEG_6.2) [LSB]	jpeg_set_quality(LIBJPEG_6.2) [LSB]

jpeg_simple_progression(LIBJPEG_6.2) [LSB]	jpeg_start_compress(LIBJPEG_6.2) [LSB]	jpeg_start_decompress(LIBJPEG_6.2) [LSB]
jpeg_start_output(LIBJPEG_6.2) [LSB]	jpeg_std_error(LIBJPEG_6.2) [LSB]	jpeg_stdio_dest(LIBJPEG_6.2) [LSB]
jpeg_stdio_src(LIBJPEG_6.2) [LSB]	jpeg_suppress_tables(LIBJPEG_6.2) [LSB]	jpeg_write_coefficients(LIBJPEG_6.2) [LSB]
jpeg_write_m_byte(LIBJPEG_6.2) [LSB]	jpeg_write_m_header(LIBJPEG_6.2) [LSB]	jpeg_write_marker(LIBJPEG_6.2) [LSB]
jpeg_write_raw_data(LIBJPEG_6.2) [LSB]	jpeg_write_scanlines(LIBJPEG_6.2) [LSB]	jpeg_write_tables(LIBJPEG_6.2) [LSB]

9.2 Data Definitions for libjpeg

This section defines global identifiers and their values that are associated with interfaces contained in libjpeg. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

9.2.1 jerror.h

```
#define JERROR_H
#define MAKESTMT(stuff) do { stuff } while (0)
#define TRACEMS(cinfo, lvl, code) \
    ((cinfo)->err->msg_code = (code), \
    ( *(cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)))
#define WARNMS(cinfo, code) \
    ((cinfo)->err->msg_code = (code), \
    ( *(cinfo)->err->emit_message) ((j_common_ptr) (cinfo), -1))
#define ERREXIT(cinfo, code) \
    ((cinfo)->err->msg_code = (code), \
    ( *(cinfo)->err->error_exit) ((j_common_ptr) (cinfo)))
#define TRACEMS1(cinfo, lvl, code, p1) \
    ((cinfo)->err->msg_code = (code), \
    (cinfo)->err->msg_parm.i[0] = (p1), \
    ( *(cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)))
#define ERREXIT1(cinfo, code, p1) \
    ((cinfo)->err->msg_code = (code), \
    (cinfo)->err->msg_parm.i[0] = (p1), \
    ( *(cinfo)->err->error_exit) ((j_common_ptr) (cinfo)))
#define TRACEMS2(cinfo, lvl, code, p1, p2) \
    ((cinfo)->err->msg_code = (code), \
    (cinfo)->err->msg_parm.i[0] = (p1), \
    (cinfo)->err->msg_parm.i[1] = (p2), \
    ( *(cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)))
#define WARNMS2(cinfo, code, p1, p2) \
```

```

((cinfo)->err->msg_code = (code), \
(cinfo)->err->msg_parm.i[0] = (p1), \
(cinfo)->err->msg_parm.i[1] = (p2), \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), -1))
#define ERREXIT2(cinfo,code,p1,p2) \
((cinfo)->err->msg_code = (code), \
(cinfo)->err->msg_parm.i[0] = (p1), \
(cinfo)->err->msg_parm.i[1] = (p2), \
(* (cinfo)->err->error_exit) ((j_common_ptr) (cinfo)))
#define ERREXIT3(cinfo,code,p1,p2,p3) \
((cinfo)->err->msg_code = (code), \
(cinfo)->err->msg_parm.i[0] = (p1), \
(cinfo)->err->msg_parm.i[1] = (p2), \
(cinfo)->err->msg_parm.i[2] = (p3), \
(* (cinfo)->err->error_exit) ((j_common_ptr) (cinfo)))
#define ERREXIT4(cinfo,code,p1,p2,p3,p4) \
((cinfo)->err->msg_code = (code), \
(cinfo)->err->msg_parm.i[0] = (p1), \
(cinfo)->err->msg_parm.i[1] = (p2), \
(cinfo)->err->msg_parm.i[2] = (p3), \
(cinfo)->err->msg_parm.i[3] = (p4), \
(* (cinfo)->err->error_exit) ((j_common_ptr) (cinfo)))
#define TRACEMSS(cinfo,lvl,code,str) \
((cinfo)->err->msg_code = (code), \
strncpy((cinfo)->err->msg_parm.s, (str), JMSG_STR_PARM_MAX), \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)))
#define ERREXITS(cinfo,code,str) \
((cinfo)->err->msg_code = (code), \
strncpy((cinfo)->err->msg_parm.s, (str), JMSG_STR_PARM_MAX), \
(* (cinfo)->err->error_exit) ((j_common_ptr) (cinfo)))
#define WARNMS1(cinfo,code,p1) \
((cinfo)->err->msg_code = (code), \
(cinfo)->err->msg_parm.i[0] = (p1), \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), -1))
#define TRACEMS3(cinfo,lvl,code,p1,p2,p3) \
MAKESTMT(int * _mp = (cinfo)->err->msg_parm.i; \
_mp[0] = (p1); _mp[1] = (p2); _mp[2] = (p3); \
(cinfo)->err->msg_code = (code); \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)); )
#define TRACEMS4(cinfo,lvl,code,p1,p2,p3,p4) \
MAKESTMT(int * _mp = (cinfo)->err->msg_parm.i; \
_mp[0] = (p1); _mp[1] = (p2); _mp[2] = (p3); _mp[3] = (p4); \
(cinfo)->err->msg_code = (code); \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)); )
#define TRACEMS5(cinfo,lvl,code,p1,p2,p3,p4,p5) \
MAKESTMT(int * _mp = (cinfo)->err->msg_parm.i; \
_mp[0] = (p1); _mp[1] = (p2); _mp[2] = (p3); _mp[3] = (p4); \
_mp[4] = (p5); \
(cinfo)->err->msg_code = (code); \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)); )
#define TRACEMS8(cinfo,lvl,code,p1,p2,p3,p4,p5,p6,p7,p8) \
MAKESTMT(int * _mp = (cinfo)->err->msg_parm.i; \
_mp[0] = (p1); _mp[1] = (p2); _mp[2] = (p3); _mp[3] = (p4); \
_mp[4] = (p5); _mp[5] = (p6); _mp[6] = (p7); _mp[7] = (p8); \
(cinfo)->err->msg_code = (code); \
(* (cinfo)->err->emit_message) ((j_common_ptr) (cinfo), (lvl)); )

typedef enum {
    JMSG_NOMESSAGE = 0,
    JERR_ARITH_NOTIMPL = 1,
    JERR_BAD_ALIGN_TYPE = 2,
    JERR_BAD_ALLOC_CHUNK = 3,
    JERR_BAD_BUFFER_MODE = 4,
    JERR_BAD_COMPONENT_ID = 5,
    JERR_BAD_CROP_SPEC = 6,
    JERR_BAD_DCT_COEF = 7,

```

JERR_BAD_DCTSIZE = 8,
 JERR_BAD_HUFF_TABLE = 9,
 JERR_BAD_IN_COLORSPACE = 10,
 JERR_BAD_J_COLORSPACE = 11,
 JERR_BAD_LENGTH = 12,
 JERR_BAD_LIB_VERSION = 13,
 JERR_BAD_MCU_SIZE = 14,
 JERR_BAD_POOL_ID = 15,
 JERR_BAD_PRECISION = 16,
 JERR_BAD_PROGRESSION = 17,
 JERR_BAD_PROG_SCRIPT = 18,
 JERR_BAD_SAMPLING = 19,
 JERR_BAD_SCAN_SCRIPT = 20,
 JERR_BAD_STATE = 21,
 JERR_BAD_STRUCT_SIZE = 22,
 JERR_BAD_VIRTUAL_ACCESS = 23,
 JERR_BUFFER_SIZE = 24,
 JERR_CANT_SUSPEND = 25,
 JERR_CCIR601_NOTIMPL = 26,
 JERR_COMPONENT_COUNT = 27,
 JERR_CONVERSION_NOTIMPL = 28,
 JERR_DAC_INDEX = 29,
 JERR_DAC_VALUE = 30,
 JERR_DHT_INDEX = 31,
 JERR_DQT_INDEX = 32,
 JERR_EMPTY_IMAGE = 33,
 JERR_EMS_READ = 34,
 JERR_EMS_WRITE = 35,
 JERR_EOI_EXPECTED = 36,
 JERR_FILE_READ = 37,
 JERR_FILE_WRITE = 38,
 JERR_FRACT_SAMPLE_NOTIMPL = 39,
 JERR_HUFF_CLEN_OVERFLOW = 40,
 JERR_HUFF_MISSING_CODE = 41,
 JERR_IMAGE_TOO_BIG = 42,
 JERR_INPUT_EMPTY = 43,
 JERR_INPUT_EOF = 44,
 JERR_MISMATCHED_QUANT_TABLE = 45,
 JERR_MISSING_DATA = 46,
 JERR_MODE_CHANGE = 47,
 JERR_NOTIMPL = 48,
 JERR_NOT_COMPILED = 49,
 JERR_NO_BACKING_STORE = 50,
 JERR_NO_HUFF_TABLE = 51,
 JERR_NO_IMAGE = 52,
 JERR_NO_QUANT_TABLE = 53,
 JERR_NO_SOI = 54,
 JERR_OUT_OF_MEMORY = 55,
 JERR_QUANT_COMPONENTS = 56,
 JERR_QUANT_FEW_COLORS = 57,
 JERR_QUANT_MANY_COLORS = 58,
 JERR_SOF_DUPLICATE = 59,
 JERR_SOF_NO_SOS = 60,
 JERR_SOF_UNSUPPORTED = 61,
 JERR_SOI_DUPLICATE = 62,
 JERR_SOS_NO_SOF = 63,
 JERR_TFILE_CREATE = 64,
 JERR_TFILE_READ = 65,
 JERR_TFILE_SEEK = 66,
 JERR_TFILE_WRITE = 67,
 JERR_TOO_LITTLE_DATA = 68,
 JERR_UNKNOWN_MARKER = 69,
 JERR_VIRTUAL_BUG = 70,
 JERR_WIDTH_OVERFLOW = 71,
 JERR_XMS_READ = 72,
 JERR_XMS_WRITE = 73,

```

JMSG_COPYRIGHT = 74,
JMSG_VERSION = 75,
JTRC_16BIT_TABLES = 76,
JTRC_ADOBE = 77,
JTRC_APP0 = 78,
JTRC_APP14 = 79,
JTRC_DAC = 80,
JTRC_DHT = 81,
JTRC_DQT = 82,
JTRC_DRI = 83,
JTRC_EMS_CLOSE = 84,
JTRC_EMS_OPEN = 85,
JTRC_EOI = 86,
JTRC_HUFFBITS = 87,
JTRC_JFIF = 88,
JTRC_JFIF_BADTHUMBNAIISIZE = 89,
JTRC_JFIF_EXTENSION = 90,
JTRC_JFIF_THUMBNAIIL = 91,
JTRC_MISC_MARKER = 92,
JTRC_PARMLESS_MARKER = 93,
JTRC_QUANTVALS = 94,
JTRC_QUANT_3_NCOLORS = 95,
JTRC_QUANT_NCOLORS = 96,
JTRC_QUANT_SELECTED = 97,
JTRC_RECOVERY_ACTION = 98,
JTRC_RST = 99,
JTRC_SMOOTH_NOTIMPL = 100,
JTRC_SOF = 101,
JTRC_SOF_COMPONENT = 102,
JTRC_SOI = 103,
JTRC_SOS = 104,
JTRC_SOS_COMPONENT = 105,
JTRC_SOS_PARAMS = 106,
JTRC_TFILE_CLOSE = 107,
JTRC_TFILE_OPEN = 108,
JTRC_THUMB_JPEG = 109,
JTRC_THUMB_PALETTE = 110,
JTRC_THUMB_RGB = 111,
JTRC_UNKNOWN_IDS = 112,
JTRC_XMS_CLOSE = 113,
JTRC_XMS_OPEN = 114,
JWRN_ADOBE_XFORM = 115,
JWRN_BOGUS_PROGRESSION = 116,
JWRN_EXTRANEIOUS_DATA = 117,
JWRN_HIT_MARKER = 118,
JWRN_HUFF_BAD_CODE = 119,
JWRN_JFIF_MAJOR = 120,
JWRN_JPEG_EOF = 121,
JWRN_MUST_RESYNC = 122,
JWRN_NOT_SEQUENTIAL = 123,
JWRN_TOO_MUCH_DATA = 124,
JMSG_LASTMSGCODE = 125
} J_MESSAGE_CODE;

```

9.2.2 jpeglib.h

```

#define FAR
#define HAVE_PROTOTYPES
#define HAVE_UNSIGNED_CHAR
#define HAVE_UNSIGNED_SHORT
#define jpeg_create_compress(cinfo) \
    jpeg_CreateCompress((cinfo), JPEG_LIB_VERSION, (size_t) \
    sizeof(struct \
        jpeg_compress_struct))
#define jpeg_create_decompress(cinfo) \

```



```

        jpeg_CreateDecompress((cinfo), JPEG_LIB_VERSION, (size_t) \
        sizeof(struct jpeg_decompress_struct))
#define jpeg_common_fields \
        struct jpeg_error_mgr * err; struct jpeg_memory_mgr * mem;
struct \
        jpeg_progress_mgr * progress; void * client_data; boolean \
        is_decompressor; int global_state
#define GETJSAMPLE(value) ((int) (value))
#define GETJOCTET(value) (value)
#define JPEG_SUSPENDED 0
#define JPOOL_PERMANENT 0
#define JPEG_RST0 0xD0
#define JPEG_EOI 0xD9
#define JPEG_APP0 0xE0
#define JPEG_COM 0xFE
#define JPEG_HEADER_OK 1
#define JPEG_REACHED_SOS 1
#define JPOOL_IMAGE 1
#define C_MAX_BLOCKS_IN_MCU 10
#define D_MAX_BLOCKS_IN_MCU 10
#define MAX_COMPONENTS 10
#define CENTERJSAMPLE 128
#define NUM_ARITH_TBLS 16
#define JPEG_HEADER_TABLES_ONLY 2
#define JPEG_REACHED_EOI 2
#define JPOOL_NUMPOOLS 2
#define JMSG_LENGTH_MAX 200
#define MAXJSAMPLE 255
#define JPEG_ROW_COMPLETED 3
#define JPEG_SCAN_COMPLETED 4
#define MAX_COMPS_IN_SCAN 4
#define MAX_SAMP_FACTOR 4
#define NUM_HUFF_TBLS 4
#define NUM_QUANT_TBLS 4
#define JPEG_LIB_VERSION 62
#define DCTSIZE2 64
#define JPEG_MAX_DIMENSION 65500L
#define BITS_IN_JSAMPLE 8
#define DCTSIZE 8
#define JMSG_STR_PARM_MAX 80
#define JPP(arglist) arglist
#define EXTERN(type) extern type
#define JDCT_FASTEST JDCT_IFAST
#define JDCT_DEFAULT JDCT_ISLOW
#define LOCAL(type) static type
#define METHODDEF(type) static type
#define GLOBAL(type) type
#define JMETHOD(type,methodname,arglist) type (*methodname)
arglist

struct jpeg_decompress_struct {
        struct jpeg_error_mgr *err;
        struct jpeg_memory_mgr *mem;
        struct jpeg_progress_mgr *progress;
        void *client_data;
        boolean is_decompressor;
        int global_state;
        struct jpeg_source_mgr *src;
        JDIMENSION image_width;
        JDIMENSION image_height;
        int num_components;
        J_COLOR_SPACE jpeg_color_space;
        J_COLOR_SPACE out_color_space;
        unsigned int scale_num;
        unsigned int scale_denom;
        double output_gamma;

```

```

boolean buffered_image;
boolean raw_data_out;
J_DCT_METHOD dct_method;
boolean do_fancy_upsampling;
boolean do_block_smoothing;
boolean quantize_colors;
J_DITHER_MODE dither_mode;
boolean two_pass_quantize;
int desired_number_of_colors;
boolean enable_1pass_quant;
boolean enable_external_quant;
boolean enable_2pass_quant;
JDIMENSION output_width;
JDIMENSION output_height;
int out_color_components;
int output_components;
int rec_outbuf_height;
int actual_number_of_colors;
JSAMPARRAY colormap;
JDIMENSION output_scanline;
int input_scan_number;
JDIMENSION input_iMCU_row;
int output_scan_number;
JDIMENSION output_iMCU_row;
int *(coef_bits)[64];
JQUANT_TBL *quant_tbl_ptrs[4];
JHUFF_TBL *dc_huff_tbl_ptrs[4];
JHUFF_TBL *ac_huff_tbl_ptrs[4];
int data_precision;
jpeg_component_info *comp_info;
boolean progressive_mode;
boolean arith_code;
UINT8 arith_dc_L[16];
UINT8 arith_dc_U[16];
UINT8 arith_ac_K[16];
unsigned int restart_interval;
boolean saw_JFIF_marker;
UINT8 JFIF_major_version;
UINT8 JFIF_minor_version;
UINT8 density_unit;
UINT16 X_density;
UINT16 Y_density;
boolean saw_Adobe_marker;
UINT8 Adobe_transform;
boolean CCIR601_sampling;
jpeg_saved_marker_ptr marker_list;
int max_h_samp_factor;
int max_v_samp_factor;
int min_DCT_scaled_size;
JDIMENSION total_iMCU_rows;
JSAMPLE *sample_range_limit;
int comps_in_scan;
jpeg_component_info *cur_comp_info[4];
JDIMENSION MCUs_per_row;
JDIMENSION MCU_rows_in_scan;
int blocks_in_MCU;
int MCU_membership[10];
int Ss;
int Se;
int Ah;
int Al;
int unread_marker;
struct jpeg_decomp_master *master;
struct jpeg_d_main_controller *main;
struct jpeg_d_coef_controller *coef;
struct jpeg_d_post_controller *post;

```

```

    struct jpeg_input_controller *inputctl;
    struct jpeg_marker_reader *marker;
    struct jpeg_entropy_decoder *entropy;
    struct jpeg_inverse_dct *idct;
    struct jpeg_upsampler *upsample;
    struct jpeg_color_deconverter *cconvert;
    struct jpeg_color_quantizer *cquantize;
};

struct jpeg_error_mgr {
    void (*error_exit) (j_common_ptr);
    void (*emit_message) (j_common_ptr, int);
    void (*output_message) (j_common_ptr);
    void (*format_message) (j_common_ptr, char *);
    void (*reset_error_mgr) (j_common_ptr);
    int msg_code;
    union {
        int i[8];
        char s[80];
    } msg_parm;
    int trace_level;
    long int num_warnings;
    const char *const *jpeg_message_table;
    int last_jpeg_message;
    const char *const *addon_message_table;
    int first_addon_message;
    int last_addon_message;
};

struct jpeg_common_struct {
    struct jpeg_error_mgr *err;
    struct jpeg_memory_mgr *mem;
    struct jpeg_progress_mgr *progress;
    void *client_data;
    boolean is_decompressor;
    int global_state;
};

struct jpeg_memory_mgr {
    void *(*alloc_small) (j_common_ptr, int, size_t);
    void *(*alloc_large) (j_common_ptr, int, size_t);
    JSAMPARRAY(*alloc_sarray) (j_common_ptr, int, JDIMENSION,
JDIMENSION);
    JBLOCKARRAY(*alloc_barray) (j_common_ptr, int, JDIMENSION,
JDIMENSION);
    jvirt_sarray_ptr(*request_virt_sarray) (j_common_ptr, int,
boolean,
JDIMENSION, JDIMENSION,
JDIMENSION);
    jvirt_barray_ptr(*request_virt_barray) (j_common_ptr, int,
boolean,
JDIMENSION, JDIMENSION,
JDIMENSION);
    void (*realize_virt_arrays) (j_common_ptr);
    JSAMPARRAY(*access_virt_sarray) (j_common_ptr,
jvirt_sarray_ptr,
JDIMENSION, JDIMENSION, boolean);
    JBLOCKARRAY(*access_virt_barray) (j_common_ptr,
jvirt_barray_ptr,
JDIMENSION, JDIMENSION, boolean);
    void (*free_pool) (j_common_ptr, int);
    void (*self_destruct) (j_common_ptr);
    long int max_memory_to_use;
    long int max_alloc_chunk;
};

typedef struct jpeg_common_struct *j_common_ptr;
typedef unsigned char JSAMPLE;
typedef JSAMPLE *JSAMPROW;
typedef JSAMPROW *JSAMPARRAY;

```

```

typedef unsigned int JDIMENSION;
typedef short int JCOEF;
typedef JCOEF JBLOCK[64];
typedef JBLOCK *JBLOCKROW;
typedef JBLOCKROW *JBLOCKARRAY;
typedef struct jvirt_sarray_control *jvirt_sarray_ptr;
typedef int boolean;
typedef struct jvirt_barray_control *jvirt_barray_ptr;
struct jpeg_progress_mgr {
    void (*progress_monitor) (j_common_ptr);
    long int pass_counter;
    long int pass_limit;
    int completed_passes;
    int total_passes;
};
struct jpeg_source_mgr {
    const JOCTET *next_input_byte;
    size_t bytes_in_buffer;
    void (*init_source) (j_decompress_ptr);
    boolean (*fill_input_buffer) (j_decompress_ptr);
    void (*skip_input_data) (j_decompress_ptr, long int);
    boolean (*resync_to_restart) (j_decompress_ptr, int);
    void (*term_source) (j_decompress_ptr);
};
typedef unsigned char JOCTET;
typedef struct jpeg_decompress_struct *j_decompress_ptr;
typedef enum {
    JCS_UNKNOWN = 0,
    JCS_GRAYSCALE = 1,
    JCS_RGB = 2,
    JCS_YCbCr = 3,
    JCS_CMYK = 4,
    JCS_YCCK = 5
} J_COLOR_SPACE;
typedef enum {
    JDCT_ISLOW = 0,
    JDCT_IFAST = 1,
    JDCT_FLOAT = 2
} J_DCT_METHOD;
typedef enum {
    JDITHER_NONE = 0,
    JDITHER_ORDERED = 1,
    JDITHER_FS = 2
} J_DITHER_MODE;
typedef short unsigned int UINT16;
typedef struct {
    UINT16 quantval[64];
    boolean sent_table;
} JQUANT_TBL;
typedef unsigned char UINT8;
typedef struct {
    UINT8 bits[17];
    UINT8 huffval[256];
    boolean sent_table;
} JHUFF_TBL;
typedef struct {
    int component_id;
    int component_index;
    int h_samp_factor;
    int v_samp_factor;
    int quant_tbl_no;
    int dc_tbl_no;
    int ac_tbl_no;
    JDIMENSION width_in_blocks;
    JDIMENSION height_in_blocks;
    int DCT_scaled_size;

```

```

        JDIMENSION downsampled_width;
        JDIMENSION downsampled_height;
        boolean component_needed;
        int MCU_width;
        int MCU_height;
        int MCU_blocks;
        int MCU_sample_width;
        int last_col_width;
        int last_row_height;
        JQUANT_TBL *quant_table;
        void *dct_table;
    } jpeg_component_info;
    struct jpeg_marker_struct {
        jpeg_saved_marker_ptr next;
        UINT8 marker;
        unsigned int original_length;
        unsigned int data_length;
        JOCTET *data;
    };
    typedef struct jpeg_marker_struct *jpeg_saved_marker_ptr;
    struct jpeg_compress_struct {
        struct jpeg_error_mgr *err;
        struct jpeg_memory_mgr *mem;
        struct jpeg_progress_mgr *progress;
        void *client_data;
        boolean is_decompressor;
        int global_state;
        struct jpeg_destination_mgr *dest;
        JDIMENSION image_width;
        JDIMENSION image_height;
        int input_components;
        J_COLOR_SPACE in_color_space;
        double input_gamma;
        int data_precision;
        int num_components;
        J_COLOR_SPACE jpeg_color_space;
        jpeg_component_info *comp_info;
        JQUANT_TBL *quant_tbl_ptrs[4];
        JHUFF_TBL *dc_huff_tbl_ptrs[4];
        JHUFF_TBL *ac_huff_tbl_ptrs[4];
        UINT8 arith_dc_L[16];
        UINT8 arith_dc_U[16];
        UINT8 arith_ac_K[16];
        int num_scans;
        const jpeg_scan_info *scan_info;
        boolean raw_data_in;
        boolean arith_code;
        boolean optimize_coding;
        boolean CCIR601_sampling;
        int smoothing_factor;
        J_DCT_METHOD dct_method;
        unsigned int restart_interval;
        int restart_in_rows;
        boolean write_JFIF_header;
        UINT8 JFIF_major_version;
        UINT8 JFIF_minor_version;
        UINT8 density_unit;
        UINT16 X_density;
        UINT16 Y_density;
        boolean write_Adobe_marker;
        JDIMENSION next_scanline;
        boolean progressive_mode;
        int max_h_samp_factor;
        int max_v_samp_factor;
        JDIMENSION total_iMCU_rows;
        int comps_in_scan;

```

```

jpeg_component_info *cur_comp_info[4];
JDIMENSION MCUs_per_row;
JDIMENSION MCU_rows_in_scan;
int blocks_in_MCU;
int MCU_membership[10];
int Ss;
int Se;
int Ah;
int Al;
struct jpeg_comp_master *master;
struct jpeg_c_main_controller *main;
struct jpeg_c_prep_controller *prep;
struct jpeg_c_coef_controller *coef;
struct jpeg_marker_writer *marker;
struct jpeg_color_converter *cconvert;
struct jpeg_downsampler *downsample;
struct jpeg_forward_dct *fdct;
struct jpeg_entropy_encoder *entropy;
jpeg_scan_info *script_space;
int script_space_size;
};
struct jpeg_destination_mgr {
    JOCTET *next_output_byte;
    size_t free_in_buffer;
    void (*init_destination) (j_compress_ptr);
    boolean (*empty_output_buffer) (j_compress_ptr);
    void (*term_destination) (j_compress_ptr);
};
typedef struct jpeg_compress_struct *j_compress_ptr;
typedef struct {
    int comps_in_scan;
    int component_index[4];
    int Ss;
    int Se;
    int Ah;
    int Al;
} jpeg_scan_info;
typedef JSAMPARRAY *JSAMPIMAGE;
typedef boolean (*jpeg_marker_parser_method) (j_decompress_ptr);
extern void jpeg_CreateCompress(j_compress_ptr, int, size_t);
extern void jpeg_CreateDecompress(j_decompress_ptr, int, size_t);
extern void jpeg_abort(j_common_ptr);
extern void jpeg_abort_compress(j_compress_ptr);
extern void jpeg_abort_decompress(j_decompress_ptr);
extern void jpeg_add_quant_table(j_compress_ptr, int, const
unsigned int *,
                                int, boolean);
extern JHUFF_TBL *jpeg_alloc_huff_table(j_common_ptr);
extern JQUANT_TBL *jpeg_alloc_quant_table(j_common_ptr);
extern void jpeg_calc_output_dimensions(j_decompress_ptr);
extern int jpeg_consume_input(j_decompress_ptr);
extern void jpeg_copy_critical_parameters(j_decompress_ptr,
j_compress_ptr);
extern void jpeg_default_colorspace(j_compress_ptr);
extern void jpeg_destroy(j_common_ptr);
extern void jpeg_destroy_compress(j_compress_ptr);
extern void jpeg_destroy_decompress(j_decompress_ptr);
extern void jpeg_finish_compress(j_compress_ptr);
extern boolean jpeg_finish_decompress(j_decompress_ptr);
extern boolean jpeg_finish_output(j_decompress_ptr);
extern boolean jpeg_has_multiple_scans(j_decompress_ptr);
extern boolean jpeg_input_complete(j_decompress_ptr);
extern void jpeg_new_colormap(j_decompress_ptr);
extern int jpeg_quality_scaling(int);
extern jvirt_barray_ptr *jpeg_read_coefficients(j_decompress_ptr);
extern int jpeg_read_header(j_decompress_ptr, boolean);

```

```

extern JDIMENSION jpeg_read_raw_data(j_decompress_ptr, JSAMPIMAGE,
                                      JDIMENSION);
extern JDIMENSION jpeg_read_scanlines(j_decompress_ptr, JSAMPARRAY,
                                      JDIMENSION);

extern boolean jpeg_resync_to_restart(j_decompress_ptr, int);
extern void jpeg_save_markers(j_decompress_ptr, int, unsigned int);
extern void jpeg_set_colorspace(j_compress_ptr, J_COLOR_SPACE);
extern void jpeg_set_defaults(j_compress_ptr);
extern void jpeg_set_linear_quality(j_compress_ptr, int, boolean);
extern void jpeg_set_marker_processor(j_decompress_ptr, int,
                                      jpeg_marker_parser_method);
extern void jpeg_set_quality(j_compress_ptr, int, boolean);
extern void jpeg_simple_progression(j_compress_ptr);
extern void jpeg_start_compress(j_compress_ptr, boolean);
extern boolean jpeg_start_decompress(j_decompress_ptr);
extern boolean jpeg_start_output(j_decompress_ptr, int);
extern struct jpeg_error_mgr *jpeg_std_error(struct jpeg_error_mgr
*);
extern void jpeg_stdio_dest(j_compress_ptr, FILE *);
extern void jpeg_stdio_src(j_decompress_ptr, FILE *);
extern void jpeg_suppress_tables(j_compress_ptr, boolean);
extern void jpeg_write_coefficients(j_compress_ptr,
jvirt_barray_ptr *);
extern void jpeg_write_m_byte(j_compress_ptr, int);
extern void jpeg_write_m_header(j_compress_ptr, int, unsigned int);
extern void jpeg_write_marker(j_compress_ptr, int, const JOCTET *,
                             unsigned int);

extern JDIMENSION jpeg_write_raw_data(j_compress_ptr, JSAMPIMAGE,
                                      JDIMENSION);
extern JDIMENSION jpeg_write_scanlines(j_compress_ptr, JSAMPARRAY,
                                      JDIMENSION);
extern void jpeg_write_tables(j_compress_ptr);

```

9.3 Interface Definitions for libjpeg

The interfaces defined on the following pages are included in libjpeg and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 9.1 shall behave as described in the referenced base document.

jpeg_CreateCompress

Name

jpeg_CreateCompress — allocate and initialize a JPEG compression object

Synopsis

```
#include <jpeglib.h>
void jpeg_CreateCompress(j_compress_ptr cinfo, int version, size_t
size);
```

Description

The function `jpeg_CreateCompress()` shall allocate and initialize a JPEG compression object of type `struct jpeg_compress_struct`. The error handler structure of type `struct jpeg_error_mgr` must be initialized before making a call to `jpeg_CreateCompress()`. `JPEG_LIB_VERSION` and size of the structure used to hold the JPEG compression object `struct jpeg_compress_struct` must be passed in arguments `version` and `size` respectively.

Errors

`jpeg_CreateCompress()` shall exit via the error handler if it fails to allocate memory.

jpeg_CreateDecompress

Name

jpeg_CreateDecompress — allocate and initialize a JPEG decompression object

Synopsis

```
#include <jpeglib.h>
void jpeg_CreateDecompress(j_decompress_ptr cinfo, int version, size_t
size);
```

Description

The function `jpeg_CreateDecompress()` shall allocate and initialize a JPEG decompression object of type `struct jpeg_decompress_struct`. The error handler structure of type `struct jpeg_error_mgr` must be initialized before making a call to `jpeg_CreateDecompress()`. `JPEG_LIB_VERSION` and size of the structure used to hold the JPEG compression object `struct jpeg_decompress_struct` must be passed in arguments `version` and `size` respectively.

Errors

`jpeg_CreateDecompress()` shall exit via the error handler if it fails to allocate memory.

jpeg_abort

Name

jpeg_abort — aborts the compression or decompression cycle

Synopsis

```
#include <jpeglib.h>
void jpeg_abort(j_common_ptr cinfo);
```

Description

The function jpeg_abort() shall abort the processing of a compression or decompression operation. The function shall return the given object to an idle state releasing any working memory. The object may be reused after a call to jpeg_abort().

jpeg_abort_compress

Name

jpeg_abort_compress — abort processing of JPEG compression operation

Synopsis

```
#include <jpeglib.h>
void jpeg_abort_compress(j_compress_ptr cinfo);
```

Description

The function jpeg_abort_compress() shall abort the processing of a JPEG compression operation. The function shall return the given compression object to an idle state releasing any working memory. The compression object may be reused after a call to jpeg_abort_compress().

jpeg_abort_decompress

Name

jpeg_abort_decompress — abort processing of JPEG decompression operation

Synopsis

```
#include <jpeglib.h>
void jpeg_abort_decompress(j_decompress_ptr cinfo);
```

Description

The function jpeg_abort_decompress() shall abort the processing of a JPEG decompression operation. The function shall return the given decompression object to an idle state releasing any working memory. The decompression object may be reused after a call to jpeg_abort_decompress().

jpeg_add_quant_table**Name**

jpeg_add_quant_table — add a quantization table

Synopsis

```
#include <jpeglib.h>
void jpeg_add_quant_table(j_compress_ptr cinfo, int table_slot, const
unsigned int * basic_table, int scale_factor, boolean force_baseline);
```

Description

The function jpeg_add_quant_table() shall add a quantization table parameter at the "table_slot" index of the quantization table array. "basic_table" points to an array of 64 unsigned ints given in normal array order. These values are multiplied by "scale_factor"/100. The values are limited to the range 1..65535. If "force_baseline" is TRUE, the computed quantization table entries are limited to 1..255 for JPEG baseline compatibility.

Errors

jpeg_add_quant_table() shall return error and exit if "table_slot" is >= NUM_QUANT_TBLS. jpeg_add_quant_table() shall return error and exit if the state of the compression object indicates that the compression cycle has started.

jpeg_alloc_huff_table**Name**

jpeg_alloc_huff_table — allocate Huffman table

Synopsis

```
#include <jpeglib.h>
JHUFF_TBL * jpeg_alloc_huff_table(j_common_ptr cinfo);
```

Description

jpeg_alloc_huff_table() shall allocate memory for a structure holding Huffman coding table.

Return Value

jpeg_alloc_huff_table() shall return a pointer to allocated data structure for the Huffman coding table.

jpeg_alloc_quant_table

Name

jpeg_alloc_quant_table — allocate quantization table

Synopsis

```
#include <jpeglib.h>
JQUANT_TBL * jpeg_alloc_quant_table(j_common_ptr cinfo);
```

Description

jpeg_alloc_quant_table() shall allocate memory for a structure holding quantization table.

Return Value

jpeg_alloc_quant_table() shall return a pointer to allocated data structure for the quantization table.

jpeg_calc_output_dimensions

Name

jpeg_calc_output_dimensions — compute output image dimensions

Synopsis

```
#include <jpeglib.h>
void jpeg_calc_output_dimensions(j_decompress_ptr cinfo);
```

Description

jpeg_calc_output_dimensions() computes the output image dimensions based on the current compression parameter settings in the compression object. The following dimensions shall be computed:

output_width

actual width of output image

output_height

actual height of output image

out_color_components

number of color components in out_color_space

output_components

number of color components returned

rec_outbuf_height

recommended height of scanline buffer

jpeg_consume_input

Name

jpeg_consume_input — decode input data

Synopsis

```
#include <jpeglib.h>
int jpeg_consume_input(j_decompress_ptr cinfo);
```

Description

jpeg_consume_input() shall decode the input data as it arrives, even if it is not to be displayed.

Return Value

jpeg_consume_input() shall return with one of the following return codes:

JPEG_REACHED_SOS

reached SOS marker

JPEG_REACHED_EOI

reached EOI marker

JPEG_ROW_COMPLETED

completed reading one MCU row of compressed data

JPEG_SCAN_COMPLETED

completed reading last MCU row of compressed scan

JPEG_SUSPENDED

if data source module requests suspension of the decompressor.

jpeg_copy_critical_parameters

Name

jpeg_copy_critical_parameters — copy critical compression parameters from source decompression object

Synopsis

```
#include <jpeglib.h>
void jpeg_copy_critical_parameters(j_decompress_ptr srcinfo,
j_compress_ptr dstinfo);
```

Description

jpeg_copy_critical_parameters() shall initialize the compression object with default parameters, then copy the parameters needed for lossless transcoding from the source decompression object.

jpeg_default_colorspace

Name

jpeg_default_colorspace — set default colorspace based on input image's color space

Synopsis

```
#include <jpeglib.h>
void jpeg_default_colorspace(j_compress_ptr cinfo);
```

Description

jpeg_default_colorspace() shall set the JPEG color space to a default value based on the color space of the input image.

jpeg_destroy

Name

jpeg_destroy — deallocation of a JPEG object

Synopsis

```
#include <jpeglib.h>
void jpeg_destroy(j_common_ptr cinfo);
```

Description

The function jpeg_destroy() shall deallocate and release all memory associated with the JPEG object. The object itself and the error manager struct allocated by the application should not be freed by this function.

jpeg_destroy_compress

Name

jpeg_destroy_compress — deallocate a JPEG compression object

Synopsis

```
#include <jpeglib.h>
void jpeg_destroy_compress(j_compress_ptr cinfo);
```

Description

The function jpeg_destroy_compress() shall deallocate and release all memory associated with the compression object. The jpeg_compress_struct itself and the error manager struct allocated by the application should not be freed by this function.

jpeg_destroy_decompress

Name

jpeg_destroy_decompress — deallocate a JPEG decompression object

Synopsis

```
#include <jpeglib.h>
void jpeg_destroy_decompress(j_decompress_ptr cinfo);
```

Description

The function jpeg_destroy_decompress() shall deallocate and release all memory associated with the decompression object. The jpeg_decompress_struct itself and the error manager struct allocated by the application should not be freed by this function.

jpeg_finish_compress

Name

jpeg_finish_compress — complete the compression cycle

Synopsis

```
#include <jpeglib.h>
void jpeg_finish_compress(j_compress_ptr cinfo);
```

Description

The function jpeg_finish_compress() shall set the compression state to completion. Any remaining bufferload of data shall be written to the data destination. jpeg_finish_compress() shall also release working memory associated with the JPEG object.

Errors

jpeg_finish_compress() shall return error if the function is invoked before writing the specified number of scanlines.

jpeg_finish_decompress

Name

jpeg_finish_decompress — complete decompression cycle

Synopsis

```
#include <jpeglib.h>
boolean jpeg_finish_decompress(j_decompress_ptr cinfo);
```

Description

The function jpeg_finish_decompress() shall set the decompression state to completion. Any remaining input data shall be decoded and consumed. jpeg_finish_decompress() shall also release working memory associated with the JPEG object.

Return Value

jpeg_finish_decompress() shall return FALSE if data source requests suspension of the decompressor. Otherwise, jpeg_finish_decompress() shall return TRUE.

Errors

jpeg_finish_decompress() shall return error if the function is invoked before reading the specified number of scanlines.

jpeg_finish_output

Name

jpeg_finish_output — complete pass in buffered-image mode

Synopsis

```
#include <jpeglib.h>
boolean jpeg_finish_output(j_decompress_ptr cinfo);
```

Description

jpeg_start_output() shall change state for completion of an output pass in buffered-image mode. The scan number shall be incremented if the end-of-image marker is not reached.

Return Value

jpeg_finish_output() shall return FALSE if decompression is suspended. Otherwise, jpeg_finish_output() shall return TRUE.

jpeg_has_multiple_scans**Name**

jpeg_has_multiple_scans — test if input image has multiple scans

Synopsis

```
#include <jpeglib.h>
boolean jpeg_has_multiple_scans(j_decompress_ptr cinfo);
```

Description

jpeg_has_multiple_scans() shall return TRUE if the incoming image file has more than one scan.

Return Value

TRUE for multiple scans, FALSE otherwise.

jpeg_input_complete**Name**

jpeg_input_complete — test for end-of-image

Synopsis

```
#include <jpeglib.h>
boolean jpeg_input_complete(j_decompress_ptr cinfo);
```

Description

jpeg_input_complete() shall return TRUE if the end-of-image marker has not been read for the input image file.

Return Value

TRUE if reached EOF, FALSE otherwise.

jpeg_new_colormap**Name**

jpeg_new_colormap — switch to a new external colormap between output passes

Synopsis

```
#include <jpeglib.h>
void jpeg_new_colormap(j_decompress_ptr cinfo);
```

Description

jpeg_new_colormap() shall select 2-pass quantizer for using an external colormap and notify the quantizer of the switch to a colormap that is different from the one used in the previous pass. The new colormap must be set using cinfo.colormap before a call to jpeg_new_colormap().

jpeg_quality_scaling

Name

`jpeg_quality_scaling` — convert quality scaling to percentage scaling factor

Synopsis

```
#include <jpeglib.h>
int jpeg_quality_scaling(int quality);
```

Description

The function `jpeg_quality_scaling()` shall convert a user-specified quality rating to a percentage scaling factor for an underlying quantization table, using JIG-recommended scaling curve. The input "quality" factor ranges from 0 to 100.

Return Value

`jpeg_quality_scaling()` shall return the integer value of quality as a percentage.

jpeg_read_coefficients

Name

`jpeg_read_coefficients` — read the contents of JPEG file as DCT coefficients

Synopsis

```
#include <jpeglib.h>
jvirt_barray_ptr * jpeg_read_coefficients(j_decompress_ptr cinfo);
```

Description

`jpeg_read_coefficients()` shall return the contents of the JPEG image file as DCT coefficients. The entire image shall be read as a set of virtual coefficient-block arrays, one array per component. The return value is a pointer to an array of virtual-array descriptors.

Each block in the block arrays shall contain quantized coefficient values in normal array order and not JPEG zigzag order. The block arrays shall contain only DCT blocks containing real data; any entirely-dummy blocks added to fill out interleaved MCUs at the right or bottom edges of the image shall be discarded during reading and shall not be stored in the block arrays. The size of each block array can be determined from the `width_in_blocks` and `height_in_blocks` fields of the component's `comp_info` entry.

Return Value

`jpeg_read_coefficients()` shall return a pointer to an array of virtual-array descriptors containing DCT coefficients.

jpeg_read_header

Name

jpeg_read_header — read start of JPEG datastream

Synopsis

```
#include <jpeglib.h>
int jpeg_read_header(j_decompress_ptr cinfo, boolean require_image);
```

Description

The function jpeg_read_header() shall read the JPEG datastream until the first SOS marker is encountered. The function shall initialize all decompression parameters to default values and save all tables and parameters in the decompression object structure.

Return Value

jpeg_read_header() shall return with one of the following return codes:

JPEG_HEADER_OK

if SOS marker is reached

JPEG_HEADER_TABLES_ONLY

for an abbreviated input image, if EOI is reached

JPEG_SUSPENDED

if data source module requests suspension of the decompressor.

Errors

jpeg_read_header() returns error if it encounters end-of-image and require_image is TRUE.

jpeg_read_raw_data

Name

jpeg_read_raw_data — read raw downsampled image data

Synopsis

```
#include <jpeglib.h>
JDIMENSION jpeg_read_raw_data(j_decompress_ptr cinfo, JSAMPIMAGE data,
JDIMENSION max_lines);
```

Description

jpeg_read_raw_data() shall return upto max_lines number of scanlines of raw downsampled data into the JSAMPIMAGE array argument. The data buffer must be of atleast max_v_samp_factor*DCTSIZE scanlines as jpeg_read_raw_data() shall return one MCU row per call.

The passed max_lines value must be atleast (cinfo->max_v_samp_factor * DCTSIZE). Before starting the decompression sequence, cinfo->raw_data_out must be set to TRUE.

Return Value

jpeg_read_raw_data() shall return the number of scanlines processed. jpeg_read_raw_data() shall return 0 if the data source is suspended.

Errors

jpeg_read_raw_data() shall report error and exit if max_lines is less than (cinfo->max_v_samp_factor*DCTSIZE).

jpeg_read_scanlines

Name

jpeg_read_scanlines — read scanlines of data from JPEG decompressor

Synopsis

```
#include <jpeglib.h>
JDIMENSION jpeg_read_scanlines(j_decompress_ptr cinfo, JSAMPARRAY scanlines, JDIMENSION max_lines);
```

Description

jpeg_read_scanlines() shall return upto the maximum number of scanlines of decompressed image data. This may be less than the number requested in cases such as bottom of image, data source suspension, and operating modes that emit multiple scanlines at a time. Image data shall be returned in top-to-bottom scanline order.

Return Value

On success, jpeg_read_scanlines() shall return the number of lines actually read.

Errors

jpeg_read_scanlines() shall report a non-fatal error if the requested number of scanlines is greater than the height of the original unscaled image.

jpeg_resync_to_restart**Name**

`jpeg_resync_to_restart` — resync if marker other than expected restart marker found

Synopsis

```
#include <jpeglib.h>
boolean jpeg_resync_to_restart(j_decompress_ptr cinfo, int desired);
```

Description

`jpeg_resync_to_restart()` is the default resync procedure that a data source manager can invoke when the decompressor fails to find a restart (RSTn) marker where one is expected. The function shall find a suitable point for resuming decompression and position the input stream to the next data segment. The desired restart marker number (0..7) is passed as argument "desired".

Return Value

`jpeg_resync_to_restart()` shall return FALSE if decompression is suspended. Otherwise, `jpeg_start_output()` shall return TRUE.

jpeg_save_markers**Name**

`jpeg_save_markers` — save the contents of special markers

Synopsis

```
#include <jpeglib.h>
void jpeg_save_markers(j_decompress_ptr cinfo, int marker_code,
unsigned int length_limit);
```

Description

`jpeg_save_markers()` shall obtain `length_limit` bytes of special markers of type specified by `marker_code`. The `marker_code` shall take one of the following values: JPEG_COM, JPEG_APP0 - JPEG_APP15. The marker contents shall be stored in the marker field of `cinfo`. If `length_limit` is 0xFFFF, all marker bytes are saved.

jpeg_set_colorspace

Name

jpeg_set_colorspace — set JPEG file's colorspace

Synopsis

```
#include <jpeglib.h>
void jpeg_set_colorspace(j_compress_ptr cinfo, J_COLOR_SPACE
colorspace);
```

Description

jpeg_set_colorspace() shall set the JPEG file's colorspace to the given color space. The function may change the value of other compression parameters that are dependent on color space.

Errors

jpeg_set_colorspace() shall return error and exit if the state of the compression object indicates that the compression cycle has started.

jpeg_set_defaults

Name

jpeg_set_defaults — set compression parameters to default values

Synopsis

```
#include <jpeglib.h>
void jpeg_set_defaults(j_compress_ptr cinfo);
```

Description

The function jpeg_set_defaults() shall set all JPEG compression parameters to default values using the input image's color space.

Errors

jpeg_set_defaults() shall return error and exit if the state of the compression object indicates that the compression cycle has started.

jpeg_set_linear_quality**Name**

`jpeg_set_linear_quality` — construct JPEG quantization tables for indicated scale factor

Synopsis

```
#include <jpeglib.h>
void jpeg_set_linear_quality(j_compress_ptr cinfo, int scale_factor,
boolean force_baseline);
```

Description

The function `jpeg_set_linear_quality()` shall construct JPEG quantization tables from the sample table values in JPEG specification Section K.1, multiplied by the given percentage scale factor. If "force_baseline" is TRUE, the computed quantization table entries are limited to 1..255 for JPEG baseline compatibility.

jpeg_set_marker_processor**Name**

`jpeg_set_marker_processor` — set processor method for COM or APPn markers

Synopsis

```
#include <jpeglib.h>
void jpeg_set_marker_processor(j_decompress_ptr cinfo, int marker_code,
jpeg_marker_parser_method routine);
```

Description

`jpeg_set_marker_processor()` shall set a marker processor routine having signature `boolean jpeg_marker_parser_method() j_decompress_ptr cinfo`. The marker processor routine must return TRUE if the marker and its parameters are processed completely. The routine must return FALSE if it is forced to suspend before reaching the end of marker parameters.

jpeg_set_quality**Name**

`jpeg_set_quality` — construct JPEG quantization tables for indicated quality

Synopsis

```
#include <jpeglib.h>
void jpeg_set_quality(j_compress_ptr cinfo, int quality, boolean
force_baseline);
```

Description

The function `jpeg_set_quality()` shall construct JPEG quantization tables for the given quality setting. The quality value ranges from 0..100. If "force_baseline" is TRUE, the computed quantization table entries are limited to 1..255 for JPEG baseline compatibility.

jpeg_simple_progression

Name

jpeg_simple_progression — generate scan script for writing progressive JPEG file

Synopsis

```
#include <jpeglib.h>
void jpeg_simple_progression(j_compress_ptr cinfo);
```

Description

The function jpeg_simple_progression() shall generate a default scan script for writing a progressive JPEG file.

jpeg_start_compress

Name

jpeg_start_compress — initialize a compression cycle

Synopsis

```
#include <jpeglib.h>
void jpeg_start_compress(j_compress_ptr cinfo, boolean
write_all_tables);
```

Description

The function jpeg_start_compress() shall initialize state for a JPEG compression cycle. The compression parameters, data destination and source image information must be set prior to the invocation of jpeg_start_compress(). Setting write_all_tables to TRUE shall indicate that a complete JPEG interchange datastream will be written and all Huffman tables shall be emitted. If write_all_tables is set to FALSE, the default behavior shall be to emit a pure abbreviated image with no tables.

jpeg_start_decompress

Name

jpeg_start_decompress — initialize a decompression cycle

Synopsis

```
#include <jpeglib.h>
boolean jpeg_start_decompress(j_decompress_ptr cinfo);
```

Description

The function jpeg_start_decompress() shall initialize state for a JPEG decompression cycle and allocate working memory. The JPEG datastream header must be read prior to the invocation of jpeg_start_decompress() to obtain the parameters for decompression.

jpeg_start_output**Name**

jpeg_start_output — initialize for a pass in buffered-image mode

Synopsis

```
#include <jpeglib.h>
boolean jpeg_start_output(j_decompress_ptr cinfo, int scan_number);
```

Description

jpeg_start_output() shall initialize state for an output pass in buffered-image mode. The scan_number indicates the scan of the image to be displayed. Scan numbers start from 1. The buffered-image mode must be selected before starting an output pass by setting cinfo.buffered_image = TRUE.

Return Value

jpeg_start_output() shall return FALSE if decompression is suspended. Otherwise, jpeg_start_output() shall return TRUE.

jpeg_std_error**Name**

jpeg_std_error — update error manager with error handling routines

Synopsis

```
#include <jpeglib.h>
struct jpeg_error_mgr * jpeg_std_error(struct jpeg_error_mgr *
error_mgr);
```

Description

jpeg_std_error() shall update the given jpeg_error_mgr object with standard error handling methods.

Return Value

jpeg_std_error() shall return the updated jpeg_error_mgr object.

jpeg_stdio_dest**Name**

jpeg_stdio_dest — Initialize state for output to stdio stream

Synopsis

```
#include <jpeglib.h>
void jpeg_stdio_dest(j_compress_ptr cinfo, FILE * outfile);
```

Description

jpeg_stdio_dest() shall initialize state for output to the given stdio stream outfile. The stream outfile must be in open state.

jpeg_stdio_src

Name

jpeg_stdio_src — Initialize state for input from stdio stream

Synopsis

```
#include <jpeglib.h>
void jpeg_stdio_src(j_decompress_ptr cinfo, FILE * infile);
```

Description

jpeg_stdio_src() shall initialize state for input from the given stdio stream infile. The stream infile must be in open state.

jpeg_suppress_tables

Name

jpeg_suppress_tables — control quantization and Huffman table suppression

Synopsis

```
#include <jpeglib.h>
void jpeg_suppress_tables(j_compress_ptr cinfo, boolean suppress);
```

Description

The function jpeg_suppress_tables() forcibly suppress or un-suppresses all quantization and Huffman tables. If "suppress" is TRUE, the function shall mark all currently defined tables as already written. Otherwise, the tables will be marked as not written. This controls the emission of tables on subsequent calls to jpeg_start_compress().

jpeg_write_coefficients

Name

jpeg_write_coefficients — Compression initialization for writing raw-coefficient data

Synopsis

```
#include <jpeglib.h>
void jpeg_write_coefficients(j_compress_ptr cinfo, jvirt_barray_ptr *
coeff_arrays);
```

Description

jpeg_write_coefficients() shall write the contents of the JPEG file as DCT coefficients. The DCT coefficients shall be obtained from the array of virtual block array descriptors pointed to be "coeff_arrays". The number of arrays to be written is obtained from cinfo.num_components. Note: The virtual arrays may not be realized before jpeg_write_coefficients() is called. The virtual arrays requested from the compression object's memory manager may be realized by calling jpeg_write_coefficients() as a side-effect.

jpeg_write_m_byte**Name**

`jpeg_write_m_byte` — emit a byte of special marker data

Synopsis

```
#include <jpeglib.h>
void jpeg_write_m_byte(j_compress_ptr cinfo, int val);
```

Description

`jpeg_write_m_byte()` shall emit a byte of marker data equal to "val".

jpeg_write_m_header**Name**

`jpeg_write_m_header` — emit header for special marker

Synopsis

```
#include <jpeglib.h>
void jpeg_write_m_header(j_compress_ptr cinfo, int marker, unsigned
int datalen);
```

Description

`jpeg_write_m_header()` shall emit the marker header. The marker header consists of the marker code derived from "marker" followed by the length of marker data "datalen".

jpeg_write_marker**Name**

`jpeg_write_marker` — write special marker data

Synopsis

```
#include <jpeglib.h>
void jpeg_write_marker(j_compress_ptr cinfo, int marker, const JOCTET
* dataptr, unsigned int datalen);
```

Description

`jpeg_write_marker()` shall emit the marker header and "datalen" number of marker bytes obtained from "*dataptr". The marker header consists of the marker code followed by the length of marker data.

jpeg_write_raw_data

Name

jpeg_write_raw_data — write raw downsampled image data

Synopsis

```
#include <jpeglib.h>
JDIMENSION jpeg_write_raw_data(j_compress_ptr cinfo, JSAMPIMAGE data,
JDIMENSION num_lines);
```

Description

jpeg_write_raw_data() shall write the compressed image data for raw downsampled data supplied in the JSAMPIMAGE array argument. The number of scanlines passed in num_lines to jpeg_write_raw_data() is measured in terms of the component with the largest v_samp_factor.

If the image dimensions are not a multiple of the MCU size, the data must be padded to a multiple of a DCT block in each component, such that each downsampled row must contain a multiple of 8 valid samples, and there must be a multiple of 8 sample rows for each component. Data must be padded so that the passed num_lines value is atleast (cinfo->max_v_samp_factor * DCTSIZE). jpeg_write_raw_data() shall process one MCU row per call which is (cinfo->comp_info[0].v_samp_factor*DCTSIZE) sample rows of each component.

Before starting the compression sequence, cinfo->raw_data_in must be set to TRUE. cinfo->jpeg_color_space must be set to colorspace to be used in the JPEG file. The sampling factors, cinfo->comp_info[i].h_samp_factor and cinfo->comp_info[i].v_samp_factor must be set to the dimensions of the supplied data.

Return Value

jpeg_write_raw_data() shall return the number of scanlines processed. jpeg_write_raw_data() shall return 0 if the compressor is suspended.

Errors

jpeg_write_raw_data() shall report error and exit if num_lines is less than (cinfo->max_v_samp_factor*DCTSIZE).

jpeg_write_scanlines**Name**

`jpeg_write_scanlines` — write scanlines of data to JPEG compressor

Synopsis

```
#include <jpeglib.h>
JDIMENSION jpeg_write_scanlines(j_compress_ptr cinfo, JSAMPARRAY
scanlines, JDIMENSION num_lines);
```

Description

`jpeg_write_scanlines()` shall write the number of scanlines of image data to be compressed from in-memory buffers. The image data shall be written in top-to-bottom scanline order.

Return Value

On success, `jpeg_read_scanlines()` shall return the number of lines actually written.

Errors

`jpeg_write_scanlines()` shall report a non-fatal error if the requested number of scanlines is greater than the height of the original unscaled image.

jpeg_write_tables**Name**

`jpeg_write_tables` — create a tables-only file

Synopsis

```
#include <jpeglib.h>
void jpeg_write_tables(j_compress_ptr cinfo);
```

Description

`jpeg_write_tables()` shall create a tables-only file by writing an abbreviated datastream containing only SOI, DQT and/or DHT and EOI markers. All the quantization and Huffman tables currently defined in the compression object shall be emitted unless their `sent_table` flag is set to TRUE. Once emitted, `sent_table` flag shall be set for all tables written.

VI Fontconfig library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

10 Libraries

10.1 Interfaces for libfontconfig

Table 10-1 defines the library name and shared object name for the libfontconfig library

Table 10-1 libfontconfig Definition

Library:	libfontconfig
SONAME:	libfontconfig.so.1

The behavior of the interfaces in this library is specified by the following specifications:

[fontconfig] Fontconfig Developers Reference

10.1.1 Font Customization and Configuration Library

10.1.1.1 Interfaces for Font Customization and Configuration Library

An LSB conforming implementation shall provide the generic functions for Font Customization and Configuration Library specified in Table 10-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 10-2 libfontconfig - Font Customization and Configuration Library Function Interfaces

FcAtomicCreate [fontconfig]	FcAtomicDeleteNew [fontconfig]	FcAtomicDestroy [fontconfig]
FcAtomicLock [fontconfig]	FcAtomicNewFile [fontconfig]	FcAtomicOrigFile [fontconfig]
FcAtomicReplaceOrig [fontconfig]	FcAtomicUnlock [fontconfig]	FcBlanksAdd [fontconfig]
FcBlanksCreate [fontconfig]	FcBlanksDestroy [fontconfig]	FcBlanksIsMember [fontconfig]
FcCharSetAddChar [fontconfig]	FcCharSetCopy [fontconfig]	FcCharSetCount [fontconfig]
FcCharSetCreate [fontconfig]	FcCharSetDestroy [fontconfig]	FcCharSetEqual [fontconfig]
FcCharSetFirstPage [fontconfig]	FcCharSetHasChar [fontconfig]	FcCharSetIntersect [fontconfig]
FcCharSetIntersectCount [fontconfig]	FcCharSetIsSubset [fontconfig]	FcCharSetNextPage [fontconfig]
FcCharSetSubtract [fontconfig]	FcCharSetSubtractCount [fontconfig]	FcCharSetUnion [fontconfig]
FcConfigAppFontAddDir [fontconfig]	FcConfigAppFontAddFile [fontconfig]	FcConfigAppFontClear [fontconfig]

FcConfigBuildFonts [fontconfig]	FcConfigDestroy [fontconfig]	FcConfigEnableHome [fontconfig]
FcConfigFilename [fontconfig]	FcConfigGetBlanks [fontconfig]	FcConfigGetCache [fontconfig]
FcConfigGetConfigDirs [fontconfig]	FcConfigGetConfigFiles [fontconfig]	FcConfigGetCurrent [fontconfig]
FcConfigGetFontDirs [fontconfig]	FcConfigGetFonts [fontconfig]	FcConfigGetRescanInterval [fontconfig]
FcConfigGetRescanInterval [fontconfig]	FcConfigHome [fontconfig]	FcConfigParseAndLoad [fontconfig]
FcConfigSetCurrent [fontconfig]	FcConfigSetRescanInterval [fontconfig]	FcConfigSetRescanInterval [fontconfig]
FcConfigSubstitute [fontconfig]	FcConfigSubstituteWithPat [fontconfig]	FcConfigUptoDate [fontconfig]
FcDefaultSubstitute [fontconfig]	FcDirCacheValid [fontconfig]	FcDirSave [fontconfig]
FcDirScan [fontconfig]	FcFileScan [fontconfig]	FcFini [fontconfig]
FcFontList [fontconfig]	FcFontMatch [fontconfig]	FcFontRenderPrepare [fontconfig]
FcFontSetAdd [fontconfig]	FcFontSetCreate [fontconfig]	FcFontSetDestroy [fontconfig]
FcFontSetList [fontconfig]	FcFontSetMatch [fontconfig]	FcFontSetPrint [fontconfig]
FcFontSetSort [fontconfig]	FcFontSetSortDestroy [fontconfig]	FcFontSort [fontconfig]
FcFreeTypeCharIndex [fontconfig]	FcFreeTypeCharSet [fontconfig]	FcFreeTypeCharSetAndSpacing [fontconfig]
FcFreeTypeQuery [fontconfig]	FcGetVersion [fontconfig]	FcInit [fontconfig]
FcInitBringUptoDate [fontconfig]	FcInitLoadConfig [fontconfig]	FcInitLoadConfigAndFonts [fontconfig]
FcInitReinitialize [fontconfig]	FcLangSetAdd [fontconfig]	FcLangSetCompare [fontconfig]
FcLangSetContains [fontconfig]	FcLangSetCopy [fontconfig]	FcLangSetCreate [fontconfig]
FcLangSetDestroy [fontconfig]	FcLangSetEqual [fontconfig]	FcLangSetHasLang [fontconfig]
FcLangSetHash [fontconfig]	FcMatrixCopy [fontconfig]	FcMatrixEqual [fontconfig]
FcMatrixMultiply [fontconfig]	FcMatrixRotate [fontconfig]	FcMatrixScale [fontconfig]

FcMatrixShear [fontconfig]	FcNameConstant [fontconfig]	FcNameGetConstant [fontconfig]
FcNameGetObjectType [fontconfig]	FcNameParse [fontconfig]	FcNameRegisterConstants [fontconfig]
FcNameRegisterObject Types [fontconfig]	FcNameUnparse [fontconfig]	FcNameUnregisterConstants [fontconfig]
FcNameUnregisterObject Types [fontconfig]	FcObjectSetAdd [fontconfig]	FcObjectSetBuild [fontconfig]
FcObjectSetCreate [fontconfig]	FcObjectSetDestroy [fontconfig]	FcObjectSetVaBuild [fontconfig]
FcPatternAdd [fontconfig]	FcPatternAddBool [fontconfig]	FcPatternAddCharSet [fontconfig]
FcPatternAddDouble [fontconfig]	FcPatternAddFTFace [fontconfig]	FcPatternAddInteger [fontconfig]
FcPatternAddLangSet [fontconfig]	FcPatternAddMatrix [fontconfig]	FcPatternAddString [fontconfig]
FcPatternAddWeak [fontconfig]	FcPatternBuild [fontconfig]	FcPatternCreate [fontconfig]
FcPatternDel [fontconfig]	FcPatternDestroy [fontconfig]	FcPatternDuplicate [fontconfig]
FcPatternEqual [fontconfig]	FcPatternEqualSubset [fontconfig]	FcPatternGet [fontconfig]
FcPatternGetBool [fontconfig]	FcPatternGetCharSet [fontconfig]	FcPatternGetDouble [fontconfig]
FcPatternGetFTFace [fontconfig]	FcPatternGetInteger [fontconfig]	FcPatternGetLangSet [fontconfig]
FcPatternGetMatrix [fontconfig]	FcPatternGetString [fontconfig]	FcPatternHash [fontconfig]
FcPatternPrint [fontconfig]	FcPatternReference [fontconfig]	FcPatternVaBuild [fontconfig]
FcStrBasename [fontconfig]	FcStrCmp [fontconfig]	FcStrCmpIgnoreCase [fontconfig]
FcStrCopy [fontconfig]	FcStrCopyFilename [fontconfig]	FcStrDirname [fontconfig]
FcStrListCreate [fontconfig]	FcStrListDone [fontconfig]	FcStrListNext [fontconfig]
FcStrSetAdd [fontconfig]	FcStrSetAddFilename [fontconfig]	FcStrSetCreate [fontconfig]
FcStrSetDel [fontconfig]	FcStrSetDestroy [fontconfig]	FcStrSetEqual [fontconfig]
FcStrSetMember [fontconfig]	FcUcs4ToUtf8 [fontconfig]	FcUtf16Len [fontconfig]

FcUtf16ToUcs4 [fontconfig]	FcUtf8Len [fontconfig]	FcUtf8ToUcs4 [fontconfig]
FcValueDestroy [fontconfig]	FcValueEqual [fontconfig]	FcValuePrint [fontconfig]
FcValueSave [fontconfig]		

An LSB conforming implementation shall provide the generic deprecated functions for Font Customization and Configuration Library specified in Table 10-3, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 10-3 libfontconfig - Font Customization and Configuration Library Depreciated Function Interfaces

FcConfigGetCache [fontconfig]	FcConfigGetRescanInve rval [fontconfig]	FcConfigSetRescanInve rval [fontconfig]
FcDirSave [fontconfig]	FcDirScan [fontconfig]	FcFontSetSortDestroy [fontconfig]

10.2 Data Definitions for libfontconfig

This section defines global identifiers and their values that are associated with interfaces contained in libfontconfig. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

10.2.1 fontconfig/fcfreeType.h

```
extern FT_UInt FcFreeTypeCharIndex(FT_Face, FcChar32);
extern FcCharSet *FcFreeTypeCharSet(FT_Face, FcBlanks *);
extern FcCharSet *FcFreeTypeCharSetAndSpacing(FT_Face, FcBlanks *,
int *);
extern FcBool FcPatternAddFTFace(FcPattern *, const char *, const
FT_Face);
extern FcResult FcPatternGetFTFace(const FcPattern *, const char *,
int,
FT_Face *);
```

10.2.2 fontconfig/fcprivate.h

```

#define FcObjectSetVapBuild(__ret__, __first__, __va__) \
    { FcObjectSet *__os__; const char *__ob__; __ret__ = 0; \
    __os__ = \
        FcObjectSetCreate (); if (!__os__) goto \
        _FcObjectSetVapBuild_bail0; \
    __ob__ = __first__; while (__ob__) { if (!FcObjectSetAdd \
    (__os__, \
    __ob__) goto _FcObjectSetVapBuild_bail1; __ob__ = va_arg \
    (__va__, \
    const char *); } __ret__ = __os__; \
    _FcObjectSetVapBuild_bail1: if \
    (!__ret__ && __os__) FcObjectSetDestroy (__os__); \
    _FcObjectSetVapBuild_bail0: ; }
#define FcPatternVapBuild(result, orig, va) { \
    FcPattern *__p__ = (orig); \
    const char *__o__; \
    FcValue __v__; \
    if (!__p__) \
    { \
        __p__ = FcPatternCreate (); \
        if (!__p__) \
            goto _FcPatternVapBuild_bail0; \
    } \
    for (;;) \
    { \
        __o__ = va_arg (va, const char *); \
        if (!__o__) \
            break; \
        __v__.type = va_arg (va, FcType); \
        switch (__v__.type) { \
        case FcTypeVoid: \
            goto _FcPatternVapBuild_bail1; \
        case FcTypeInteger: \
            __v__.u.i = va_arg (va, int); \
            break; \
        case FcTypeDouble: \
            __v__.u.d = va_arg (va, double); \
            break; \
        case FcTypeString: \
            __v__.u.s = va_arg (va, const FcChar8 *); \
            break; \
        case FcTypeBool: \
            __v__.u.b = va_arg (va, FcBool); \
            break; \
        case FcTypeMatrix: \
            __v__.u.m = va_arg (va, const FcMatrix *); \
            break; \
        case FcTypeCharSet: \
            __v__.u.c = va_arg (va, const FcCharSet *); \
            break; \
        case FcTypeFTFace: \
            __v__.u.f = va_arg (va, FT_Face); \
            break; \
        case FcTypeLangSet: \
            __v__.u.l = va_arg (va, const FcLangSet *); \
            break; \
        } \
        if (!FcPatternAdd (__p__, __o__, __v__, FcTrue)) \
            goto _FcPatternVapBuild_bail1; \
    } \
    result = __p__; \
    goto _FcPatternVapBuild_return; \
}

```

```

_FcPatternVapBuild_bail1:\
    if (!orig)\
        FcPatternDestroy (__p__);\
_FcPatternVapBuild_bail0:\
    result = (void*)0;\
\
_FcPatternVapBuild_return:\
    ;\
}

```

10.2.3 fontconfig/fontconfig.h

```

#define _FCFUNCPROTOBEGIN
#define _FCFUNCPROTOEND
#define FC_USER_CACHE_FILE ".fonts.cache-FC_CACHE_VERSION"
#define FC_CACHE_VERSION "1"
#define FC_DIR_CACHE_FILE "fonts.cache-FC_CACHE_VERSION"
#define FcIsUpper(c) ((0101 <= (c) && (c) <= 0132))
#define FcIsLower(c) ((0141 <= (c) && (c) <= 0172))
#define FC_CHARSET_DONE ((FcChar32) -1)
#define FC_VERSION ((FC_MAJOR * 10000) + (FC_MINOR * 100) + (FC_REVISION))
#define FcMatrixInit(m) ((m)->xx = (m)->yy = 1, (m)->xy = (m)->yx = 0)
#define FC_CHARSET_MAP_SIZE (256/32)
#define FcToLower(c) (FcIsUpper(c) ? (c) - 0101 + 0141 : (c))
#define FC_HINT_NONE 0
#define FC_PROPORTIONAL 0
#define FC_RGBA_UNKNOWN 0
#define FC_SLANT_ROMAN 0
#define FC_WEIGHT_THIN 0
#define FcFalse 0
#define FC_HINT_SLIGHT 1
#define FC_RGBA_RGB 1
#define FcTrue 1
#define FC_MONO 100
#define FC_SLANT_ITALIC 100
#define FC_WEIGHT_MEDIUM 100
#define FC_WIDTH_NORMAL 100
#define FC_CHARCELL 110
#define FC_SLANT_OBLIQUE 110
#define FC_WIDTH_SEMIEXPANDED 113
#define FC_WIDTH_EXPANDED 125
#define FC_WIDTH_EXTRAEXPANDED 150
#define FC_WEIGHT_DEMIBOLD 180
#define FC_HINT_MEDIUM 2
#define FC_MAJOR 2
#define FC_REVISION 2
#define FC_RGBA_BGR 2
#define FC_WEIGHT_BOLD 200
#define FC_WIDTH_ULTRAEXPANDED 200
#define FC_WEIGHT_EXTRABOLD 205
#define FC_WEIGHT_BLACK 210
#define FC_HINT_FULL 3
#define FC_MINOR 3
#define FC_RGBA_VRGB 3
#define FC_RGBA_VBGR 4
#define FC_WEIGHT_EXTRALIGHT 40
#define FC_RGBA_NONE 5
#define FC_WEIGHT_LIGHT 50
#define FC_WIDTH_ULTRACONDENSED 50
#define FC_UTF8_MAX_LEN 6
#define FC_WIDTH_EXTRACONDENSED 63
#define FC_WEIGHT_BOOK 75
#define FC_WIDTH_CONDENSED 75

```

```

#define FC_WEIGHT_REGULAR      80
#define FC_WIDTH_SEMICONDENSED 87
#define FC_DUAL 90
#define FC_ANTIALIAS           "antialias"
#define FC_ASPECT              "aspect"
#define FC_AUTOHINT            "autohint"
#define FC_CAPABILITY          "capability"
#define FC_CHAR_HEIGHT         "charheight"
#define FC_CHARSET             "charset"
#define FC_CHAR_WIDTH          "charwidth"
#define FC_DPI                 "dpi"
#define FC_EMBOLDEN            "embolden"
#define FC_FAMILY              "family"
#define FC_FAMILYLANG          "familylang"
#define FC_WEIGHT_HEAVY FC_WEIGHT_BLACK
#define FC_WEIGHT_SEMIBOLD     FC_WEIGHT_DEMIBOLD
#define FC_WEIGHT_ULTRABOLD    FC_WEIGHT_EXTRABOLD
#define FC_WEIGHT_ULTRALIGHT   FC_WEIGHT_EXTRALIGHT
#define FC_WEIGHT_NORMAL       FC_WEIGHT_REGULAR
#define FC_FILE "file"
#define FC_FONTFORMAT          "fontformat"
#define FC_FONTVERSION         "fontversion"
#define FC_FOUNDRY             "foundry"
#define FC_FT_FACE             "ftface"
#define FC_FULLNAME            "fullname"
#define FC_FULLNAMELANG        "fullnamelang"
#define FC_GLOBAL_ADVANCE      "globaladvance"
#define FC_HINTING             "hinting"
#define FC_HINT_STYLE          "hintstyle"
#define FC_INDEX               "index"
#define FC_LANG "lang"
#define FC_MATRIX              "matrix"
#define FC_MINSPEACE           "minspace"
#define FC_OUTLINE             "outline"
#define FC_PIXEL_SIZE          "pixelsize"
#define FC_RASTERIZER          "rasterizer"
#define FC_RGBA "rgba"
#define FC_SCALABLE             "scalable"
#define FC_SCALE               "scale"
#define FC_SIZE "size"
#define FC_SLANT                "slant"
#define FC_SOURCE              "source"
#define FC_SPACING              "spacing"
#define FC_STYLE               "style"
#define FC_STYLELANG           "stylelang"
#define FC_VERTICAL_LAYOUT     "verticallayout"
#define FC_WEIGHT              "weight"
#define FC_WIDTH               "width"

typedef int FcBool;
typedef struct _FcPattern FcPattern;
typedef enum _FcType {
    FcTypeVoid = 0,
    FcTypeInteger = 1,
    FcTypeDouble = 2,
    FcTypeString = 3,
    FcTypeBool = 4,
    FcTypeMatrix = 5,
    FcTypeCharSet = 6,
    FcTypeFTFace = 7,
    FcTypeLangSet = 8
} FcType;
typedef unsigned char FcChar8;
typedef struct _FcMatrix {
    double xx;
    double xy;

```

```

    double yx;
    double yy;
} FcMatrix;
typedef struct _FcCharSet FcCharSet;
typedef struct _FcLangSet FcLangSet;
typedef struct _FcValue {
    FcType type;
    union {
        const FcChar8 *s;
        int i;
        FcBool b;
        double d;
        const FcMatrix *m;
        const FcCharSet *c;
        void *f;
        const FcPattern *p;
        const FcLangSet *l;
    } u;
} FcValue;
typedef struct _FcStrList FcStrList;
typedef struct _FcConfig FcConfig;
typedef struct _FcStrSet FcStrSet;
typedef unsigned int FcChar32;
typedef enum _FcMatchKind {
    FcMatchPattern = 0,
    FcMatchFont = 1
} FcMatchKind;
typedef struct _FcObjectSet {
    int nobject;
    int sobject;
    const char **objects;
} FcObjectSet;
typedef enum {
    FcEndianBig = 0,
    FcEndianLittle = 1
} FcEndian;
typedef struct _FcAtomic FcAtomic;
typedef struct _FcConstant {
    const FcChar8 *name;
    const char *object;
    int value;
} FcConstant;
typedef struct _FcFontSet {
    int nfont;
    int sfont;
    FcPattern **fonts;
} FcFontSet;
typedef enum _FcSetName {
    FcSetSystem = 0,
    FcSetApplication = 1
} FcSetName;
typedef struct _FcBlanks FcBlanks;
typedef enum _FcResult {
    FcResultMatch = 0,
    FcResultNoMatch = 1,
    FcResultTypeMismatch = 2,
    FcResultNoId = 3,
    FcResultOutOfMemory = 4
} FcResult;
typedef struct _FcObjectType {
    const char *object;
    FcType type;
} FcObjectType;
typedef struct _FcGlobalCache FcGlobalCache;
typedef enum _FcLangResult {
    FcLangEqual = 0,

```

```

    FcLangDifferentCountry = 1,
    FcLangDifferentLang = 2
} FcLangResult;
typedef short unsigned int FcChar16;
extern FcAtomic *FcAtomicCreate(const FcChar8 * file);
extern void FcAtomicDeleteNew(FcAtomic * atomic);
extern void FcAtomicDestroy(FcAtomic * atomic);
extern FcBool FcAtomicLock(FcAtomic * atomic);
extern FcChar8 *FcAtomicNewFile(FcAtomic * atomic);
extern FcChar8 *FcAtomicOrigFile(FcAtomic * atomic);
extern FcBool FcAtomicReplaceOrig(FcAtomic * atomic);
extern void FcAtomicUnlock(FcAtomic * atomic);
extern FcBool FcBlanksAdd(FcBlanks * b, FcChar32 ucs4);
extern FcBlanks *FcBlanksCreate(void);
extern void FcBlanksDestroy(FcBlanks * b);
extern FcBool FcBlanksIsMember(FcBlanks * b, FcChar32 ucs4);
extern FcBool FcCharSetAddChar(FcCharSet * fcs, FcChar32 ucs4);
extern FcCharSet *FcCharSetCopy(FcCharSet * src);
extern FcChar32 FcCharSetCount(const FcCharSet * a);
extern FcCharSet *FcCharSetCreate(void);
extern void FcCharSetDestroy(FcCharSet * fcs);
extern FcBool FcCharSetEqual(const FcCharSet * a, const FcCharSet
* b);
extern FcChar32 FcCharSetFirstPage(const FcCharSet * a, FcChar32 *
map,
                                FcChar32 * next);
extern FcBool FcCharSetHasChar(const FcCharSet * fcs, FcChar32
ucs4);
extern FcCharSet *FcCharSetIntersect(const FcCharSet * a,
                                const FcCharSet * b);
extern FcChar32 FcCharSetIntersectCount(const FcCharSet * a,
                                const FcCharSet * b);
extern FcBool FcCharSetIsSubset(const FcCharSet * a, const
FcCharSet * b);
extern FcChar32 FcCharSetNextPage(const FcCharSet * a, FcChar32 *
map,
                                FcChar32 * next);
extern FcCharSet *FcCharSetSubtract(const FcCharSet * a,
                                const FcCharSet * b);
extern FcChar32 FcCharSetSubtractCount(const FcCharSet * a,
                                const FcCharSet * b);
extern FcCharSet *FcCharSetUnion(const FcCharSet * a, const
FcCharSet * b);
extern FcBool FcConfigAppFontAddDir(FcConfig * config,
                                const FcChar8 * dir);
extern FcBool FcConfigAppFontAddFile(FcConfig * config,
                                const FcChar8 * file);
extern void FcConfigAppFontClear(FcConfig * config);
extern FcBool FcConfigBuildFonts(FcConfig * config);
extern void FcConfigDestroy(FcConfig * config);
extern FcBool FcConfigEnableHome(FcBool enable);
extern FcChar8 *FcConfigFilename(const FcChar8 * url);
extern FcBlanks *FcConfigGetBlanks(FcConfig * config);
extern FcChar8 *FcConfigGetCache(FcConfig * config);
extern FcStrList *FcConfigGetConfigDirs(FcConfig * config);
extern FcStrList *FcConfigGetConfigFiles(FcConfig * config);
extern FcConfig *FcConfigGetCurrent(void);
extern FcStrList *FcConfigGetFontDirs(FcConfig * config);
extern FcFontSet *FcConfigGetFonts(FcConfig * config, FcSetName
set);
extern int FcConfigGetRescanInterval(FcConfig * config);
extern int FcConfigGetRescanInterval(FcConfig * config);
extern FcChar8 *FcConfigHome(void);
extern FcBool FcConfigParseAndLoad(FcConfig * config, const FcChar8
* file,
                                FcBool complain);

```

```

extern FcBool FcConfigSetCurrent(FcConfig * config);
extern FcBool FcConfigSetRescanInterval(FcConfig * config,
                                       int rescanInterval);
extern FcBool FcConfigSetRescanInterval(FcConfig * config,
                                       int rescanInterval);
extern FcBool FcConfigSubstitute(FcConfig * config, FcPattern * p,
                                FcMatchKind kind);
extern FcBool FcConfigSubstituteWithPat(FcConfig * config,
                                       FcPattern * p,
                                       FcPattern * p_pat,
                                       FcMatchKind kind);
extern FcBool FcConfigUptoDate(FcConfig * config);
extern void FcDefaultSubstitute(FcPattern * pattern);
extern FcBool FcDirCacheValid(const FcChar8 * cache_file);
extern FcBool FcDirSave(FcFontSet * set, FcStrSet * dirs,
                       const FcChar8 * dir);
extern FcBool FcDirScan(FcFontSet * set, FcStrSet * dirs,
                       FcGlobalCache * cache, FcBlanks * blanks,
                       const FcChar8 * dir, FcBool force);
extern FcBool FcFileScan(FcFontSet * set, FcStrSet * dirs,
                       FcGlobalCache * cache, FcBlanks * blanks,
                       const FcChar8 * file, FcBool force);
extern void FcFini(void);
extern FcFontSet *FcFontList(FcConfig * config, FcPattern * p,
                             FcObjectSet * os);
extern FcPattern *FcFontMatch(FcConfig * config, FcPattern * p,
                              FcResult * result);
extern FcPattern *FcFontRenderPrepare(FcConfig * config, FcPattern
* pat,
                                     FcPattern * font);
extern FcBool FcFontSetAdd(FcFontSet * s, FcPattern * font);
extern FcFontSet *FcFontSetCreate(void);
extern void FcFontSetDestroy(FcFontSet * s);
extern FcFontSet *FcFontSetList(FcConfig * config, FcFontSet *
*sets,
                               int nsets, FcPattern * p,
                               FcObjectSet * os);
extern FcPattern *FcFontSetMatch(FcConfig * config, FcFontSet *
*sets,
                               int nsets, FcPattern * p,
                               FcResult * result);
extern void FcFontSetPrint(const FcFontSet * s);
extern FcFontSet *FcFontSetSort(FcConfig * config, FcFontSet *
*sets,
                               int nsets, FcPattern * p, FcBool trim,
                               FcCharSet * *csp, FcResult * result);
extern void FcFontSetSortDestroy(FcFontSet * fs);
extern FcFontSet *FcFontSetSort(FcConfig * config, FcPattern * p,
FcBool trim,
                               FcCharSet * *csp, FcResult * result);
extern FcPattern *FcFreeTypeQuery(const FcChar8 * file, int id,
                                FcBlanks * blanks, int *count);
extern int FcGetVersion(void);
extern FcBool FcInit(void);
extern FcBool FcInitBringUptoDate(void);
extern FcConfig *FcInitLoadConfig(void);
extern FcConfig *FcInitLoadConfigAndFonts(void);
extern FcBool FcInitReinitialize(void);
extern FcBool FcLangSetAdd(FcLangSet * ls, const FcChar8 * lang);
extern FcLangResult FcLangSetCompare(const FcLangSet * lsa,
                                    const FcLangSet * lsb);
extern FcBool FcLangSetContains(const FcLangSet * lsa,
                                const FcLangSet * lsb);
extern FcLangSet *FcLangSetCopy(const FcLangSet * ls);
extern FcLangSet *FcLangSetCreate(void);
extern void FcLangSetDestroy(FcLangSet * ls);

```

```

extern FcBool FcLangSetEqual(const FcLangSet * lsa, const FcLangSet
* lsb);
extern FcLangResult FcLangSetHasLang(const FcLangSet * ls,
const FcChar8 * lang);
extern FcChar32 FcLangSetHash(const FcLangSet * ls);
extern FcMatrix *FcMatrixCopy(const FcMatrix * mat);
extern FcBool FcMatrixEqual(const FcMatrix * mat1, const FcMatrix
* mat2);
extern void FcMatrixMultiply(FcMatrix * result, const FcMatrix * a,
const FcMatrix * b);
extern void FcMatrixRotate(FcMatrix * m, double c, double s);
extern void FcMatrixScale(FcMatrix * m, double sx, double sy);
extern void FcMatrixShear(FcMatrix * m, double sh, double sv);
extern FcBool FcNameConstant(FcChar8 * string, int *result);
extern const FcConstant *FcNameGetConstant(FcChar8 * string);
extern const FcObjectType *FcNameGetObjectType(const char *object);
extern FcPattern *FcNameParse(const FcChar8 * name);
extern FcBool FcNameRegisterConstants(const FcConstant * consts,
int nconsts);
extern FcBool FcNameRegisterObjectTypes(const FcObjectType * types,
int ntype);
extern FcChar8 *FcNameUnparse(FcPattern * pat);
extern FcBool FcNameUnregisterConstants(const FcConstant * consts,
int nconsts);
extern FcBool FcNameUnregisterObjectTypes(const FcObjectType *
types,
int ntype);
extern FcBool FcObjectSetAdd(FcObjectSet * os, const char *object);
extern FcObjectSet *FcObjectSetBuild(const char *first, ...);
extern FcObjectSet *FcObjectSetCreate(void);
extern void FcObjectSetDestroy(FcObjectSet * os);
extern FcObjectSet *FcObjectSetVaBuild(const char *first, va_list
va);
extern FcBool FcPatternAdd(FcPattern * p, const char *object,
FcValue value, FcBool append);
extern FcBool FcPatternAddBool(FcPattern * p, const char *object,
FcBool b);
extern FcBool FcPatternAddCharSet(FcPattern * p, const char *object,
const FcCharSet * c);
extern FcBool FcPatternAddDouble(FcPattern * p, const char *object,
double d);
extern FcBool FcPatternAddInteger(FcPattern * p, const char *object,
int i);
extern FcBool FcPatternAddLangSet(FcPattern * p, const char *object,
const FcLangSet * ls);
extern FcBool FcPatternAddMatrix(FcPattern * p, const char *object,
const FcMatrix * s);
extern FcBool FcPatternAddString(FcPattern * p, const char *object,
const FcChar8 * s);
extern FcBool FcPatternAddWeak(FcPattern * p, const char *object,
FcValue value, FcBool append);
extern FcPattern *FcPatternBuild(FcPattern * orig, ...);
extern FcPattern *FcPatternCreate(void);
extern FcBool FcPatternDel(FcPattern * p, const char *object);
extern void FcPatternDestroy(FcPattern * p);
extern FcPattern *FcPatternDuplicate(const FcPattern * p);
extern FcBool FcPatternEqual(const FcPattern * pa, const FcPattern
* pb);
extern FcBool FcPatternEqualSubset(const FcPattern * pa,
const FcPattern * pb,
const FcObjectSet * os);
extern FcResult FcPatternGet(const FcPattern * p, const char
*object,
int id, FcValue * v);
extern FcResult FcPatternGetBool(const FcPattern * p, const char
*object,

```



```

        int n, FcBool * b);
extern FcResult FcPatternGetCharSet(const FcPattern * p,
        const char *object, int n,
        FcCharSet * *c);
extern FcResult FcPatternGetDouble(const FcPattern * p, const char
*object,
        int n, double *d);
extern FcResult FcPatternGetInteger(const FcPattern * p,
        const char *object, int n, int *i);
extern FcResult FcPatternGetLangSet(const FcPattern * p,
        const char *object, int n,
        FcLangSet * *ls);
extern FcResult FcPatternGetMatrix(const FcPattern * p, const char
*object,
        int n, FcMatrix * *s);
extern FcResult FcPatternGetString(const FcPattern * p, const char
*object,
        int n, FcChar8 * *s);
extern FcChar32 FcPatternHash(const FcPattern * p);
extern void FcPatternPrint(const FcPattern * p);
extern void FcPatternReference(FcPattern * p);
extern FcPattern *FcPatternVaBuild(FcPattern * orig, va_list va);
extern FcChar8 *FcStrBasename(const FcChar8 * file);
extern int FcStrCmp(const FcChar8 * s1, const FcChar8 * s2);
extern int FcStrCmpIgnoreCase(const FcChar8 * s1, const FcChar8 *
s2);
extern FcChar8 *FcStrCopy(const FcChar8 * s);
extern FcChar8 *FcStrCopyFilename(const FcChar8 * s);
extern FcChar8 *FcStrDirname(const FcChar8 * file);
extern FcStrList *FcStrListCreate(FcStrSet * set);
extern void FcStrListDone(FcStrList * list);
extern FcChar8 *FcStrListNext(FcStrList * list);
extern FcBool FcStrSetAdd(FcStrSet * set, const FcChar8 * s);
extern FcBool FcStrSetAddFilename(FcStrSet * set, const FcChar8 *
s);
extern FcStrSet *FcStrSetCreate(void);
extern FcBool FcStrSetDel(FcStrSet * set, const FcChar8 * s);
extern void FcStrSetDestroy(FcStrSet * set);
extern FcBool FcStrSetEqual(FcStrSet * sa, FcStrSet * sb);
extern FcBool FcStrSetMember(FcStrSet * set, const FcChar8 * s);
extern int FcUcs4ToUtf8(FcChar32 ucs4, FcChar8 * dest);
extern FcBool FcUtf16Len(const FcChar8 * string, FcEndian endian,
int len,
        int *nchar, int *wchar);
extern int FcUtf16ToUcs4(const FcChar8 * src_orig, FcEndian endian,
        FcChar32 * dst, int len);
extern FcBool FcUtf8Len(const FcChar8 * string, int len, int *nchar,
        int *wchar);
extern int FcUtf8ToUcs4(const FcChar8 * src_orig, FcChar32 * dst,
        int len);
extern void FcValueDestroy(FcValue v);
extern FcBool FcValueEqual(FcValue va, FcValue vb);
extern void FcValuePrint(const FcValue v);
extern FcValue FcValueSave(FcValue v);

```

11 Commands and Utilities

11.1 Commands and Utilities

An LSB conforming implementation shall provide the commands and utilities as described in Table 11-1, with at least the behavior described as mandatory in the referenced underlying specification, with the following exceptions:

1. If any operand (except one which follows `--`) starts with a hyphen, the behavior is unspecified.

Rationale (Informative): Applications should place options before operands, or use `--`, as needed. This text is needed because, by default, GNU option parsing differs from POSIX, unless the environment variable `POSIXLY_CORRECT` is set. For example, `ls . -a` in GNU `ls` means to list the current directory, showing all files (that is, `"."` is an operand and `-a` is an option). In POSIX, `"."` and `-a` are both operands, and the command means to list the current directory, and also the file named `-a`. Suggesting that applications rely on the setting of the `POSIXLY_CORRECT` environment variable, or try to set it, seems worse than just asking the applications to invoke commands in ways which work with either the POSIX or GNU behaviors.

Table 11-1 Commands And Utilities

fc-cache [1]	fc-list [1]	fc-match [1]		

Referenced Specification(s)

[1]. This Specification

11.2 Command Behavior

This section contains descriptions for commands and utilities whose specified behavior in the LSB contradicts or extends the standards referenced. It also contains commands and utilities only required by the LSB and not specified by other standards.

fc-cache

Name

`fc-cache` — build font information cache files

Description

fc-cache scans the font directories on the system and builds font information cache files for applications using fontconfig for their font handling. If directory arguments are not given, `fc-cache` uses each directory in the current font configuration. Each directory is scanned for font files readable by FreeType. A cache is created which contains properties of each font and the associated filename. This cache is used to speed up application startup when using the fontconfig library.

Synopsis

```
fc-cache [ -fsvV? ] [ --force ] [ --system-only ] [ --verbose ] [ --version ] [ --help ]
[ dirs ]
```

Option and Operand Handling

This program follows the usual GNU command line syntax, with long options starting with two dashes ('-'). A summary of options is included below.

`-f --force`

Force re-generation of apparently up-to-date cache files, overriding the timestamp checking.

`-s --system-only`

Only scan system-wide directories, omitting the places located in the user's home directory.

`-v --verbose`

Display status information while busy.

`-? --help`

Show summary of options.

`-V --version`

Show version of the program and exit.

`dirs`

A list of directories to scan for fonts.

fc-list**Name**

`fc-list` — list available fonts.

Description

`fc-list` lists fonts and styles available on the system for applications using fontconfig.

Synopsis

`fc-list` [`-vV?`] [`--verbose`] [`--version`] [`--help`] [`pattern`] [`element`]

Option and Operand Handling

This program follows the usual GNU command line syntax, with long options starting with two dashes ('-'). A summary of options is included below.

`-v --verbose`

Display status information while busy.

`-? --help`

Show summary of options.

`-V --version`

Show version of the program and exit.

`pattern`

If this argument is set, only fonts matching pattern are displayed.

`element`

If set, the element property is displayed for matching fonts.

fc-match**Name**

`fc-match` — match available fonts

Description

`fc-match` matches font-pattern (empty pattern by default) using the normal fontconfig matching rules to find the best font available. If `--sort` is given, the sorted list of best matching fonts is displayed. With `--verbose`, the whole font pattern for each match is printed, otherwise only the file, family and style are printed.

VII Freetype library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

12 Libraries

12.1 Interfaces for libfreetype

Table 12-1 defines the library name and shared object name for the libfreetype library

Table 12-1 libfreetype Definition

Library:	libfreetype
SONAME:	libfreetype.so.6

The behavior of the interfaces in this library is specified by the following specifications:

[freetype 2.2] FreeType 2.2 Reference
 [LSB] This Specification

12.1.1 FreeType Library

12.1.1.1 Interfaces for FreeType Library

An LSB conforming implementation shall provide the generic functions for FreeType Library specified in Table 12-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 12-2 libfreetype - FreeType Library Function Interfaces

FT_Activate_Size [freetype 2.2]	FT_Add_Default_Modules [freetype 2.2]	FT_Add_Module [freetype 2.2]
FT_Angle_Diff [freetype 2.2]	FT_Atan2 [freetype 2.2]	FT_Attach_File [freetype 2.2]
FT_Attach_Stream [freetype 2.2]	FT_Bitmap_Convert [freetype 2.2]	FT_Bitmap_Copy [freetype 2.2]
FT_Bitmap_Done [freetype 2.2]	FT_Bitmap_Embolden [freetype 2.2]	FT_Bitmap_New [freetype 2.2]
FT_CeilFix [freetype 2.2]	FT_Cos [freetype 2.2]	FT_DivFix [freetype 2.2]
FT_Done_Face [freetype 2.2]	FT_Done_FreeType [freetype 2.2]	FT_Done_Glyph [freetype 2.2]
FT_Done_Library [freetype 2.2]	FT_Done_Size [freetype 2.2]	FT_FloorFix [freetype 2.2]
FT_Get_BDF_Charset_ID [freetype 2.2]	FT_Get_BDF_Property [freetype 2.2]	FT_Get_CMap_Language_ID [freetype 2.2]
FT_Get_Char_Index [freetype 2.2]	FT_Get_Charmap_Index [freetype 2.2]	FT_Get_First_Char [freetype 2.2]
FT_Get_Glyph [freetype 2.2]	FT_Get_Glyph_Name [freetype 2.2]	FT_Get_Kerning [freetype 2.2]
FT_Get_MM_Var [freetype 2.2]	FT_Get_Module [freetype 2.2]	FT_Get_Multi_Master [freetype 2.2]

FT_Get_Name_Index [freetype 2.2]	FT_Get_Next_Char [freetype 2.2]	FT_Get_PFR_Advance [freetype 2.2]
FT_Get_PFR_Kerning [freetype 2.2]	FT_Get_PFR_Metrics [freetype 2.2]	FT_Get_PS_Font_Info [freetype 2.2]
FT_Get_PS_Font_Private [freetype 2.2]	FT_Get_Postscript_Name [freetype 2.2]	FT_Get_Renderer [freetype 2.2]
FT_Get_Sfnt_Name [freetype 2.2]	FT_Get_Sfnt_Name_Count [freetype 2.2]	FT_Get_Sfnt_Table [freetype 2.2]
FT_Get_X11_Font_Format [LSB]	FT_Glyph_Copy [freetype 2.2]	FT_Glyph_Get_CBox [freetype 2.2]
FT_Glyph_Stroke [freetype 2.2]	FT_Glyph_StrokeBorder [freetype 2.2]	FT_Glyph_To_Bitmap [freetype 2.2]
FT_Glyph_Transform [freetype 2.2]	FT_Has_PS_Glyph_Names [freetype 2.2]	FT_Init_FreeType [freetype 2.2]
FT_Library_Version [freetype 2.2]	FT_List_Add [freetype 2.2]	FT_List_Finalize [freetype 2.2]
FT_List_Find [freetype 2.2]	FT_List_Insert [freetype 2.2]	FT_List_Iterate [freetype 2.2]
FT_List_Remove [freetype 2.2]	FT_List_Up [freetype 2.2]	FT_Load_Char [freetype 2.2]
FT_Load_Glyph [freetype 2.2]	FT_Load_Sfnt_Table [freetype 2.2]	FT_Matrix_Invert [freetype 2.2]
FT_Matrix_Multiply [freetype 2.2]	FT_MulDiv [freetype 2.2]	FT_MulFix [freetype 2.2]
FT_New_Face [freetype 2.2]	FT_New_Library [freetype 2.2]	FT_New_Memory_Face [freetype 2.2]
FT_New_Size [freetype 2.2]	FT_OpenType_Validate [freetype 2.2]	FT_Open_Face [freetype 2.2]
FT_Outline_Check [freetype 2.2]	FT_Outline_Copy [freetype 2.2]	FT_Outline_Decompose [freetype 2.2]
FT_Outline_Done [freetype 2.2]	FT_Outline_GetInsideBorder [freetype 2.2]	FT_Outline_GetOutsideBorder [freetype 2.2]
FT_Outline_Get_BBox [freetype 2.2]	FT_Outline_Get_Bitmap [freetype 2.2]	FT_Outline_Get_CBox [freetype 2.2]
FT_Outline_Get_Orientation [freetype 2.2]	FT_Outline_New [freetype 2.2]	FT_Outline_Render [freetype 2.2]
FT_Outline_Reverse [freetype 2.2]	FT_Outline_Transform [freetype 2.2]	FT_Outline_Translate [freetype 2.2]
FT_Remove_Module [freetype 2.2]	FT_Render_Glyph [freetype 2.2]	FT_RoundFix [freetype 2.2]
FT_Select_Charmap [freetype 2.2]	FT_Set_Char_Size [freetype 2.2]	FT_Set_Charmap [freetype 2.2]

FT_Set_Debug_Hook [freetype 2.2]	FT_Set_MM_Blend_Coordinates [freetype 2.2]	FT_Set_MM_Design_Coordinates [freetype 2.2]
FT_Set_Pixel_Sizes [freetype 2.2]	FT_Set_Renderer [freetype 2.2]	FT_Set_Transform [freetype 2.2]
FT_Set_Var_Blend_Coordinates [freetype 2.2]	FT_Set_Var_Design_Coordinates [freetype 2.2]	FT_Sfnt_Table_Info [freetype 2.2]
FT_Sin [freetype 2.2]	FT_Stroker_BeginSubPath [freetype 2.2]	FT_Stroker_ConicTo [freetype 2.2]
FT_Stroker_CubicTo [freetype 2.2]	FT_Stroker_Done [freetype 2.2]	FT_Stroker_EndSubPath [freetype 2.2]
FT_Stroker_Export [freetype 2.2]	FT_Stroker_ExportBorder [freetype 2.2]	FT_Stroker_GetBorderCounts [freetype 2.2]
FT_Stroker_GetCounts [freetype 2.2]	FT_Stroker_LineTo [freetype 2.2]	FT_Stroker_New [freetype 2.2]
FT_Stroker_ParseOutline [freetype 2.2]	FT_Stroker_Rewind [freetype 2.2]	FT_Stroker_Set [freetype 2.2]
FT_Tan [freetype 2.2]	FT_Vector_From_Polar [freetype 2.2]	FT_Vector_Length [freetype 2.2]
FT_Vector_Polarize [freetype 2.2]	FT_Vector_Rotate [freetype 2.2]	FT_Vector_Transform [freetype 2.2]
FT_Vector_Unit [freetype 2.2]		

12.2 Data Definitions for libfreetype

This section defines global identifiers and their values that are associated with interfaces contained in libfreetype. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

12.2.1 freetype/config/ftheader.h

```
#define FT_BEGIN_HEADER
#define FT_END_HEADER
#define FT_ERRORDEF( e, v, s ) \
    FT_ERRORDEF( FT_ERR_CAT( FT_ERR_PREFIX, e ), v + FT_ERR_BASE, \
s )
```



```

#define FT_NOERRORDEF( e,v,s) \
    FT_ERRORDEF( FT_ERR_CAT( FT_ERR_PREFIX, e ), v, s )
#define FT_ERR_BASE 0
#define FT_CONFIG_CONFIG_H <freetype/config/ftconfig.h>
#define FT_CONFIG_MODULES_H <freetype/config/ftmodule.h>
#define FT_CONFIG_OPTIONS_H <freetype/config/ftoption.h>
#define FT_CONFIG_STANDARD_LIBRARY_H <freetype/config/ftstdlib.h>
#define FT_FREETYPE_H <freetype/freetype.h>
#define FT_BBOX_H <freetype/ftbbox.h>
#define FT_BDF_H <freetype/ftbdf.h>
#define FT_BITMAP_H <freetype/ftbitmap.h>
#define FT_CACHE_H <freetype/ftcache.h>
#define FT_CACHE_INTERNAL_CACHE_H <freetype/ftcache.h>
#define FT_CACHE_INTERNAL_GLYPH_H <freetype/ftcache.h>
#define FT_CACHE_INTERNAL_IMAGE_H <freetype/ftcache.h>
#define FT_CACHE_INTERNAL_MANAGER_H <freetype/ftcache.h>
#define FT_CACHE_INTERNAL_MRU_H <freetype/ftcache.h>
#define FT_CACHE_INTERNAL_SBITS_H <freetype/ftcache.h>
#define FT_CACHE_MANAGER_H <freetype/ftcache.h>
#define FT_ERROR_DEFINITIONS_H <freetype/fterrdef.h>
#define FT_ERRORS_H <freetype/fterrors.h>
#define FT_GLYPH_H <freetype/ftglyph.h>
#define FT_GZIP_H <freetype/ftgzip.h>
#define FT_IMAGE_H <freetype/ftimage.h>
#define FT_INCREMENTAL_H <freetype/ftinrm.h>
#define FT_LIST_H <freetype/ftlist.h>
#define FT_LZW_H <freetype/ftlzw.h>
#define FT_MAC_H <freetype/ftmac.h>
#define FT_MULTIPLE_MASTERS_H <freetype/ftmm.h>
#define FT_MODULE_H <freetype/ftmodapi.h>
#define FT_MODULE_ERRORS_H <freetype/ftmoderr.h>
#define FT_OPENTYPE_VALIDATE_H <freetype/ftotval.h>
#define FT_OUTLINE_H <freetype/ftoutln.h>
#define FT_RENDER_H <freetype/ftrender.h>
#define FT_SIZES_H <freetype/ftsizes.h>
#define FT_SFNT_NAMES_H <freetype/ftsnames.h>
#define FT_STROKER_H <freetype/ftstroke.h>
#define FT_SYNTHESIS_H <freetype/ftsynth.h>
#define FT_SYSTEM_H <freetype/ftsystem.h>
#define FT_TRIGONOMETRY_H <freetype/fttrigon.h>
#define FT_TYPES_H <freetype/fttypes.h>
#define FT_WINFONTS_H <freetype/ftwinfnt.h>
#define FT_XFREE86_H <freetype/ftxf86.h>
#define FT_TYPE1_TABLES_H <freetype/tltables.h>
#define FT_TRUETYPE_IDS_H <freetype/ttnameid.h>
#define FT_TRUETYPE_TABLES_H <freetype/tttables.h>
#define FT_TRUETYPE_TAGS_H <freetype/tttags.h>
#define FT_TRUETYPE_UNPATENTED_H <freetype/ttunpat.h>
#define FT_ERRORDEF( e,v,s) e = v,
#define FT_ERROR_START_LIST enum {
#define FT_CACHE_CHARMAP_H FT_CACHE_H
#define FT_CACHE_IMAGE_H FT_CACHE_H
#define FT_CACHE_SMALL_BITMAPS_H FT_CACHE_H
#define FT_ERR_PREFIX FT_Err_
#define FT_ERROR_END_LIST FT_ERR_CAT( FT_ERR_PREFIX, Max ) };
#define FT_ERR_CAT(x,y) FT_ERR_XCAT( x, y )
#define FT_ERR_XCAT(x,y) x ## y

```

12.2.2 freetype/config/ftoption.h

```

#define FT_CONFIG_OPTION_ADOBE_GLYPH_LIST
#define FT_CONFIG_OPTION_GUESSING_EMBEDDED_RFORK
#define FT_CONFIG_OPTION_MAC_FONTS
#define FT_CONFIG_OPTION_POSTSCRIPT_NAMES

```

```

#define FT_CONFIG_OPTION_USE_LZW
#define FT_CONFIG_OPTION_USE_ZLIB
#define TT_CONFIG_CMAP_FORMAT_0
#define TT_CONFIG_CMAP_FORMAT_10
#define TT_CONFIG_CMAP_FORMAT_12
#define TT_CONFIG_CMAP_FORMAT_2
#define TT_CONFIG_CMAP_FORMAT_4
#define TT_CONFIG_CMAP_FORMAT_6
#define TT_CONFIG_CMAP_FORMAT_8
#define TT_CONFIG_OPTION_EMBEDDED_BITMAPS
#define TT_CONFIG_OPTION_GX_VAR_SUPPORT
#define TT_CONFIG_OPTION_INTERPRETER_SWITCH
#define TT_CONFIG_OPTION_POSTSCRIPT_NAMES
#define TT_CONFIG_OPTION_SFNT_NAMES
#define TT_CONFIG_OPTION_UNPATENTED_HINTING
#define T1_MAX_SUBRS_CALLS 16
#define FT_RENDER_POOL_SIZE 16384L
#define T1_MAX_CHARSTRINGS_OPERANDS 256
#define FT_MAX_MODULES 32
#define T1_MAX_DICT_DEPTH 5

```

12.2.3 freetype/config/ftstdlib.h

```

#define ft_atol atol
#define ft_exit exit
#define FT_INT_MAX INT_MAX
#define ft_isalnum isalnum
#define ft_isdigit isdigit
#define ft_islower islower
#define ft_isupper isupper
#define ft_isxdigit isxdigit
#define ft_jump_buf jmp_buf
#define ft_longjmp longjmp
#define ft_memcmp memcmp
#define ft_memcpy memcpy
#define ft_memmove memmove
#define ft_memset memset
#define ft_ptrdiff_t ptrdiff_t
#define ft_qsort qsort
#define ft_setjmp setjmp
#define ft_sprintf sprintf
#define ft_strcat strcat
#define ft_strcmp strcmp
#define ft_strcpy strcpy
#define ft_strlen strlen
#define ft_strncmp strncmp
#define ft_strncpy strncpy
#define ft_strchr strchr
#define FT_UINT_MAX UINT_MAX
#define FT_ULONG_MAX ULONG_MAX

```

12.2.4 freetype/freetype.h

```

#define FT_HAS_FAST_GLYPHS(face) \
    ( face->face_flags & FT_FACE_FLAG_FAST_GLYPHS )
#define FT_HAS_FIXED_SIZES(face) \
    ( face->face_flags & FT_FACE_FLAG_FIXED_SIZES )
#define FT_IS_FIXED_WIDTH(face) \
    ( face->face_flags & FT_FACE_FLAG_FIXED_WIDTH )
#define FT_HAS_GLYPH_NAMES(face) \
    ( face->face_flags & FT_FACE_FLAG_GLYPH_NAMES )
#define FT_HAS_HORIZONTAL(face) \
    ( face->face_flags & FT_FACE_FLAG_HORIZONTAL )

```

```

#define FT_HAS_MULTIPLE_MASTERS(face) \
    ( face->face_flags & FT_FACE_FLAG_MULTIPLE_MASTERS )
#define FT_ENC_TAG(value,a,b,c,d) \
    value = ( ( (FT_UInt32)(a) << 24 ) | ( (FT_UInt32)(b) << 16 ) \
    | ( \
        (FT_UInt32)(c) << 8 ) | (FT_UInt32)(d) ) )
#define FT_LOAD_TARGET_(x) ( (FT_Int32)( (x) & 15 ) << 16 )
#define FT_LOAD_TARGET_MODE(x) ( (FT_Render_Mode)( ( (x) >16 ) & \
15 ) ) )
#define FT_STYLE_FLAG_ITALIC ( 1 << 0 )
#define FT_STYLE_FLAG_BOLD ( 1 << 1 )
#define FT_FACE_FLAG_SCALABLE ( 1L << 0 )
#define FT_FACE_FLAG_FIXED_SIZES ( 1L << 1 )
#define FT_FACE_FLAG_EXTERNAL_STREAM ( 1L << 10 )
#define FT_FACE_FLAG_FIXED_WIDTH ( 1L << 2 )
#define FT_FACE_FLAG_SFNT ( 1L << 3 )
#define FT_FACE_FLAG_HORIZONTAL ( 1L << 4 )
#define FT_FACE_FLAG_VERTICAL ( 1L << 5 )
#define FT_FACE_FLAG_KERNING ( 1L << 6 )
#define FT_FACE_FLAG_FAST_GLYPHS ( 1L << 7 )
#define FT_FACE_FLAG_MULTIPLE_MASTERS ( 1L << 8 )
#define FT_FACE_FLAG_GLYPH_NAMES ( 1L << 9 )
#define FT_HAS_KERNING(face) ( face->face_flags & \
FT_FACE_FLAG_KERNING )
#define FT_IS_SCALABLE(face) ( face->face_flags & \
FT_FACE_FLAG_SCALABLE )
#define FT_IS_SFNT(face) ( face->face_flags & \
FT_FACE_FLAG_SFNT )
#define FT_HAS_VERTICAL(face) ( face->face_flags & \
FT_FACE_FLAG_VERTICAL )
#define FT_LOAD_DEFAULT 0x0
#define FT_LOAD_NO_SCALE 0x1
#define FT_OPEN_MEMORY 0x1
#define FT_LOAD_VERTICAL_LAYOUT 0x10
#define FT_OPEN_PARAMS 0x10
#define FT_LOAD_MONOCHROME 0x1000
#define FT_LOAD_NO_HINTING 0x2
#define FT_OPEN_STREAM 0x2
#define FT_LOAD_FORCE_AUTOHINT 0x20
#define FT_LOAD_IGNORE_GLOBAL_ADVANCE_WIDTH 0x200
#define FT_LOAD_LINEAR_DESIGN 0x2000
#define FT_LOAD_RENDER 0x4
#define FT_OPEN_PATHNAME 0x4
#define FT_LOAD_CROP_BITMAP 0x40
#define FT_LOAD_NO_RECURSE 0x400
#define FT_LOAD_SBITS_ONLY 0x4000
#define FT_LOAD_NO_BITMAP 0x8
#define FT_OPEN_DRIVER 0x8
#define FT_LOAD_PEDANTIC 0x80
#define FT_LOAD_IGNORE_TRANSFORM 0x800
#define FT_LOAD_NO_AUTOHINT 0x8000U
#define FREETYPE_MINOR 1
#define FREETYPE_PATCH 10
#define FREETYPE_MAJOR 2
#define FT_LOAD_TARGET_LCD
FT_LOAD_TARGET_( FT_RENDER_MODE_LCD )
#define FT_LOAD_TARGET_LCD_V
FT_LOAD_TARGET_( FT_RENDER_MODE_LCD_V )
#define FT_LOAD_TARGET_LIGHT
FT_LOAD_TARGET_( FT_RENDER_MODE_LIGHT )
#define FT_LOAD_TARGET_MONO
FT_LOAD_TARGET_( FT_RENDER_MODE_MONO )
#define FT_LOAD_TARGET_NORMAL
FT_LOAD_TARGET_( FT_RENDER_MODE_NORMAL )

typedef struct FT_FaceRec_ *FT_Face;

```

```

typedef struct FT_ModuleRec_ *FT_Module;
typedef struct FT_LibraryRec_ *FT_Library;
typedef struct FT_ModuleRec_ FT_ModuleRec;
typedef struct FT_RendererRec_ *FT_Renderer;
typedef struct FT_GlyphSlotRec_ *FT_GlyphSlot;
struct FT_GlyphSlotRec_ {
    FT_Library library;
    FT_Face face;
    FT_GlyphSlot next;
    FT_UInt reserved;
    FT_Generic generic;
    FT_Glyph_Metrics metrics;
    FT_Fixed linearHoriAdvance;
    FT_Fixed linearVertAdvance;
    FT_Vector advance;
    FT_Glyph_Format format;
    FT_Bitmap bitmap;
    FT_Int bitmap_left;
    FT_Int bitmap_top;
    FT_Outline outline;
    FT_UInt num_subglyphs;
    FT_SubGlyph subglyphs;
    void *control_data;
    long int control_len;
    FT_Pos lsb_delta;
    FT_Pos rsb_delta;
    void *other;
    FT_Slot_Internal internal;
};
typedef struct FT_Glyph_Metrics_ {
    FT_Pos width;
    FT_Pos height;
    FT_Pos horiBearingX;
    FT_Pos horiBearingY;
    FT_Pos horiAdvance;
    FT_Pos vertBearingX;
    FT_Pos vertBearingY;
    FT_Pos vertAdvance;
} FT_Glyph_Metrics;
typedef struct FT_SubGlyphRec_ *FT_SubGlyph;
typedef struct FT_Slot_InternalRec_ *FT_Slot_Internal;
struct FT_CharMapRec_ {
    FT_Face face;
    FT_Encoding encoding;
    FT_UShort platform_id;
    FT_UShort encoding_id;
};
typedef enum FT_Encoding_ {
    FT_ENCODING_NONE = 0,
    FT_ENCODING_MS_SYMBOL = 1937337698,
    FT_ENCODING_UNICODE = 1970170211,
    FT_ENCODING_SJIS = 1936353651,
    FT_ENCODING_GB2312 = 1734484000,
    FT_ENCODING_BIG5 = 1651074869,
    FT_ENCODING_WANSUNG = 2002873971,
    FT_ENCODING_JOHAB = 1785686113,
    FT_ENCODING_MS_SJIS = 1936353651,
    FT_ENCODING_MS_GB2312 = 1734484000,
    FT_ENCODING_MS_BIG5 = 1651074869,
    FT_ENCODING_MS_WANSUNG = 2002873971,
    FT_ENCODING_MS_JOHAB = 1785686113,
    FT_ENCODING_ADOBE_STANDARD = 1094995778,
    FT_ENCODING_ADOBE_EXPERT = 1094992453,
    FT_ENCODING_ADOBE_CUSTOM = 1094992451,
    FT_ENCODING_ADOBE_LATIN_1 = 1818326065,
    FT_ENCODING_OLD_LATIN_2 = 1818326066,

```

```

    FT_ENCODING_APPLE_ROMAN = 1634889070
} FT_Encoding;
typedef struct FT_CharMapRec_ *FT_CharMap;
typedef long int FT_F26Dot6;
typedef enum FT_Render_Mode_ {
    FT_RENDER_MODE_NORMAL = 0,
    FT_RENDER_MODE_LIGHT = 1,
    FT_RENDER_MODE_MONO = 2,
    FT_RENDER_MODE_LCD = 3,
    FT_RENDER_MODE_LCD_V = 4,
    FT_RENDER_MODE_MAX = 5
} FT_Render_Mode;
typedef struct FT_Parameter_ {
    FT_ULong tag;
    FT_Pointer data;
} FT_Parameter;
typedef struct FT_Open_Args_ {
    FT_UInt flags;
    const FT_Byte *memory_base;
    FT_Long memory_size;
    FT_String *pathname;
    FT_Stream stream;
    FT_Module driver;
    FT_Int num_params;
    FT_Parameter *params;
} FT_Open_Args;
struct FT_SizeRec_ {
    FT_Face face;
    FT_Generic generic;
    FT_Size_Metrics metrics;
    FT_Size_Internal internal;
};
typedef struct FT_Size_Metrics_ {
    FT_UShort x_ppem;
    FT_UShort y_ppem;
    FT_Fixed x_scale;
    FT_Fixed y_scale;
    FT_Pos ascender;
    FT_Pos descender;
    FT_Pos height;
    FT_Pos max_advance;
} FT_Size_Metrics;
typedef struct FT_Size_InternalRec_ *FT_Size_Internal;
typedef struct FT_SizeRec_ *FT_Size;
typedef struct FT_Bitmap_Size_ {
    FT_Short height;
    FT_Short width;
    FT_Pos size;
    FT_Pos x_ppem;
    FT_Pos y_ppem;
} FT_Bitmap_Size;
typedef struct FT_Face_InternalRec_ *FT_Face_Internal;
typedef struct FT_DriverRec_ *FT_Driver;
typedef struct FT_FaceRec_ {
    FT_Long num_faces;
    FT_Long face_index;
    FT_Long face_flags;
    FT_Long style_flags;
    FT_Long num_glyphs;
    const FT_String *family_name;
    const FT_String *style_name;
    FT_Int num_fixed_sizes;
    FT_Bitmap_Size *available_sizes;
    FT_Int num_charmaps;
    FT_CharMap *charmaps;
    FT_Generic generic;

```

```

    FT_BBox bbox;
    FT_UShort units_per_EM;
    FT_Short ascender;
    FT_Short descender;
    FT_Short height;
    FT_Short max_advance_width;
    FT_Short max_advance_height;
    FT_Short underline_position;
    FT_Short underline_thickness;
    FT_GlyphSlot glyph;
    FT_Size size;
    FT_CharMap charmap;
    FT_Driver driver;
    FT_Memory memory;
    FT_Stream stream;
    FT_ListRec sizes_list;
    FT_Generic autohint;
    void *extensions;
    FT_Face_Internal internal;
} FT_FaceRec;
typedef enum FT_Kerning_Mode_ {
    FT_KERNING_DEFAULT = 0,
    FT_KERNING_UNFITTED = 1,
    FT_KERNING_UNSCALED = 2
} FT_Kerning_Mode;
extern FT_Error FT_Attach_File(FT_Face, const char *);
extern FT_Error FT_Attach_Stream(FT_Face, FT_Open_Args *);
extern FT_Fixed FT_CeilFix(FT_Fixed);
extern FT_Long FT_DivFix(FT_Long, FT_Long);
extern FT_Error FT_Done_Face(FT_Face);
extern FT_Error FT_Done_FreeType(FT_Library);
extern FT_Fixed FT_FloorFix(FT_Fixed);
extern FT_UInt FT_Get_Char_Index(FT_Face, FT_ULong);
extern FT_Int FT_Get_Charmap_Index(FT_CharMap);
extern FT_ULong FT_Get_First_Char(FT_Face, FT_UInt *);
extern FT_Error FT_Get_Glyph_Name(FT_Face, FT_UInt, FT_Pointer,
FT_UInt);
extern FT_Error FT_Get_Kerning(FT_Face, FT_UInt, FT_UInt, FT_UInt,
    FT_Vector *);
extern FT_UInt FT_Get_Name_Index(FT_Face, FT_String *);
extern FT_ULong FT_Get_Next_Char(FT_Face, FT_ULong, FT_UInt *);
extern const char *FT_Get_Postscript_Name(FT_Face);
extern FT_Error FT_Init_FreeType(FT_Library *);
extern void FT_Library_Version(FT_Library, FT_Int *, FT_Int *,
FT_Int *);
extern FT_Error FT_Load_Char(FT_Face, FT_ULong, FT_Int32);
extern FT_Error FT_Load_Glyph(FT_Face, FT_UInt, FT_Int32);
extern FT_Long FT_MulDiv(FT_Long, FT_Long, FT_Long);
extern FT_Long FT_MulFix(FT_Long, FT_Long);
extern FT_Error FT_New_Face(FT_Library, const char *, FT_Long,
FT_Face *);
extern FT_Error FT_New_Memory_Face(FT_Library, const FT_Byte *,
FT_Long,
    FT_Long, FT_Face *);
extern FT_Error FT_Open_Face(FT_Library, const FT_Open_Args *,
FT_Long,
    FT_Face *);
extern FT_Error FT_Render_Glyph(FT_GlyphSlot, FT_Render_Mode);
extern FT_Fixed FT_RoundFix(FT_Fixed);
extern FT_Error FT_Select_Charmap(FT_Face, FT_Encoding);
extern FT_Error FT_Set_Char_Size(FT_Face, FT_F26Dot6, FT_F26Dot6,
FT_UInt,
    FT_UInt);
extern FT_Error FT_Set_Charmap(FT_Face, FT_CharMap);
extern FT_Error FT_Set_Pixel_Sizes(FT_Face, FT_UInt, FT_UInt);
extern void FT_Set_Transform(FT_Face, FT_Matrix *, FT_Vector *);

```

```
extern void FT_Vector_Transform(FT_Vector *, const FT_Matrix *);
```

12.2.5 freetype/ftbbox.h

```
extern FT_Error FT_Outline_Get_BBox(FT_Outline *, FT_BBox *);
```

12.2.6 freetype/ftbdf.h

```
typedef enum BDF_PropertyType_ {
    BDF_PROPERTY_TYPE_NONE = 0,
    BDF_PROPERTY_TYPE_ATOM = 1,
    BDF_PROPERTY_TYPE_INTEGER = 2,
    BDF_PROPERTY_TYPE_CARDINAL = 3
} BDF_PropertyType;
typedef struct BDF_PropertyRec_ {
    BDF_PropertyType type;
    union {
        const char *atom;
        FT_Int32 integer;
        FT_UInt32 cardinal;
    } u;
} BDF_PropertyRec;
extern FT_Error FT_Get_BDF_Charset_ID(FT_Face, const char **,
                                      const char **);
extern FT_Error FT_Get_BDF_Property(FT_Face, const char *,
                                    BDF_PropertyRec *);
```

12.2.7 freetype/ftbitmap.h

```
extern FT_Error FT_Bitmap_Convert(FT_Library, const FT_Bitmap *,
                                  FT_Bitmap *, FT_Int);
extern FT_Error FT_Bitmap_Copy(FT_Library, const FT_Bitmap *,
                               FT_Bitmap *);
extern FT_Error FT_Bitmap_Done(FT_Library, FT_Bitmap *);
extern FT_Error FT_Bitmap_Embolden(FT_Library, FT_Bitmap *, FT_Pos,
                                   FT_Pos);
extern void FT_Bitmap_New(FT_Bitmap *);
```

12.2.8 freetype/fterrors.h

```
enum {
    FT_Err_Ok = 0x00,
    FT_Err_Cannot_Open_Resource = 0x01,
    FT_Err_Unknown_File_Format = 0x02,
    FT_Err_Invalid_File_Format = 0x03,
    FT_Err_Invalid_Version = 0x04,
    FT_Err_Lower_Module_Version = 0x05,
    FT_Err_Invalid_Argument = 0x06,
    FT_Err_Unimplemented_Feature = 0x07,
    FT_Err_Invalid_Table = 0x08,
    FT_Err_Invalid_Offset = 0x09,
    FT_Err_Invalid_Glyph_Index = 0x10,
    FT_Err_Invalid_Character_Code = 0x11,
    FT_Err_Invalid_Glyph_Format = 0x12,
    FT_Err_Cannot_Render_Glyph = 0x13,
    FT_Err_Invalid_Outline = 0x14,
    FT_Err_Invalid_Composite = 0x15,
    FT_Err_Too_Many_Hints = 0x16,
    FT_Err_Invalid_Pixel_Size = 0x17,
    FT_Err_Invalid_Handle = 0x20,
```

```

FT_Err_Invalid_Library_Handle = 0x21,
FT_Err_Invalid_Driver_Handle = 0x22,
FT_Err_Invalid_Face_Handle = 0x23,
FT_Err_Invalid_Size_Handle = 0x24,
FT_Err_Invalid_Slot_Handle = 0x25,
FT_Err_Invalid_CharMap_Handle = 0x26,
FT_Err_Invalid_Cache_Handle = 0x27,
FT_Err_Invalid_Stream_Handle = 0x28,
FT_Err_Too_Many_Drivers = 0x30,
FT_Err_Too_Many_Extensions = 0x31,
FT_Err_Out_Of_Memory = 0x40,
FT_Err_Unlisted_Object = 0x41,
FT_Err_Cannot_Open_Stream = 0x51,
FT_Err_Invalid_Stream_Seek = 0x52,
FT_Err_Invalid_Stream_Skip = 0x53,
FT_Err_Invalid_Stream_Read = 0x54,
FT_Err_Invalid_Stream_Operation = 0x55,
FT_Err_Invalid_Frame_Operation = 0x56,
FT_Err_Nested_Frame_Access = 0x57,
FT_Err_Invalid_Frame_Read = 0x58,
FT_Err_Raster_Uninitialized = 0x60,
FT_Err_Raster_Corrupted = 0x61,
FT_Err_Raster_Overflow = 0x62,
FT_Err_Raster_Negative_Height = 0x63,
FT_Err_Too_Many_Caches = 0x70,
FT_Err_Invalid_Opcode = 0x80,
FT_Err_Too_Few_Arguments = 0x81,
FT_Err_Stack_Overflow = 0x82,
FT_Err_Code_Overflow = 0x83,
FT_Err_Bad_Argument = 0x84,
FT_Err_Divide_By_Zero = 0x85,
FT_Err_Invalid_Reference = 0x86,
FT_Err_Debug_OpCode = 0x87,
FT_Err_ENDF_In_Exec_Stream = 0x88,
FT_Err_Nested_DEFS = 0x89,
FT_Err_Invalid_CodeRange = 0x8A,
FT_Err_Execution_Too_Long = 0x8B,
FT_Err_Too_Many_Function_Defs = 0x8C,
FT_Err_Too_Many_Instruction_Defs = 0x8D,
FT_Err_Table_Missing = 0x8E,
FT_Err_Horiz_Header_Missing = 0x8F,
FT_Err_Locations_Missing = 0x90,
FT_Err_Name_Table_Missing = 0x91,
FT_Err_CMap_Table_Missing = 0x92,
FT_Err_Hmtx_Table_Missing = 0x93,
FT_Err_Post_Table_Missing = 0x94,
FT_Err_Invalid_Horiz_Metrics = 0x95,
FT_Err_Invalid_CharMap_Format = 0x96,
FT_Err_Invalid_PPem = 0x97,
FT_Err_Invalid_Vert_Metrics = 0x98,
FT_Err_Could_Not_Find_Context = 0x99,
FT_Err_Invalid_Post_Table_Format = 0x9A,
FT_Err_Invalid_Post_Table = 0x9B,
FT_Err_Syntax_Error = 0xA0,
FT_Err_Stack_Underflow = 0xA1,
FT_Err_Ignore = 0xA2,
FT_Err_Missing_Startfont_Field = 0xB0,
FT_Err_Missing_Font_Field = 0xB1,
FT_Err_Missing_Size_Field = 0xB2,
FT_Err_Missing_Chars_Field = 0xB3,
FT_Err_Missing_Startchar_Field = 0xB4,
FT_Err_Missing_Encoding_Field = 0xB5,
FT_Err_Missing_Bbx_Field = 0xB6,
FT_Err_Bbx_Too_Big = 0xB7,
FT_Err_Max = 0xB8
};

```


12.2.9 freetype/ftglyph.h

```

typedef struct FT_Glyph_Class_ {
    FT_Long glyph_size;
    FT_Glyph_Format glyph_format;
    FT_Glyph_InitFunc glyph_init;
    FT_Glyph_DoneFunc glyph_done;
    FT_Glyph_CopyFunc glyph_copy;
    FT_Glyph_TransformFunc glyph_transform;
    FT_Glyph_GetBBoxFunc glyph_bbox;
    FT_Glyph_PrepareFunc glyph_prepare;
} FT_Glyph_Class;
typedef struct FT_GlyphRec_ *FT_Glyph;
typedef FT_Error(*FT_Glyph_InitFunc) (FT_Glyph, FT_GlyphSlot);
typedef void (*FT_Glyph_DoneFunc) (FT_Glyph);
typedef FT_Error(*FT_Glyph_CopyFunc) (FT_Glyph, FT_Glyph);
typedef void (*FT_Glyph_TransformFunc) (FT_Glyph, const FT_Matrix
*,
                                     const FT_Vector *);
typedef void (*FT_Glyph_GetBBoxFunc) (FT_Glyph, FT_BBox*);
typedef FT_Error(*FT_Glyph_PrepareFunc) (FT_Glyph, FT_GlyphSlot);
typedef enum FT_Glyph_BBox_Mode_ {
    FT_GLYPH_BBOX_UNSCALED = 0,
    FT_GLYPH_BBOX_SUBPIXELS = 0,
    FT_GLYPH_BBOX_GRIDFIT = 1,
    FT_GLYPH_BBOX_TRUNCATE = 2,
    FT_GLYPH_BBOX_PIXELS = 3
} FT_Glyph_BBox_Mode;
struct FT_OutlineGlyphRec_ {
    FT_GlyphRec root;
    FT_Outline outline;
};
typedef struct FT_OutlineGlyphRec_ *FT_OutlineGlyph;
typedef struct FT_GlyphRec_ {
    FT_Library library;
    const FT_Glyph_Class_ *clazz;
    FT_Glyph_Format format;
    FT_Vector advance;
} FT_GlyphRec;
struct FT_BitmapGlyphRec_ {
    FT_GlyphRec root;
    FT_Int left;
    FT_Int top;
    FT_Bitmap bitmap;
};
typedef struct FT_BitmapGlyphRec_ *FT_BitmapGlyph;
extern void FT_Done_Glyph(FT_Glyph);
extern FT_Error FT_Get_Glyph(FT_GlyphSlot, FT_Glyph *);
extern FT_Error FT_Glyph_Copy(FT_Glyph, FT_Glyph *);
extern void FT_Glyph_Get_CBox(FT_Glyph, FT_UInt, FT_BBox *);
extern FT_Error FT_Glyph_To_Bitmap(FT_Glyph *, FT_Render_Mode,
FT_Vector *,
                                FT_Bool);
extern FT_Error FT_Glyph_Transform(FT_Glyph, FT_Matrix *, FT_Vector
*);
extern FT_Error FT_Matrix_Invert(FT_Matrix *);
extern void FT_Matrix_Multiply(const FT_Matrix *, FT_Matrix *);

```

12.2.10 freetype/ftimage.h

```

#define FT_CURVE_TAG_TOUCH_BOTH \
    ( FT_CURVE_TAG_TOUCH_X | FT_CURVE_TAG_TOUCH_Y )
#define FT_IMAGE_TAG(value, _x1, _x2, _x3, _x4) \

```

```

        value = ( ( (unsigned long)_x1 << 24 ) | ( (unsigned long)_x2
<< 16 ) \
        | ( (unsigned long)_x3 << 8 ) | (unsigned long)_x4 )
#define FT_CURVE_TAG(flag)      ( flag & 3 )
#define FT_CURVE_TAG_CONIC      0
#define FT_OUTLINE_NONE         0x0
#define FT_RASTER_FLAG_DEFAULT  0x0
#define FT_OUTLINE_OWNER        0x1
#define FT_RASTER_FLAG_AA        0x1
#define FT_OUTLINE_HIGH_PRECISION 0x100
#define FT_OUTLINE_EVEN_ODD_FILL 0x2
#define FT_RASTER_FLAG_DIRECT    0x2
#define FT_OUTLINE_SINGLE_PASS   0x200
#define FT_OUTLINE_REVERSE_FILL  0x4
#define FT_RASTER_FLAG_CLIP      0x4
#define FT_OUTLINE_IGNORE_DROPOUTS 0x8
#define FT_CURVE_TAG_ON         1
#define FT_CURVE_TAG_TOUCH_Y    16
#define FT_CURVE_TAG_CUBIC      2
#define FT_CURVE_TAG_TOUCH_X    8
#define FT_Curve_Tag_Conic      FT_CURVE_TAG_CONIC
#define FT_Curve_Tag_Cubic      FT_CURVE_TAG_CUBIC
#define FT_Curve_Tag_On         FT_CURVE_TAG_ON
#define FT_Curve_Tag_Touch_X    FT_CURVE_TAG_TOUCH_X
#define FT_Curve_Tag_Touch_Y    FT_CURVE_TAG_TOUCH_Y
#define FT_Outline_ConicTo_Func FT_Outline_ConicToFunc
#define FT_Outline_CubicTo_Func FT_Outline_CubicToFunc
#define FT_Outline_LineTo_Func  FT_Outline_LineToFunc
#define FT_Outline_MoveTo_Func  FT_Outline_MoveToFunc
#define FT_Raster_Done_Func      FT_Raster_DoneFunc
#define FT_Raster_New_Func       FT_Raster_NewFunc
#define FT_Raster_Render_Func    FT_Raster_RenderFunc
#define FT_Raster_Reset_Func     FT_Raster_ResetFunc
#define FT_Raster_Set_Mode_Func  FT_Raster_SetModeFunc
#define FT_Raster_Span_Func      FT_SpanFunc

typedef struct FT_Bitmap_ {
    int rows;
    int width;
    int pitch;
    unsigned char *buffer;
    short num_grays;
    char pixel_mode;
    char palette_mode;
    void *palette;
} FT_Bitmap;
typedef long int FT_Pos;
typedef struct FT_Vector_ {
    FT_Pos x;
    FT_Pos y;
} FT_Vector;
typedef struct FT_Outline_ {
    short n_contours;
    short n_points;
    FT_Vector *points;
    char *tags;
    short int *contours;
    int flags;
} FT_Outline;
typedef enum FT_Glyph_Format_ {
    FT_GLYPH_FORMAT_NONE = 0,
    FT_GLYPH_FORMAT_COMPOSITE = 1668246896,
    FT_GLYPH_FORMAT_BITMAP = 1651078259,
    FT_GLYPH_FORMAT_OUTLINE = 1869968492,
    FT_GLYPH_FORMAT_PLOTTER = 1886154612
} FT_Glyph_Format;

```

```

typedef struct FT_BBox_ {
    FT_Pos xMin;
    FT_Pos yMin;
    FT_Pos xMax;
    FT_Pos yMax;
} FT_BBox;

typedef struct FT_RasterRec_ *FT_Raster;
typedef int (*FT_Raster_NewFunc) (void *, FT_Raster *);
typedef void (*FT_Raster_ResetFunc) (FT_Raster, unsigned char *,
                                     long unsigned int);
typedef int (*FT_Raster_SetModeFunc) (FT_Raster, long unsigned int,
                                     void *);

typedef struct FT_Span_ {
    short x;
    unsigned short len;
    unsigned char coverage;
} FT_Span;

typedef void (*FT_SpanFunc) (int, int, const FT_Span *, void *);
typedef int (*FT_Raster_BitTest_Func) (int, int, void *);
typedef void (*FT_Raster_BitSet_Func) (int, int, void *);
typedef struct FT_Raster_Params_ {
    const FT_Bitmap *target;
    const void *source;
    int flags;
    FT_SpanFunc gray_spans;
    FT_SpanFunc black_spans;
    FT_Raster_BitTest_Func bit_test;
    FT_Raster_BitSet_Func bit_set;
    void *user;
    FT_BBox clip_box;
} FT_Raster_Params;

typedef int (*FT_Raster_RenderFunc) (FT_Raster, const
FT_Raster_Params *);
typedef void (*FT_Raster_DoneFunc) (FT_Raster);
typedef struct FT_Raster_Funcs_ {
    FT_Glyph_Format glyph_format;
    FT_Raster_NewFunc raster_new;
    FT_Raster_ResetFunc raster_reset;
    FT_Raster_SetModeFunc raster_set_mode;
    FT_Raster_RenderFunc raster_render;
    FT_Raster_DoneFunc raster_done;
} FT_Raster_Funcs;

typedef int (*FT_Outline_MoveToFunc) (const FT_Vector *, void *);
typedef int (*FT_Outline_LineToFunc) (const FT_Vector *, void *);
typedef int (*FT_Outline_ConicToFunc) (const FT_Vector *,
                                     const FT_Vector *, void *);
typedef int (*FT_Outline_CubicToFunc) (const FT_Vector *,
                                     const FT_Vector *,
                                     const FT_Vector *, void *);

typedef struct FT_Outline_Funcs_ {
    FT_Outline_MoveToFunc move_to;
    FT_Outline_LineToFunc line_to;
    FT_Outline_ConicToFunc conic_to;
    FT_Outline_CubicToFunc cubic_to;
    int shift;
    FT_Pos delta;
} FT_Outline_Funcs;

typedef enum FT_Pixel_Mode_ {
    FT_PIXEL_MODE_NONE = 0,
    FT_PIXEL_MODE_MONO,
    FT_PIXEL_MODE_GRAY,
    FT_PIXEL_MODE_GRAY2,
    FT_PIXEL_MODE_GRAY4,
    FT_PIXEL_MODE_LCD,
    FT_PIXEL_MODE_LCD_V,
    FT_PIXEL_MODE_MAX

```

```
} FT_Pixel_Mode;
```

12.2.11 freetype/ftincrem.h

```
#define FT_PARAM_TAG_INCREMENTAL      FT_MAKE_TAG( 'i', 'n', 'c',  
'r' )
```

12.2.12 freetype/ftlist.h

```
typedef void (*FT_List_Destructor) (FT_Memory, void *, void *);  
typedef FT_Error (*FT_List_Iterator) (FT_ListNode, void *);  
extern void FT_List_Add(FT_List, FT_ListNode);  
extern void FT_List_Finalize(FT_List, FT_List_Destructor,  
FT_Memory,  
void *);  
extern FT_ListNode FT_List_Find(FT_List, void *);  
extern void FT_List_Insert(FT_List, FT_ListNode);  
extern FT_Error FT_List_Iterate(FT_List, FT_List_Iterator, void *);  
extern void FT_List_Remove(FT_List, FT_ListNode);  
extern void FT_List_Up(FT_List, FT_ListNode);
```

12.2.13 freetype/ftmm.h

```
typedef struct FT_MM_Axis_ {  
    FT_String *name;  
    FT_Long minimum;  
    FT_Long maximum;  
} FT_MM_Axis;  
typedef struct FT_Multi_Master_ {  
    FT_UInt num_axis;  
    FT_UInt num_designs;  
    FT_MM_Axis axis[4];  
} FT_Multi_Master;  
typedef struct FT_Var_Axis_ {  
    FT_String *name;  
    FT_Fixed minimum;  
    FT_Fixed def;  
    FT_Fixed maximum;  
    FT_ULong tag;  
    FT_UInt strid;  
} FT_Var_Axis;  
typedef struct FT_Var_Named_Style_ {  
    FT_Fixed *coords;  
    FT_UInt strid;  
} FT_Var_Named_Style;  
typedef struct FT_MM_Var_ {  
    FT_UInt num_axis;  
    FT_UInt num_designs;  
    FT_UInt num_namedstyles;  
    FT_Var_Axis *axis;  
    FT_Var_Named_Style *namedstyle;  
} FT_MM_Var;  
extern FT_Error FT_Get_MM_Var(FT_Face, FT_MM_Var * *);  
extern FT_Error FT_Get_Multi_Master(FT_Face, FT_Multi_Master *);  
extern FT_Error FT_Set_MM_Blend_Coordinates(FT_Face, FT_UInt,  
FT_Fixed *);  
extern FT_Error FT_Set_MM_Design_Coordinates(FT_Face, FT_UInt,  
FT_Long *);  
extern FT_Error FT_Set_Var_Blend_Coordinates(FT_Face, FT_UInt,  
FT_Fixed *);  
extern FT_Error FT_Set_Var_Design_Coordinates(FT_Face, FT_UInt,  
FT_Fixed *);
```

12.2.14 freetype/ftmodapi.h

```

#define FT_MODULE_DRIVER_SCALABLE      0x100
#define FT_MODULE_DRIVER_NO_OUTLINES  0x200
#define FT_MODULE_DRIVER_HAS_HINTER    0x400
#define FT_MODULE_FONT_DRIVER          1
#define FT_MODULE_RENDERER              2
#define FT_MODULE_HINTER                4
#define FT_MODULE_STYLER                8

typedef FT_Error (*FT_Module_Constructor) (FT_Module);
typedef void (*FT_Module_Destructor) (FT_Module);
typedef FT_Pointer FT_Module_Interface;
typedef FT_Module_Interface (*FT_Module_Reqister) (FT_Module,
                                                    const char *);

typedef struct FT_Module_Class_ {
    FT_ULong module_flags;
    FT_Long module_size;
    const FT_String *module_name;
    FT_Fixed module_version;
    FT_Fixed module_requires;
    const void *module_interface;
    FT_Module_Constructor module_init;
    FT_Module_Destructor module_done;
    FT_Module_Reqister get_interface;
} FT_Module_Class;

typedef void (*FT_DebugHook_Func) (void *);
extern void FT_Add_Default_Modules(FT_Library);
extern FT_Error FT_Add_Module(FT_Library, const FT_Module_Class *);
extern FT_Error FT_Done_Library(FT_Library);
extern FT_Module FT_Get_Module(FT_Library, const char *);
extern FT_Error FT_New_Library(FT_Memory, FT_Library *);
extern FT_Error FT_Remove_Module(FT_Library, FT_Module);
extern void FT_Set_Debug_Hook(FT_Library, FT_UInt,
                              FT_DebugHook_Func);

```

12.2.15 freetype/ftmoderr.h

```

#define FT_MODERR_START_LIST    enum {
#define FT_MODERRDEF(e,v,s)    FT_Mod_Err_ ## e = 0,
#define FT_MODERR_END_LIST      FT_Mod_Err_Max };

```

12.2.16 freetype/ftotval.h

```

#define FT_VALIDATE_OT \
    FT_VALIDATE_BASE | FT_VALIDATE_GDEF | FT_VALIDATE_GPOS | \
    FT_VALIDATE_GSUB | FT_VALIDATE_JSTF
#define FT_VALIDATE_BASE      0x0100
#define FT_VALIDATE_GDEF     0x0200
#define FT_VALIDATE_GPOS     0x0400
#define FT_VALIDATE_GSUB     0x0800
#define FT_VALIDATE_JSTF     0x1000

extern FT_Error FT_OpenType_Validate(FT_Face, FT_UInt, FT_Bytes *,
                                     FT_Bytes *, FT_Bytes *, FT_Bytes *);

```

12.2.17 freetype/ftoutln.h

```

typedef enum {

```

```

    FT_ORIENTATION_TRUETYPE = 0,
    FT_ORIENTATION_POSTSCRIPT = 1,
    FT_ORIENTATION_FILL_RIGHT = 0,
    FT_ORIENTATION_FILL_LEFT = 1
} FT_Orientation;
extern FT_Error FT_Outline_Check(FT_Outline *);
extern FT_Error FT_Outline_Copy(const FT_Outline *, FT_Outline *);
extern FT_Error FT_Outline-Decompose(FT_Outline *,
                                     const FT_Outline_Funcs *, void *);
extern FT_Error FT_Outline_Done(FT_Library, FT_Outline *);
extern FT_Error FT_Outline_Get_Bitmap(FT_Library, FT_Outline *,
                                     const FT_Bitmap *);
extern void FT_Outline_Get_CBox(const FT_Outline *, FT_BBox *);
extern FT_Orientation FT_Outline_Get_Orientation(FT_Outline *);
extern FT_Error FT_Outline_New(FT_Library, FT_UInt, FT_Int,
FT_Outline *);
extern FT_Error FT_Outline_Render(FT_Library, FT_Outline *,
FT_Raster_Params *);
extern void FT_Outline_Reverse(FT_Outline *);
extern void FT_Outline_Transform(const FT_Outline *, const
FT_Matrix *);
extern void FT_Outline_Translate(const FT_Outline *, FT_Pos,
FT_Pos);

```

12.2.18 freetype/ftpr.h

```

extern FT_Error FT_Get_PFR_Advance(FT_Face, FT_UInt, FT_Pos *);
extern FT_Error FT_Get_PFR_Kerning(FT_Face, FT_UInt, FT_UInt,
FT_Vector *);
extern FT_Error FT_Get_PFR_Metrics(FT_Face, FT_UInt *, FT_UInt *,
FT_Fixed *, FT_Fixed *);

```

12.2.19 freetype/ftrender.h

```

#define FT_Glyph_Done_Func      FT_Glyph_DoneFunc
#define FT_Glyph_BBox_Func     FT_Glyph_GetBBoxFunc
#define FT_Glyph_Init_Func     FT_Glyph_InitFunc
#define FT_Glyph_Prepare_Func  FT_Glyph_PrepareFunc
#define FT_Renderer_getCBox     FT_Renderer_GetCBoxFunc
#define FT_Renderer_render     FT_Renderer_RenderFunc
#define FT_Renderer_setMode     FT_Renderer_SetModeFunc
#define FT_Renderer_transform   FT_Renderer_TransformFunc

typedef FT_Error(*FT_Renderer_RenderFunc) (FT_Renderer,
FT_GlyphSlot,
                                     FT_UInt, const FT_Vector *);
typedef FT_Error(*FT_Renderer_TransformFunc) (FT_Renderer,
FT_GlyphSlot,
                                     const FT_Matrix *,
                                     const FT_Vector *);
typedef void (*FT_Renderer_GetCBoxFunc) (FT_Renderer, FT_GlyphSlot,
FT_BBox *);
typedef FT_Error(*FT_Renderer_SetModeFunc) (FT_Renderer, FT_ULong,
FT_Pointer);
typedef struct FT_Renderer_Class_ {
    FT_Module_Class root;
    FT_Glyph_Format glyph_format;
    FT_Renderer_RenderFunc render_glyph;
    FT_Renderer_TransformFunc transform_glyph;
    FT_Renderer_GetCBoxFunc get_glyph_cbox;
    FT_Renderer_SetModeFunc set_mode;
    FT_Raster_Funcs *raster_class;
} FT_Renderer_Class;

```

```
extern FT_Renderer FT_Get_Renderer(FT_Library, FT_Glyph_Format);
extern FT_Error FT_Set_Renderer(FT_Library, FT_Renderer, FT_UInt,
                               FT_Parameter *);
```

12.2.20 freetype/ftsizes.h

```
extern FT_Error FT_Activate_Size(FT_Size);
extern FT_Error FT_Done_Size(FT_Size);
extern FT_Error FT_New_Size(FT_Face, FT_Size *);
```

12.2.21 freetype/ftsnames.h

```
typedef struct FT_SfntName_ {
    FT_UShort platform_id;
    FT_UShort encoding_id;
    FT_UShort language_id;
    FT_UShort name_id;
    FT_Byte *string;
    FT_UInt string_len;
} FT_SfntName;
extern FT_Error FT_Get_Sfnt_Name(FT_Face, FT_UInt, FT_SfntName *);
extern FT_UInt FT_Get_Sfnt_Name_Count(FT_Face);
```

12.2.22 freetype/ftstroke.h

```
typedef enum {
    FT_STROKER_LINECAP_BUTT = 0,
    FT_STROKER_LINECAP_ROUND = 1,
    FT_STROKER_LINECAP_SQUARE = 2
} FT_Stroker_LineCap;
typedef enum {
    FT_STROKER_LINEJOIN_ROUND = 0,
    FT_STROKER_LINEJOIN_BEVEL = 1,
    FT_STROKER_LINEJOIN_MITER = 2
} FT_Stroker_LineJoin;
typedef struct FT_StrokerRec_ *FT_Stroker;
typedef enum {
    FT_STROKER_BORDER_LEFT = 0,
    FT_STROKER_BORDER_RIGHT = 1
} FT_StrokerBorder;
extern FT_Error FT_Glyph_Stroke(FT_Glyph *, FT_Stroker, FT_Bool);
extern FT_Error FT_Glyph_StrokeBorder(FT_Glyph *, FT_Stroker,
FT_Bool,
                                FT_Bool);
extern FT_StrokerBorder FT_Outline_GetInsideBorder(FT_Outline *);
extern FT_StrokerBorder FT_Outline_GetOutsideBorder(FT_Outline *);
extern FT_Error FT_Stroker_BeginSubPath(FT_Stroker, FT_Vector *,
FT_Bool);
extern FT_Error FT_Stroker_ConicTo(FT_Stroker, FT_Vector *,
FT_Vector *);
extern FT_Error FT_Stroker_CubicTo(FT_Stroker, FT_Vector *,
FT_Vector *,
                                FT_Vector *);
extern void FT_Stroker_Done(FT_Stroker);
extern FT_Error FT_Stroker_EndSubPath(FT_Stroker);
extern void FT_Stroker_Export(FT_Stroker, FT_Outline *);
extern void FT_Stroker_ExportBorder(FT_Stroker, FT_StrokerBorder,
                                FT_Outline *);
extern FT_Error FT_Stroker_GetBorderCounts(FT_Stroker,
FT_StrokerBorder,
                                FT_UInt *, FT_UInt *);
```

```

extern FT_Error FT_Stroker_GetCounts(FT_Stroker, FT_UInt *, FT_UInt
*);
extern FT_Error FT_Stroker_LineTo(FT_Stroker, FT_Vector *);
extern FT_Error FT_Stroker_New(FT_Library, FT_Stroker *);
extern FT_Error FT_Stroker_ParseOutline(FT_Stroker, FT_Outline *,
FT_Bool);
extern void FT_Stroker_Rewind(FT_Stroker);
extern void FT_Stroker_Set(FT_Stroker, FT_Fixed,
FT_Stroker_LineCap,
FT_Stroker_LineJoin, FT_Fixed);

```

12.2.23 freetype/ftsystem.h

```

struct FT_MemoryRec_ {
    void *user;
    FT_Alloc_Func alloc;
    FT_Free_Func free;
    FT_Realloc_Func realloc;
};
typedef struct FT_MemoryRec_ *FT_Memory;
typedef void (*FT_Alloc_Func) (FT_Memory, long int);
typedef void (*FT_Free_Func) (FT_Memory, void *);
typedef void (*FT_Realloc_Func) (FT_Memory, long int, long int,
void *);
union FT_StreamDesc_ {
    long int value;
    void *pointer;
};
struct FT_StreamRec_ {
    unsigned char *base;
    long unsigned int size;
    long unsigned int pos;
    FT_StreamDesc descriptor;
    FT_StreamDesc pathname;
    FT_Stream_IOFunc read;
    FT_Stream_CloseFunc close;
    FT_Memory memory;
    unsigned char *cursor;
    unsigned char *limit;
};
typedef union FT_StreamDesc_ {
    long int value;
    void *pointer;
} FT_StreamDesc;
typedef struct FT_StreamRec_ *FT_Stream;
typedef long unsigned int (*FT_Stream_IOFunc) (FT_Stream,
long unsigned int,
unsigned char *,
long unsigned int);
typedef void (*FT_Stream_CloseFunc) (FT_Stream);

```

12.2.24 freetype/fttrigon.h

```

#define FT_ANGLE_PI    ( 180L << 16 )
#define FT_ANGLE_2PI  ( FT_ANGLE_PI * 2 )
#define FT_ANGLE_PI2   ( FT_ANGLE_PI / 2 )
#define FT_ANGLE_PI4   ( FT_ANGLE_PI / 4 )

typedef FT_Fixed FT_Angle;
extern FT_Angle FT_Angle_Diff(FT_Angle, FT_Angle);
extern FT_Angle FT_Atan2(FT_Fixed, FT_Fixed);
extern FT_Fixed FT_Cos(FT_Angle);
extern FT_Fixed FT_Sin(FT_Angle);

```



```

extern FT_Fixed FT_Tan(FT_Angle);
extern void FT_Vector_From_Polar(FT_Vector *, FT_Fixed, FT_Angle);
extern FT_Fixed FT_Vector_Length(FT_Vector *);
extern void FT_Vector_Polarize(FT_Vector *, FT_Fixed *, FT_Angle
*);
extern void FT_Vector_Rotate(FT_Vector *, FT_Angle);
extern void FT_Vector_Unit(FT_Vector *, FT_Angle);

```

12.2.25 freetype/fttypes.h

```

#define FT_MAKE_TAG(_x1, _x2, _x3, _x4) \
    ( ( (FT_ULong)_x1 << 24 ) | ( (FT_ULong)_x2 << 16 ) | \
    ( (FT_ULong)_x3 << 8 ) | (FT_ULong)_x4 )
#define FT_BOOL(x) (FT_Bool)( x )
#define FT_IS_EMPTY(list) ( (list).head == 0 )
#define FT_ERROR_BASE(x) ( (x) & 0xFF )
#define FT_ERROR_MODULE(x) ( (x) & 0xFF00U )

typedef unsigned int FT_UInt;
typedef struct FT_ListNodeRec_ *FT_ListNode;
typedef struct FT_ListRec_ *FT_List;
typedef int FT_Error;
typedef void (*FT_Generic_Finalizer) (void *);
typedef struct FT_Generic_ {
    void *data;
    FT_Generic_Finalizer finalizer;
} FT_Generic;
typedef int FT_Int;
typedef long unsigned int FT_ULong;
typedef long int FT_Long;
typedef char FT_String;
typedef long int FT_Fixed;
typedef void *FT_Pointer;
typedef struct FT_ListRec_ {
    FT_ListNode head;
    FT_ListNode tail;
} FT_ListRec;
typedef short unsigned int FT_UShort;
typedef struct FT_Matrix_ {
    FT_Fixed xx;
    FT_Fixed xy;
    FT_Fixed yx;
    FT_Fixed yy;
} FT_Matrix;
typedef unsigned char FT_Bool;
typedef unsigned char FT_Byte;
typedef int FT_Int32;
typedef unsigned int FT_UInt32;
typedef const FT_Byte *FT_Bytes;
typedef short int FT_Short;
typedef struct FT_ListNodeRec_ {
    FT_ListNode prev;
    FT_ListNode next;
    void *data;
} FT_ListNodeRec;
typedef char FT_Char;

```

12.2.26 freetype/ftxf86.h

```

extern const char *FT_Get_X11_Font_Format(FT_Face);

```

12.2.27 freetype/t1tables.h

```

#define T1_MAX_MM_DESIGNS      16
#define T1_MAX_MM_MAP_POINTS  20
#define T1_MAX_MM_AXIS      4
#define t1_blend_blue_scale    T1_BLEND_BLUE_SCALE
#define t1_blend_blue_shift    T1_BLEND_BLUE_SHIFT
#define t1_blend_blue_values   T1_BLEND_BLUE_VALUES
#define t1_blend_family_blues  T1_BLEND_FAMILY_BLUES
#define                          t1_blend_family_other_blues
T1_BLEND_FAMILY_OTHER_BLUES
#define t1_blend_force_bold    T1_BLEND_FORCE_BOLD
#define t1_blend_italic_angle  T1_BLEND_ITALIC_ANGLE
#define t1_blend_max          T1_BLEND_MAX
#define t1_blend_other_blues   T1_BLEND_OTHER_BLUES
#define t1_blend_standard_height T1_BLEND_STANDARD_HEIGHT
#define t1_blend_standard_widths T1_BLEND_STANDARD_WIDTH
#define t1_blend_stem_snap_heights T1_BLEND_STEM_SNAP_HEIGHTS
#define t1_blend_stem_snap_widths T1_BLEND_STEM_SNAP_WIDTHS
#define                          t1_blend_underline_position
T1_BLEND_UNDERLINE_POSITION
#define                          t1_blend_underline_thickness
T1_BLEND_UNDERLINE_THICKNESS

typedef struct PS_PrivateRec_ {
    FT_Int unique_id;
    FT_Int lenIV;
    FT_Byte num_blue_values;
    FT_Byte num_other_blues;
    FT_Byte num_family_blues;
    FT_Byte num_family_other_blues;
    FT_Short blue_values[14];
    FT_Short other_blues[10];
    FT_Short family_blues[14];
    FT_Short family_other_blues[10];
    FT_Fixed blue_scale;
    FT_Int blue_shift;
    FT_Int blue_fuzz;
    FT_UShort standard_width[1];
    FT_UShort standard_height[1];
    FT_Byte num_snap_widths;
    FT_Byte num_snap_heights;
    FT_Bool force_bold;
    FT_Bool round_stem_up;
    FT_Short snap_widths[13];
    FT_Short snap_heights[13];
    FT_Fixed expansion_factor;
    FT_Long language_group;
    FT_Long password;
    FT_Short min_feature[2];
} PS_PrivateRec;
typedef struct PS_FontInfoRec {
    FT_String *version;
    FT_String *notice;
    FT_String *full_name;
    FT_String *family_name;
    FT_String *weight;
    FT_Long italic_angle;
    FT_Bool is_fixed_pitch;
    FT_Short underline_position;
    FT_UShort underline_thickness;
} PS_FontInfoRec;
typedef enum {
    T1_BLEND_UNDERLINE_POSITION,
    T1_BLEND_UNDERLINE_THICKNESS,

```

```

T1_BLEND_ITALIC_ANGLE,
T1_BLEND_BLUE_VALUES,
T1_BLEND_OTHER_BLUES,
T1_BLEND_STANDARD_WIDTH,
T1_BLEND_STANDARD_HEIGHT,
T1_BLEND_STEM_SNAP_WIDTHS,
T1_BLEND_STEM_SNAP_HEIGHTS,
T1_BLEND_BLUE_SCALE,
T1_BLEND_BLUE_SHIFT,
T1_BLEND_FAMILY_BLUES,
T1_BLEND_FAMILY_OTHER_BLUES,
T1_BLEND_FORCE_BOLD,
T1_BLEND_MAX
} T1_Blend_Flags;
extern FT_Error FT_Get_PS_Font_Info(FT_Face, PS_FontInfoRec *);
extern FT_Error FT_Get_PS_Font_Private(FT_Face, PS_PrivateRec *);
extern FT_Int FT_Has_PS_Glyph_Names(FT_Face);

```

12.2.28 freetype/ttnameid.h

```

#define TT_MS_LANGID_KIRGHIZ_KIRGHIZ_REPUBLIC \
    TT_MS_LANGID_KIRGHIZ_KIRGHIZSTAN
#define TT_MS_LANGID_SOTHO_SOUTHERN_SOUTH_AFRICA \
    TT_MS_LANGID_SEPEDI_SOUTH_AFRICA
#define TT_UCR_COMBINING_DIACRITICAL_MARKS_SYMB \
    TT_UCR_COMBINING_DIACRITICS_SYMB
#define TT_UCR_BASIC_LATIN (1L << 0)
#define TT_UCR_COMBINING_HALF_MARKS (1L << 0)
#define TT_UCR_SUPERSCRIPTS_SUBSCRIPTS (1L << 0)
#define TT_UCR_CJK_COMPATIBILITY_FORMS (1L << 1)
#define TT_UCR_CURRENCY_SYMBOLS (1L << 1)
#define TT_UCR_LATIN1_SUPPLEMENT (1L << 1)
#define TT_UCR_ARMENIAN (1L << 10)
#define TT_UCR_ENCLOSED_ALPHANUMERICS (1L << 10)
#define TT_UCR_Myanmar (1L << 10)
#define TT_UCR_BOX_DRAWING (1L << 11)
#define TT_UCR_ETHIOPIA (1L << 11)
#define TT_UCR_HEBREW (1L << 11)
#define TT_UCR_BLOCK_ELEMENTS (1L << 12)
#define TT_UCR_CHEROKEE (1L << 12)
#define TT_UCR_ARABIC (1L << 13)
#define TT_UCR_CANADIAN_ABORIGINAL_SYLLABICS (1L << 13)
#define TT_UCR_GEOMETRIC_SHAPES (1L << 13)
#define TT_UCR_MISCELLANEOUS_SYMBOLS (1L << 14)
#define TT_UCR_OGHAM (1L << 14)
#define TT_UCR_DEVANAGARI (1L << 15)
#define TT_UCR_DINGBATS (1L << 15)
#define TT_UCR_RUNIC (1L << 15)
#define TT_UCR_BENGALI (1L << 16)
#define TT_UCR_CJK_SYMBOLS (1L << 16)
#define TT_UCR_KHMER (1L << 16)
#define TT_UCR_GURMUKHI (1L << 17)
#define TT_UCR_HIRAGANA (1L << 17)
#define TT_UCR_MONGOLIAN (1L << 17)
#define TT_UCR_BRAILLE (1L << 18)
#define TT_UCR_GUJARATI (1L << 18)
#define TT_UCR_KATAKANA (1L << 18)
#define TT_UCR_BOOMOFO (1L << 19)
#define TT_UCR_ORIYA (1L << 19)
#define TT_UCR_YI (1L << 19)
#define TT_UCR_COMBINING_DIACRITICS_SYMB (1L << 2)
#define TT_UCR_LATIN_EXTENDED_A (1L << 2)
#define TT_UCR_SMALL_FORM_VARIANTS (1L << 2)
#define TT_UCR_HANGUL_COMPATIBILITY_JAMO (1L << 20)
#define TT_UCR_PHILIPPINE (1L << 20)

```

```

#define TT_UCR_TAMIL (1L << 20)
#define TT_UCR_CJK_MISC (1L << 21)
#define TT_UCR_OLD_ITALIC (1L << 21)
#define TT_UCR_TELUGU (1L << 21)
#define TT_UCR_ENCLOSED_CJK_LETTERS_MONTHS (1L << 22)
#define TT_UCR_GOTHIC (1L << 22)
#define TT_UCR_KANNADA (1L << 22)
#define TT_UCR_CJK_COMPATIBILITY (1L << 23)
#define TT_UCR_DESERET (1L << 23)
#define TT_UCR_MALAYALAM (1L << 23)
#define TT_UCR_HANGUL (1L << 24)
#define TT_UCR_MUSICAL_SYMBOLS (1L << 24)
#define TT_UCR_THAI (1L << 24)
#define TT_UCR_LAO (1L << 25)
#define TT_UCR_MATH_ALPHANUMERIC_SYMBOLS (1L << 25)
#define TT_UCR_SURROGATES (1L << 25)
#define TT_UCR_GEORGIAN (1L << 26)
#define TT_UCR_PRIVATE_USE_SUPPLEMENTARY (1L << 26)
#define TT_UCR_CJK_UNIFIED_IDEOGRAPHS (1L << 27)
#define TT_UCR_VARIATION_SELECTORS (1L << 27)
#define TT_UCR_HANGUL_JAMO (1L << 28)
#define TT_UCR_PRIVATE_USE (1L << 28)
#define TT_UCR_TAGS (1L << 28)
#define TT_UCR_CJK_COMPATIBILITY_IDEOGRAPHS (1L << 29)
#define TT_UCR_LATIN_EXTENDED_ADDITIONAL (1L << 29)
#define TT_UCR_ARABIC_PRESENTATIONS_B (1L << 3)
#define TT_UCR_LATIN_EXTENDED_B (1L << 3)
#define TT_UCR_LETTERLIKE_SYMBOLS (1L << 3)
#define TT_UCR_ALPHABETIC_PRESENTATION_FORMS (1L << 30)
#define TT_UCR_GREEK_EXTENDED (1L << 30)
#define TT_UCR_ARABIC_PRESENTATIONS_A (1L << 31)
#define TT_UCR_GENERAL_PUNCTUATION (1L << 31)
#define TT_UCR_HALFWIDTH_FULLWIDTH_FORMS (1L << 4)
#define TT_UCR_IPA_EXTENSIONS (1L << 4)
#define TT_UCR_NUMBER_FORMS (1L << 4)
#define TT_UCR_ARROWS (1L << 5)
#define TT_UCR_SPACING_MODIFIER (1L << 5)
#define TT_UCR_SPECIALS (1L << 5)
#define TT_UCR_COMBINING_DIACRITICS (1L << 6)
#define TT_UCR_MATHEMATICAL_OPERATORS (1L << 6)
#define TT_UCR_TIBETAN (1L << 6)
#define TT_UCR_GREEK (1L << 7)
#define TT_UCR_MISCELLANEOUS_TECHNICAL (1L << 7)
#define TT_UCR_SYRIAC (1L << 7)
#define TT_UCR_CONTROL_PICTURES (1L << 8)
#define TT_UCR_THAANA (1L << 8)
#define TT_UCR_CYRILLIC (1L << 9)
#define TT_UCR_OCR (1L << 9)
#define TT_UCR_SINHALA (1L << 9)
#define TT_ADOBE_ID_STANDARD 0
#define TT_APPLE_ID_DEFAULT 0
#define TT_ISO_ID_7BIT_ASCII 0
#define TT_MAC_ID_ROMAN 0
#define TT_MAC_LANGID_ENGLISH 0
#define TT_MS_ID_SYMBOL_CS 0
#define TT_NAME_ID_COPYRIGHT 0
#define TT_PLATFORM_APPLE_UNICODE 0
#define TT_MS_LANGID_ARABIC_GENERAL 0x0001
#define TT_MS_LANGID_CHINESE_GENERAL 0x0004
#define TT_MS_LANGID_ENGLISH_GENERAL 0x0009
#define TT_MS_LANGID_ARABIC_SAUDI_ARABIA 0x0401
#define TT_MS_LANGID_BULGARIAN_BULGARIA 0x0402
#define TT_MS_LANGID_CATALAN_SPAIN 0x0403
#define TT_MS_LANGID_CHINESE_TAIWAN 0x0404
#define TT_MS_LANGID_CZECH_CZECH_REPUBLIC 0x0405
#define TT_MS_LANGID_DANISH_DENMARK 0x0406

```

```

#define TT_MS_LANGID_GERMAN_GERMANY 0x0407
#define TT_MS_LANGID_GREEK_GREECE 0x0408
#define TT_MS_LANGID_ENGLISH_UNITED_STATES 0x0409
#define TT_MS_LANGID_SPANISH_SPAIN_TRADITIONAL_SORT 0x040a
#define TT_MS_LANGID_FINNISH_FINLAND 0x040b
#define TT_MS_LANGID_FRENCH_FRANCE 0x040c
#define TT_MS_LANGID_HEBREW_ISRAEL 0x040d
#define TT_MS_LANGID_HUNGARIAN_HUNGARY 0x040e
#define TT_MS_LANGID_ICELANDIC_ICELAND 0x040f
#define TT_MS_LANGID_ITALIAN_ITALY 0x0410
#define TT_MS_LANGID_JAPANESE_JAPAN 0x0411
#define TT_MS_LANGID_KOREAN_EXTENDED_WANSUNG_KOREA 0x0412
#define TT_MS_LANGID_DUTCH_NETHERLANDS 0x0413
#define TT_MS_LANGID_NORWEGIAN_NORWAY_BOKMAL 0x0414
#define TT_MS_LANGID_POLISH_POLAND 0x0415
#define TT_MS_LANGID_PORTUGUESE_BRAZIL 0x0416
#define TT_MS_LANGID_RHAETO_ROMANIC_SWITZERLAND 0x0417
#define TT_MS_LANGID_ROMANIAN_ROMANIA 0x0418
#define TT_MS_LANGID_RUSSIAN_RUSSIA 0x0419
#define TT_MS_LANGID_CROATIAN_CROATIA 0x041a
#define TT_MS_LANGID_SLOVAK_SLOVAKIA 0x041b
#define TT_MS_LANGID_ALBANIAN_ALBANIA 0x041c
#define TT_MS_LANGID_SWEDISH_SWEDEN 0x041d
#define TT_MS_LANGID_THAI_THAILAND 0x041e
#define TT_MS_LANGID_TURKISH_TURKEY 0x041f
#define TT_MS_LANGID_URDU_PAKISTAN 0x0420
#define TT_MS_LANGID_INDONESIAN_INDONESIA 0x0421
#define TT_MS_LANGID_UKRAINIAN_UKRAINE 0x0422
#define TT_MS_LANGID_BELARUSIAN_BELARUS 0x0423
#define TT_MS_LANGID_SLOVENE_SLOVENIA 0x0424
#define TT_MS_LANGID_ESTONIAN_ESTONIA 0x0425
#define TT_MS_LANGID_LATVIAN_LATVIA 0x0426
#define TT_MS_LANGID_LITHUANIAN_LITHUANIA 0x0427
#define TT_MS_LANGID_TAJIK_TAJIKISTAN 0x0428
#define TT_MS_LANGID_FARSI_IRAN 0x0429
#define TT_MS_LANGID_VIETNAMESE_VIET_NAM 0x042a
#define TT_MS_LANGID_ARMENIAN_ARMENIA 0x042b
#define TT_MS_LANGID_AZERI_AZERBAIJAN_LATIN 0x042c
#define TT_MS_LANGID_BASQUE_SPAIN 0x042d
#define TT_MS_LANGID_SERBIAN_GERMANY 0x042e
#define TT_MS_LANGID_MACEDONIAN_MACEDONIA 0x042f
#define TT_MS_LANGID_SUTU_SOUTH_AFRICA 0x0430
#define TT_MS_LANGID_TSONGA_SOUTH_AFRICA 0x0431
#define TT_MS_LANGID_TSWANA_SOUTH_AFRICA 0x0432
#define TT_MS_LANGID_VENDA_SOUTH_AFRICA 0x0433
#define TT_MS_LANGID_XHOSA_SOUTH_AFRICA 0x0434
#define TT_MS_LANGID_ZULU_SOUTH_AFRICA 0x0435
#define TT_MS_LANGID_AFRIKAANS_SOUTH_AFRICA 0x0436
#define TT_MS_LANGID_GEORGIAN_GEORGIA 0x0437
#define TT_MS_LANGID_FAEROESE_FAEROE_ISLANDS 0x0438
#define TT_MS_LANGID_HINDI_INDIA 0x0439
#define TT_MS_LANGID_MALTESE_MALTA 0x043a
#define TT_MS_LANGID_SAAMI_LAPONIA 0x043b
#define TT_MS_LANGID_SAMI_NORTHERN_NORWAY 0x043b
#define TT_MS_LANGID_IRISH_GAELIC_IRELAND 0x043c
#define TT_MS_LANGID_YIDDISH_GERMANY 0x043d
#define TT_MS_LANGID_MALAY_MALAYSIA 0x043e
#define TT_MS_LANGID_KAZAK_KAZAKSTAN 0x043f
#define TT_MS_LANGID_KIRGHIZ_KIRGHIZSTAN 0x0440
#define TT_MS_LANGID_SWAHILI_KENYA 0x0441
#define TT_MS_LANGID_TURKMEN_TURKMENISTAN 0x0442
#define TT_MS_LANGID_UZBEK_UZBEKISTAN_LATIN 0x0443
#define TT_MS_LANGID_TATAR_TATARSTAN 0x0444
#define TT_MS_LANGID_BENGALI_INDIA 0x0445
#define TT_MS_LANGID_PUNJABI_INDIA 0x0446
#define TT_MS_LANGID_GUJARATI_INDIA 0x0447

```

```

#define TT_MS_LANGID_ORIYA_INDIA 0x0448
#define TT_MS_LANGID_TAMIL_INDIA 0x0449
#define TT_MS_LANGID_TELUGU_INDIA 0x044a
#define TT_MS_LANGID_KANNADA_INDIA 0x044b
#define TT_MS_LANGID_MALAYALAM_INDIA 0x044c
#define TT_MS_LANGID_ASSAMESE_INDIA 0x044d
#define TT_MS_LANGID_MARATHI_INDIA 0x044e
#define TT_MS_LANGID_SANSKRIT_INDIA 0x044f
#define TT_MS_LANGID_MONGOLIAN_MONGOLIA 0x0450
#define TT_MS_LANGID_TIBETAN_CHINA 0x0451
#define TT_MS_LANGID_WELSH_WALES 0x0452
#define TT_MS_LANGID_KHMER_CAMBODIA 0x0453
#define TT_MS_LANGID_LAO_LAOS 0x0454
#define TT_MS_LANGID_BURMESE_MYANMAR 0x0455
#define TT_MS_LANGID_GALICIAN_SPAIN 0x0456
#define TT_MS_LANGID_KONKANI_INDIA 0x0457
#define TT_MS_LANGID_MANIPURI_INDIA 0x0458
#define TT_MS_LANGID_SINDHI_INDIA 0x0459
#define TT_MS_LANGID_SYRIAC_SYRIA 0x045a
#define TT_MS_LANGID_SINHALESE_SRI_LANKA 0x045b
#define TT_MS_LANGID_CHEROKEE_UNITED_STATES 0x045c
#define TT_MS_LANGID_INUKTITUT_CANADA 0x045d
#define TT_MS_LANGID_AMHARIC_ETHIOPIA 0x045e
#define TT_MS_LANGID_TAMAZIGHT_MOROCCO 0x045f
#define TT_MS_LANGID_KASHMIRI_PAKISTAN 0x0460
#define TT_MS_LANGID_NEPALI_NEPAL 0x0461
#define TT_MS_LANGID_FRISIAN_NETHERLANDS 0x0462
#define TT_MS_LANGID_PASHTO_AFGHANISTAN 0x0463
#define TT_MS_LANGID_FILIPINO_PHILIPPINES 0x0464
#define TT_MS_LANGID_DHIVEHI_MALDIVES 0x0465
#define TT_MS_LANGID_EDO_NIGERIA 0x0466
#define TT_MS_LANGID_FULFULDE_NIGERIA 0x0467
#define TT_MS_LANGID_HAUSA_NIGERIA 0x0468
#define TT_MS_LANGID_IBIBIO_NIGERIA 0x0469
#define TT_MS_LANGID_YORUBA_NIGERIA 0x046a
#define TT_MS_LANGID_QUECHUA_BOLIVIA 0x046b
#define TT_MS_LANGID_SEPEDI_SOUTH_AFRICA 0x046c
#define TT_MS_LANGID_IGBO_NIGERIA 0x0470
#define TT_MS_LANGID_KANURI_NIGERIA 0x0471
#define TT_MS_LANGID_OROMO_ETHIOPIA 0x0472
#define TT_MS_LANGID_TIGRIGNA_ETHIOPIA 0x0473
#define TT_MS_LANGID_GUARANI_PARAGUAY 0x0474
#define TT_MS_LANGID_HAWAIIAN_UNITED_STATES 0x0475
#define TT_MS_LANGID_LATIN 0x0476
#define TT_MS_LANGID_SOMALI_SOMALIA 0x0477
#define TT_MS_LANGID_YI_CHINA 0x0478
#define TT_MS_LANGID_PAPIAMENTU_NETHERLANDS_ANTILLES 0x0479
#define TT_MS_LANGID_UGHUR_CHINA 0x0480
#define TT_MS_LANGID_MAORI_NEW_ZEALAND 0x0481
#define TT_MS_LANGID_ARABIC_IRAQ 0x0801
#define TT_MS_LANGID_CHINESE_PRC 0x0804
#define TT_MS_LANGID_GERMAN_SWITZERLAND 0x0807
#define TT_MS_LANGID_ENGLISH_UNITED_KINGDOM 0x0809
#define TT_MS_LANGID_SPANISH_MEXICO 0x080a
#define TT_MS_LANGID_FRENCH_BELGIUM 0x080c
#define TT_MS_LANGID_ITALIAN_SWITZERLAND 0x0810
#define TT_MS_LANGID_KOREAN_JOHAB_KOREA 0x0812
#define TT_MS_LANGID_DUTCH_BELGIUM 0x0813
#define TT_MS_LANGID_NORWEGIAN_NORWAY_NYNORSK 0x0814
#define TT_MS_LANGID_PORTUGUESE_PORTUGAL 0x0816
#define TT_MS_LANGID_MOLDAVIAN_MOLDAVIA 0x0818
#define TT_MS_LANGID_RUSSIAN_MOLDAVIA 0x0819
#define TT_MS_LANGID_SERBIAN_SERBIA_LATIN 0x081a
#define TT_MS_LANGID_SWEDISH_FINLAND 0x081d
#define TT_MS_LANGID_URDU_INDIA 0x0820
#define TT_MS_LANGID_CLASSIC_LITHUANIAN_LITHUANIA 0x0827

```

```

#define TT_MS_LANGID_AZERI_AZERBAIJAN_CYRILLIC 0x082c
#define TT_MS_LANGID_SAMI_NORTHERN_SWEDEN 0x083b
#define TT_MS_LANGID_SCOTTISH_GAELIC_UNITED_KINGDOM 0x083c
#define TT_MS_LANGID_MALAY_BRUNEI_DARUSSALAM 0x083e
#define TT_MS_LANGID_UZBEK_UZBEKISTAN_CYRILLIC 0x0843
#define TT_MS_LANGID_BENGALI_BANGLADESH 0x0845
#define TT_MS_LANGID_PUNJABI_ARABIC_PAKISTAN 0x0846
#define TT_MS_LANGID_MONGOLIAN_MONGOLIA_MONGOLIAN 0x0850
#define TT_MS_LANGID_DZONGHKA_BHUTAN 0x0851
#define TT_MS_LANGID_SINDHI_PAKISTAN 0x0859
#define TT_MS_LANGID_TAMAZIGHT_MOROCCO_LATIN 0x085f
#define TT_MS_LANGID_KASHMIRI_SASIA 0x0860
#define TT_MS_LANGID_NEPALI_INDIA 0x0861
#define TT_MS_LANGID_QUECHUA_ECUADOR 0x086b
#define TT_MS_LANGID_TIGRIGNA_ERYTHREA 0x0873
#define TT_MS_LANGID_ARABIC_EGYPT 0x0c01
#define TT_MS_LANGID_CHINESE_HONG_KONG 0x0c04
#define TT_MS_LANGID_GERMAN_AUSTRIA 0x0c07
#define TT_MS_LANGID_ENGLISH_AUSTRALIA 0x0c09
#define TT_MS_LANGID_SPANISH_SPAIN_INTERNATIONAL_SORT 0x0c0a
#define TT_MS_LANGID_FRENCH_CANADA 0x0c0c
#define TT_MS_LANGID_SERBIAN_SERBIA_CYRILLIC 0x0c1a
#define TT_MS_LANGID_SAMI_NORTHERN_FINLAND 0x0c3b
#define TT_MS_LANGID_QUECHUA_PERU 0x0c6b
#define TT_MS_LANGID_ARABIC_LIBYA 0x1001
#define TT_MS_LANGID_CHINESE_SINGAPORE 0x1004
#define TT_MS_LANGID_GERMAN_LUXEMBOURG 0x1007
#define TT_MS_LANGID_ENGLISH_CANADA 0x1009
#define TT_MS_LANGID_SPANISH_GUATEMALA 0x100a
#define TT_MS_LANGID_FRENCH_SWITZERLAND 0x100c
#define TT_MS_LANGID_CROATIAN_BOSNIA_HERZEGOVINA 0x101a
#define TT_MS_LANGID_SAMI_LULE_NORWAY 0x103b
#define TT_MS_LANGID_ARABIC_ALGERIA 0x1401
#define TT_MS_LANGID_CHINESE_MACAU 0x1404
#define TT_MS_LANGID_GERMAN_LIECHTENSTEIN 0x1407
#define TT_MS_LANGID_ENGLISH_NEW_ZEALAND 0x1409
#define TT_MS_LANGID_SPANISH_COSTA_RICA 0x140a
#define TT_MS_LANGID_FRENCH_LUXEMBOURG 0x140c
#define TT_MS_LANGID_BOSNIAN_BOSNIA_HERZEGOVINA 0x141a
#define TT_MS_LANGID_SAMI_LULE_SWEDEN 0x143b
#define TT_MS_LANGID_ARABIC_MOROCCO 0x1801
#define TT_MS_LANGID_ENGLISH_IRELAND 0x1809
#define TT_MS_LANGID_SPANISH_PANAMA 0x180a
#define TT_MS_LANGID_FRENCH_MONACO 0x180c
#define TT_MS_LANGID_SERBIAN_BOSNIA_HERZ_CYRILLIC 0x181a
#define TT_MS_LANGID_SERBIAN_BOSNIA_HERZ_LATIN 0x181a
#define TT_MS_LANGID_SAMI_SOUTHERN_NORWAY 0x183b
#define TT_MS_LANGID_ARABIC_TUNISIA 0x1c01
#define TT_MS_LANGID_ENGLISH_SOUTH_AFRICA 0x1c09
#define TT_MS_LANGID_SPANISH_DOMINICAN_REPUBLIC 0x1c0a
#define TT_MS_LANGID_FRENCH_WEST_INDIES 0x1c0c
#define TT_MS_LANGID_SAMI_SOUTHERN_SWEDEN 0x1c3b
#define TT_MS_LANGID_ARABIC_OMAN 0x2001
#define TT_MS_LANGID_ENGLISH_JAMAICA 0x2009
#define TT_MS_LANGID_SPANISH_VENEZUELA 0x200a
#define TT_MS_LANGID_FRENCH_REUNION 0x200c
#define TT_MS_LANGID_SAMI_SKOLT_FINLAND 0x203b
#define TT_MS_LANGID_ARABIC_YEMEN 0x2401
#define TT_MS_LANGID_ENGLISH_CARIBBEAN 0x2409
#define TT_MS_LANGID_SPANISH_COLOMBIA 0x240a
#define TT_MS_LANGID_FRENCH_CONGO 0x240c
#define TT_MS_LANGID_SAMI_INARI_FINLAND 0x243b
#define TT_MS_LANGID_ARABIC_SYRIA 0x2801
#define TT_MS_LANGID_ENGLISH_BELIZE 0x2809
#define TT_MS_LANGID_SPANISH_PERU 0x280a
#define TT_MS_LANGID_FRENCH_SENEGAL 0x280c

```



```

#define TT_MS_LANGID_ARABIC_JORDAN 0x2c01
#define TT_MS_LANGID_ENGLISH_TRINIDAD 0x2c09
#define TT_MS_LANGID_SPANISH_ARGENTINA 0x2c0a
#define TT_MS_LANGID_FRENCH_CAMEROON 0x2c0c
#define TT_MS_LANGID_ARABIC_LEBANON 0x3001
#define TT_MS_LANGID_ENGLISH_ZIMBABWE 0x3009
#define TT_MS_LANGID_SPANISH_ECUADOR 0x300a
#define TT_MS_LANGID_FRENCH_COTE_D_IVOIRE 0x300c
#define TT_MS_LANGID_ARABIC_KUWAIT 0x3401
#define TT_MS_LANGID_ENGLISH_PHILIPPINES 0x3409
#define TT_MS_LANGID_SPANISH_CHILE 0x340a
#define TT_MS_LANGID_FRENCH_MALI 0x340c
#define TT_MS_LANGID_ARABIC_UAE 0x3801
#define TT_MS_LANGID_ENGLISH_INDONESIA 0x3809
#define TT_MS_LANGID_SPANISH_URUGUAY 0x380a
#define TT_MS_LANGID_FRENCH_MOROCCO 0x380c
#define TT_MS_LANGID_ARABIC_BAHRAIN 0x3c01
#define TT_MS_LANGID_ENGLISH_HONG_KONG 0x3c09
#define TT_MS_LANGID_SPANISH_PARAGUAY 0x3c0a
#define TT_MS_LANGID_FRENCH_HAITI 0x3c0c
#define TT_MS_LANGID_ARABIC_QATAR 0x4001
#define TT_MS_LANGID_ENGLISH_INDIA 0x4009
#define TT_MS_LANGID_SPANISH_BOLIVIA 0x400a
#define TT_MS_LANGID_ENGLISH_MALAYSIA 0x4409
#define TT_MS_LANGID_SPANISH_EL_SALVADOR 0x440a
#define TT_MS_LANGID_ENGLISH_SINGAPORE 0x4809
#define TT_MS_LANGID_SPANISH_HONDURAS 0x480a
#define TT_MS_LANGID_SPANISH_NICARAGUA 0x4c0a
#define TT_MS_LANGID_SPANISH_PUERTO_RICO 0x500a
#define TT_MS_LANGID_SPANISH_UNITED_STATES 0x540a
#define TT_MS_LANGID_SPANISH_LATIN_AMERICA 0xE40aU
#define TT_MS_LANGID_FRENCH_NORTH_AFRICA 0xE40cU
#define TT_ADOBE_ID_EXPERT 1
#define TT_APPLE_ID_UNICODE_1_1 1
#define TT_ISO_ID_10646 1
#define TT_MAC_ID_JAPANESE 1
#define TT_MAC_LANGID_FRENCH 1
#define TT_MS_ID_UNICODE_CS 1
#define TT_NAME_ID_FONT_FAMILY 1
#define TT_PLATFORM_MACINTOSH 1
#define TT_MAC_ID_GURMUKHI 10
#define TT_MAC_LANGID_HEBREW 10
#define TT_MS_ID_UCS_4 10
#define TT_NAME_ID_DESCRIPTION 10
#define TT_MAC_ID_GUJARATI 11
#define TT_MAC_LANGID_JAPANESE 11
#define TT_NAME_ID_VENDOR_URL 11
#define TT_MAC_ID_ORIYA 12
#define TT_MAC_LANGID_ARABIC 12
#define TT_NAME_ID_DESIGNER_URL 12
#define TT_MAC_LANGID_WELSH 128
#define TT_MAC_LANGID_BASQUE 129
#define TT_MAC_ID_BENGALI 13
#define TT_MAC_LANGID_FINNISH 13
#define TT_NAME_ID_LICENSE 13
#define TT_MAC_LANGID_CATALAN 130
#define TT_MAC_LANGID_LATIN 131
#define TT_MAC_LANGID_QUECHUA 132
#define TT_MAC_LANGID_GUARANI 133
#define TT_MAC_LANGID_AYMARA 134
#define TT_MAC_LANGID_TATAR 135
#define TT_MAC_LANGID_UGHUR 136
#define TT_MAC_LANGID_DZONGKHA 137
#define TT_MAC_LANGID_JAVANESE 138
#define TT_MAC_LANGID_SUNDANESE 139
#define TT_MAC_ID_TAMIL 14

```



```

#define TT_MAC_LANGID_GREEK      14
#define TT_NAME_ID_LICENSE_URL  14
#define TT_MAC_LANGID_GALICIAN  140
#define TT_MAC_LANGID_AFRIKAANS 141
#define TT_MAC_LANGID_BRETON     142
#define TT_MAC_LANGID_INUKTITUT  143
#define TT_MAC_LANGID_SCOTTISH_GAELIC 144
#define TT_MAC_LANGID_MANX_GAELIC 145
#define TT_MAC_LANGID_IRISH_GAELIC 146
#define TT_MAC_LANGID_TONGAN     147
#define TT_MAC_LANGID_GREEK_POLYTONIC 148
#define TT_MAC_LANGID_GREELANDIC 149
#define TT_MAC_ID_TELUGU         15
#define TT_MAC_LANGID_ICELANDIC 15
#define TT_MAC_LANGID_AZERBAIJANI_ROMAN_SCRIPT 150
#define TT_MAC_ID_KANNADA        16
#define TT_MAC_LANGID_MALTESE    16
#define TT_NAME_ID_PREFERRED_FAMILY 16
#define TT_MAC_ID_MALAYALAM      17
#define TT_MAC_LANGID_TURKISH    17
#define TT_NAME_ID_PREFERRED_SUBFAMILY 17
#define TT_MAC_ID_SINHALESE      18
#define TT_MAC_LANGID_CROATIAN   18
#define TT_NAME_ID_MAC_FULL_NAME 18
#define TT_MAC_ID_BURMESE        19
#define TT_MAC_LANGID_CHINESE_TRADITIONAL 19
#define TT_NAME_ID_SAMPLE_TEXT   19
#define TT_ADOBE_ID_CUSTOM        2
#define TT_APPLE_ID_ISO_10646     2
#define TT_ISO_ID_8859_1          2
#define TT_MAC_ID_TRADITIONAL_CHINESE 2
#define TT_MAC_LANGID_GERMAN     2
#define TT_MS_ID_SJIS             2
#define TT_NAME_ID_FONT_SUBFAMILY 2
#define TT_PLATFORM_ISO          2
#define TT_MAC_ID_KHMER           20
#define TT_MAC_LANGID_URDU        20
#define TT_NAME_ID_CID_FINDFONT_NAME 20
#define TT_MAC_ID_THAI            21
#define TT_MAC_LANGID_HINDI        21
#define TT_MAC_ID_LAOTIAN          22
#define TT_MAC_LANGID_THAI         22
#define TT_MAC_ID_GEORGIAN         23
#define TT_MAC_LANGID_KOREAN       23
#define TT_MAC_ID_ARMENIAN         24
#define TT_MAC_LANGID_LITHUANIAN   24
#define TT_MAC_ID_MALDIVIAN        25
#define TT_MAC_ID_SIMPLIFIED_CHINESE 25
#define TT_MAC_LANGID_POLISH       25
#define TT_MAC_ID_TIBETAN          26
#define TT_MAC_LANGID_HUNGARIAN    26
#define TT_MAC_ID_MONGOLIAN        27
#define TT_MAC_LANGID_ESTONIAN     27
#define TT_MAC_ID_GEEZ             28
#define TT_MAC_LANGID_LETTISH      28
#define TT_MAC_ID_SLAVIC           29
#define TT_MAC_LANGID_SAAMISK       29
#define TT_ADOBE_ID_LATIN_1        3
#define TT_APPLE_ID_UNICODE_2_0    3
#define TT_MAC_ID_KOREAN           3
#define TT_MAC_LANGID_ITALIAN      3
#define TT_MS_ID_GB2312            3
#define TT_NAME_ID_UNIQUE_ID       3
#define TT_PLATFORM_MICROSOFT      3
#define TT_MAC_ID_VIETNAMESE       30
#define TT_MAC_LANGID_FAEROESE     30

```

```

#define TT_MAC_ID_SINDHI 31
#define TT_MAC_LANGID_FARSI 31
#define TT_MAC_ID_UNINTERP 32
#define TT_MAC_LANGID_RUSSIAN 32
#define TT_MAC_LANGID_CHINESE_SIMPLIFIED 33
#define TT_MAC_LANGID_FLEMISH 34
#define TT_MAC_LANGID_IRISH 35
#define TT_MAC_LANGID_ALBANIAN 36
#define TT_MAC_LANGID_ROMANIAN 37
#define TT_MAC_LANGID_CZECH 38
#define TT_MAC_LANGID_SLOVAK 39
#define TT_APPLE_ID_UNICODE_32 4
#define TT_MAC_ID_ARABIC 4
#define TT_MAC_LANGID_DUTCH 4
#define TT_MS_ID_BIG_5 4
#define TT_NAME_ID_FULL_NAME 4
#define TT_PLATFORM_CUSTOM 4
#define TT_MAC_LANGID_SLOVENIAN 40
#define TT_MAC_LANGID_YIDDISH 41
#define TT_MAC_LANGID_SERBIAN 42
#define TT_MAC_LANGID_MACEDONIAN 43
#define TT_MAC_LANGID_BULGARIAN 44
#define TT_MAC_LANGID_UKRAINIAN 45
#define TT_MAC_LANGID_BYELORUSSIAN 46
#define TT_MAC_LANGID_UZBEK 47
#define TT_MAC_LANGID_KAZAKH 48
#define TT_MAC_LANGID_AZERBAIJANI 49
#define TT_MAC_LANGID_AZERBAIJANI_CYRILLIC_SCRIPT 49
#define TT_MAC_ID_HEBREW 5
#define TT_MAC_LANGID_SWEDISH 5
#define TT_MS_ID_WANSUNG 5
#define TT_NAME_ID_VERSION_STRING 5
#define TT_MAC_LANGID_AZERBAIJANI_ARABIC_SCRIPT 50
#define TT_MAC_LANGID_ARMENIAN 51
#define TT_MAC_LANGID_GEORGIAN 52
#define TT_MAC_LANGID_MOLDAVIAN 53
#define TT_MAC_LANGID_KIRGHIZ 54
#define TT_MAC_LANGID_TAJIKI 55
#define TT_MAC_LANGID_TURKMEN 56
#define TT_MAC_LANGID_MONGOLIAN 57
#define TT_MAC_LANGID_MONGOLIAN_MONGOLIAN_SCRIPT 57
#define TT_MAC_LANGID_MONGOLIAN_CYRILLIC_SCRIPT 58
#define TT_MAC_LANGID_PASHTO 59
#define TT_MAC_ID_GREEK 6
#define TT_MAC_LANGID_SPANISH 6
#define TT_MS_ID_JOHAB 6
#define TT_NAME_ID_PS_NAME 6
#define TT_MAC_LANGID_KURDISH 60
#define TT_MAC_LANGID_KASHMIRI 61
#define TT_MAC_LANGID_SINDHI 62
#define TT_MAC_LANGID_TIBETAN 63
#define TT_MAC_LANGID_NEPALI 64
#define TT_MAC_LANGID_SANSKRIT 65
#define TT_MAC_LANGID_MARATHI 66
#define TT_MAC_LANGID_BENGALI 67
#define TT_MAC_LANGID_ASSAMESE 68
#define TT_MAC_LANGID_GUJARATI 69
#define TT_MAC_ID_RUSSIAN 7
#define TT_MAC_LANGID_DANISH 7
#define TT_NAME_ID_TRADEMARK 7
#define TT_PLATFORM_ADOBE 7
#define TT_MAC_LANGID_PUNJABI 70
#define TT_MAC_LANGID_ORIYA 71
#define TT_MAC_LANGID_MALAYALAM 72
#define TT_MAC_LANGID_KANNADA 73
#define TT_MAC_LANGID_TAMIL 74

```

```

#define TT_MAC_LANGID_TELUGU 75
#define TT_MAC_LANGID_SINHALESE 76
#define TT_MAC_LANGID_BURMESE 77
#define TT_MAC_LANGID_KHMER 78
#define TT_MAC_LANGID_LAO 79
#define TT_MAC_ID_RSMBOL 8
#define TT_MAC_LANGID_PORTUGUESE 8
#define TT_NAME_ID_MANUFACTURER 8
#define TT_MAC_LANGID_VIETNAMESE 80
#define TT_MAC_LANGID_INDONESIAN 81
#define TT_MAC_LANGID_TAGALOG 82
#define TT_MAC_LANGID_MALAY_ROMAN_SCRIPT 83
#define TT_MAC_LANGID_MALAY_ARABIC_SCRIPT 84
#define TT_MAC_LANGID_AMHARIC 85
#define TT_MAC_LANGID_TIGRINYA 86
#define TT_MAC_LANGID_GALLA 87
#define TT_MAC_LANGID_SOMALI 88
#define TT_MAC_LANGID_SWAHILI 89
#define TT_MAC_ID_DEVANAGARI 9
#define TT_MAC_LANGID_NORWEGIAN 9
#define TT_NAME_ID_DESIGNER 9
#define TT_MAC_LANGID_RUANDA 90
#define TT_MAC_LANGID_RUNDI 91
#define TT_MAC_LANGID_CHEWA 92
#define TT_MAC_LANGID_MALAGASY 93
#define TT_MAC_LANGID_ESPERANTO 94
#define TT_MS_LANGID_DIVEHI_MALDIVES
TT_MS_LANGID_DHIVEHI_MALDIVES
#define TT_MS_LANGID_TIBETAN_BHUTAN
TT_MS_LANGID_DZONGHKA_BHUTAN
#define TT_MS_LANGID_FRENCH_ZAIRE TT_MS_LANGID_FRENCH_CONGO
#define TT_MS_LANGID_KASHMIRI_INDIA
TT_MS_LANGID_KASHMIRI_SASIA
#define TT_MS_LANGID_TIGRIGNA_ERYTHREA
TT_MS_LANGID_TIGRIGNA_ERYTHREA
#define TT_UCR_ARABIC_PRESENTATION_FORMS_A
TT_UCR_ARABIC_PRESENTATIONS_A
#define TT_UCR_ARABIC_PRESENTATION_FORMS_B
TT_UCR_ARABIC_PRESENTATIONS_B
#define TT_UCR_KANBUN TT_UCR_CJK_MISC
#define TT_UCR_COMBINING_DIACRITICAL_MARKS
TT_UCR_COMBINING_DIACRITICS

```

12.2.29 freetype/tttables.h

```

typedef enum {
    ft_sfnt_head = 0,
    ft_sfnt_maxp = 1,
    ft_sfnt_os2 = 2,
    ft_sfnt_hhea = 3,
    ft_sfnt_vhea = 4,
    ft_sfnt_post = 5,
    ft_sfnt_pclt = 6,
    sfnt_max = 7
} FT_Sfnt_Tag;

typedef struct TT_Header_ {
    FT_Fixed Table_Version;
    FT_Fixed Font_Revision;
    FT_Long CheckSum_Adjust;
    FT_Long Magic_Number;
    FT_UShort Flags;
    FT_UShort Units_Per_EM;
    FT_Long Created[2];
    FT_Long Modified[2];
    FT_Short xMin;

```

```

    FT_Short yMin;
    FT_Short xMax;
    FT_Short yMax;
    FT_UShort Mac_Style;
    FT_UShort Lowest_Rec_PPEM;
    FT_Short Font_Direction;
    FT_Short Index_To_Loc_Format;
    FT_Short Glyph_Data_Format;
} TT_Header;
typedef struct TT_HoriHeader_ {
    FT_Fixed Version;
    FT_Short Ascender;
    FT_Short Descender;
    FT_Short Line_Gap;
    FT_UShort advance_Width_Max;
    FT_Short min_Left_Side_Bearing;
    FT_Short min_Right_Side_Bearing;
    FT_Short xMax_Extent;
    FT_Short caret_Slope_Rise;
    FT_Short caret_Slope_Run;
    FT_Short caret_Offset;
    FT_Short Reserved[4];
    FT_Short metric_Data_Format;
    FT_UShort number_Of_HMetrics;
    void *long_metrics;
    void *short_metrics;
} TT_HoriHeader;
typedef struct TT_VertHeader_ {
    FT_Fixed Version;
    FT_Short Ascender;
    FT_Short Descender;
    FT_Short Line_Gap;
    FT_UShort advance_Height_Max;
    FT_Short min_Top_Side_Bearing;
    FT_Short min_Bottom_Side_Bearing;
    FT_Short yMax_Extent;
    FT_Short caret_Slope_Rise;
    FT_Short caret_Slope_Run;
    FT_Short caret_Offset;
    FT_Short Reserved[4];
    FT_Short metric_Data_Format;
    FT_UShort number_Of_VMetrics;
    void *long_metrics;
    void *short_metrics;
} TT_VertHeader;
typedef struct TT_OS2_ {
    FT_UShort version;
    FT_Short xAvgCharWidth;
    FT_UShort usWeightClass;
    FT_UShort usWidthClass;
    FT_Short fsType;
    FT_Short ySubscriptXSize;
    FT_Short ySubscriptYSize;
    FT_Short ySubscriptXOffset;
    FT_Short ySubscriptYOffset;
    FT_Short ySuperscriptXSize;
    FT_Short ySuperscriptYSize;
    FT_Short ySuperscriptXOffset;
    FT_Short ySuperscriptYOffset;
    FT_Short yStrikeoutSize;
    FT_Short yStrikeoutPosition;
    FT_Short sFamilyClass;
    FT_Byte panose[10];
    FT_ULong ulUnicodeRange1;
    FT_ULong ulUnicodeRange2;
    FT_ULong ulUnicodeRange3;

```

```

    FT_ULong ulUnicodeRange4;
    FT_Char achVendID[4];
    FT_UShort fsSelection;
    FT_UShort usFirstCharIndex;
    FT_UShort usLastCharIndex;
    FT_Short sTypoAscender;
    FT_Short sTypoDescender;
    FT_Short sTypoLineGap;
    FT_UShort usWinAscent;
    FT_UShort usWinDescent;
    FT_ULong ulCodePageRange1;
    FT_ULong ulCodePageRange2;
    FT_Short sxHeight;
    FT_Short sCapHeight;
    FT_UShort usDefaultChar;
    FT_UShort usBreakChar;
    FT_UShort usMaxContext;
} TT_OS2;
typedef struct TT_Postscript_ {
    FT_Fixed FormatType;
    FT_Fixed italicAngle;
    FT_Short underlinePosition;
    FT_Short underlineThickness;
    FT_ULong isFixedPitch;
    FT_ULong minMemType42;
    FT_ULong maxMemType42;
    FT_ULong minMemType1;
    FT_ULong maxMemType1;
} TT_Postscript;
typedef struct TT_PCLT_ {
    FT_Fixed Version;
    FT_ULong FontNumber;
    FT_UShort Pitch;
    FT_UShort xHeight;
    FT_UShort Style;
    FT_UShort TypeFamily;
    FT_UShort CapHeight;
    FT_UShort SymbolSet;
    FT_Char TypeFace[16];
    FT_Char CharacterComplement[8];
    FT_Char FileName[6];
    FT_Char StrokeWeight;
    FT_Char WidthType;
    FT_Byte SerifStyle;
    FT_Byte Reserved;
} TT_PCLT;
typedef struct TT_MaxProfile_ {
    FT_Fixed version;
    FT_UShort numGlyphs;
    FT_UShort maxPoints;
    FT_UShort maxContours;
    FT_UShort maxCompositePoints;
    FT_UShort maxCompositeContours;
    FT_UShort maxZones;
    FT_UShort maxTwilightPoints;
    FT_UShort maxStorage;
    FT_UShort maxFunctionDefs;
    FT_UShort maxInstructionDefs;
    FT_UShort maxStackElements;
    FT_UShort maxSizeOfInstructions;
    FT_UShort maxComponentElements;
    FT_UShort maxComponentDepth;
} TT_MaxProfile;
extern FT_ULong FT_Get_CMap_Language_ID(FT_CharMap);
extern void *FT_Get_Sfnt_Table(FT_Face, FT_Sfnt_Tag);

```

```

extern FT_Error FT_Load_Sfnt_Table(FT_Face, FT_ULong, FT_Long,
FT_Byte *,
                                FT_ULong *);
extern FT_Error FT_Sfnt_Table_Info(FT_Face, FT_UInt, FT_ULong *,
                                FT_ULong *);

```

12.2.30 freetype/tttags.h

```

#define TTAG_avar      FT_MAKE_TAG( 'a', 'v', 'a', 'r' )
#define TTAG_BASE     FT_MAKE_TAG( 'B', 'A', 'S', 'E' )
#define TTAG_bdat     FT_MAKE_TAG( 'b', 'd', 'a', 't' )
#define TTAG_bhed     FT_MAKE_TAG( 'b', 'h', 'e', 'd' )
#define TTAG_bloc     FT_MAKE_TAG( 'b', 'l', 'o', 'c' )
#define TTAG_CFF      FT_MAKE_TAG( 'C', 'F', 'F', ' ' )
#define TTAG_cmap     FT_MAKE_TAG( 'c', 'm', 'a', 'p' )
#define TTAG_cvar     FT_MAKE_TAG( 'c', 'v', 'a', 'r' )
#define TTAG_cvt      FT_MAKE_TAG( 'c', 'v', 't', ' ' )
#define TTAG_DSIG     FT_MAKE_TAG( 'D', 'S', 'I', 'G' )
#define TTAG_EBDT     FT_MAKE_TAG( 'E', 'B', 'D', 'T' )
#define TTAG_EBLC     FT_MAKE_TAG( 'E', 'B', 'L', 'C' )
#define TTAG_EBSC     FT_MAKE_TAG( 'E', 'B', 'S', 'C' )
#define TTAG_fpgm     FT_MAKE_TAG( 'f', 'p', 'g', 'm' )
#define TTAG_fvar     FT_MAKE_TAG( 'f', 'v', 'a', 'r' )
#define TTAG_gasp     FT_MAKE_TAG( 'g', 'a', 's', 'p' )
#define TTAG_GDEF     FT_MAKE_TAG( 'G', 'D', 'E', 'F' )
#define TTAG_glyf     FT_MAKE_TAG( 'g', 'l', 'y', 'f' )
#define TTAG_GPOS     FT_MAKE_TAG( 'G', 'P', 'O', 'S' )
#define TTAG_GSUB     FT_MAKE_TAG( 'G', 'S', 'U', 'B' )
#define TTAG_gvar     FT_MAKE_TAG( 'g', 'v', 'a', 'r' )
#define TTAG_hdmx     FT_MAKE_TAG( 'h', 'd', 'm', 'x' )
#define TTAG_head     FT_MAKE_TAG( 'h', 'e', 'a', 'd' )
#define TTAG_hhea     FT_MAKE_TAG( 'h', 'h', 'e', 'a' )
#define TTAG_hmtx     FT_MAKE_TAG( 'h', 'm', 't', 'x' )
#define TTAG_JSTF     FT_MAKE_TAG( 'J', 'S', 'T', 'F' )
#define TTAG_kern     FT_MAKE_TAG( 'k', 'e', 'r', 'n' )
#define TTAG_loca     FT_MAKE_TAG( 'l', 'o', 'c', 'a' )
#define TTAG_LTSH     FT_MAKE_TAG( 'L', 'T', 'S', 'H' )
#define TTAG_maxp     FT_MAKE_TAG( 'm', 'a', 'x', 'p' )
#define TTAG_MMFx     FT_MAKE_TAG( 'M', 'M', 'F', 'X' )
#define TTAG_MMSD     FT_MAKE_TAG( 'M', 'M', 'S', 'D' )
#define TTAG_name     FT_MAKE_TAG( 'n', 'a', 'm', 'e' )
#define TTAG_OS2      FT_MAKE_TAG( 'O', 'S', '/', '2' )
#define TTAG_OTTO     FT_MAKE_TAG( 'O', 'T', 'T', 'O' )
#define TTAG_PCLT     FT_MAKE_TAG( 'P', 'C', 'L', 'T' )
#define TTAG_post     FT_MAKE_TAG( 'p', 'o', 's', 't' )
#define TTAG_prep     FT_MAKE_TAG( 'p', 'r', 'e', 'p' )
#define TTAG_true     FT_MAKE_TAG( 't', 'r', 'u', 'e' )
#define TTAG_ttc      FT_MAKE_TAG( 't', 't', 'c', ' ' )
#define TTAG_ttcf     FT_MAKE_TAG( 't', 't', 'c', 'f' )
#define TTAG_VDMX     FT_MAKE_TAG( 'V', 'D', 'M', 'X' )
#define TTAG_vhea     FT_MAKE_TAG( 'v', 'h', 'e', 'a' )
#define TTAG_vmtx     FT_MAKE_TAG( 'v', 'm', 't', 'x' )

```

12.2.31 freetype/ttunpat.h

```

#define FT_PARAM_TAG_UNPATENTED_HINTING FT_MAKE_TAG( 'u', 'n', 'p',
'a' )

```

12.3 Interface Definitions for libfreetype

The interfaces defined on the following pages are included in libfreetype and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 12.1 shall behave as described in the referenced base document.

FT_Get_X11_Font_Format

Name

FT_Get_X11_Font_Format — Get font format

Synopsis

```
#include <freetype/ftxf86.h>
const char *FT_Get_X11_Font_Format(FT_Face face);
```

Description

The `FT_Get_X11_Font_Format()` function can be used to return a string describing the format of the font face referenced by *face*.

Note that this information is not needed normally; however, there are special cases (like in PDF devices) where it is important to differentiate, in spite of FreeType's uniform API.

Return Value

The `FT_Get_X11_Font_Format()` function returns a string describing the format of a given face, using values which can be used as an X11 `FONT_PROPERTY`. Possible values are TrueType, Type 1, BDF, PCF, Type 42, CID Type 1, CFF, PFR, and Windows FNT. In case of an error, `NULL` is returned.

VIII Xft library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

13 Libraries

13.1 Interfaces for libXft

Table 13-1 defines the library name and shared object name for the libXft library

Table 13-1 libXft Definition

Library:	libXft
SONAME:	libXft.so.2

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

13.1.1 X FreeType Library

13.1.1.1 Interfaces for X FreeType Library

An LSB conforming implementation shall provide the generic functions for X FreeType Library specified in Table 13-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 13-2 libXft - X FreeType Library Function Interfaces

XftCharExists [LSB]	XftCharFontSpecRender [LSB]	XftCharIndex [LSB]
XftCharSpecRender [LSB]	XftColorAllocName [LSB]	XftColorAllocValue [LSB]
XftColorFree [LSB]	XftDefaultHasRender [LSB]	XftDefaultSet [LSB]
XftDefaultSubstitute [LSB]	XftDrawChange [LSB]	XftDrawCharFontSpec [LSB]
XftDrawCharSpec [LSB]	XftDrawColormap [LSB]	XftDrawCreate [LSB]
XftDrawCreateAlpha [LSB]	XftDrawCreateBitmap [LSB]	XftDrawDestroy [LSB]
XftDrawDisplay [LSB]	XftDrawDrawable [LSB]	XftDrawGlyphFontSpec [LSB]
XftDrawGlyphSpec [LSB]	XftDrawGlyphs [LSB]	XftDrawPicture [LSB]
XftDrawRect [LSB]	XftDrawSetClip [LSB]	XftDrawSetClipRectangles [LSB]
XftDrawSetSubwindow Mode [LSB]	XftDrawSrcPicture [LSB]	XftDrawString16 [LSB]
XftDrawString32 [LSB]	XftDrawString8 [LSB]	XftDrawStringUtf16 [LSB]

XftDrawStringUtf8 [LSB]	XftDrawVisual [LSB]	XftFontCheckGlyph [LSB]
XftFontClose [LSB]	XftFontCopy [LSB]	XftFontInfoCreate [LSB]
XftFontInfoDestroy [LSB]	XftFontInfoEqual [LSB]	XftFontInfoHash [LSB]
XftFontLoadGlyphs [LSB]	XftFontMatch [LSB]	XftFontOpen [LSB]
XftFontOpenInfo [LSB]	XftFontOpenName [LSB]	XftFontOpenPattern [LSB]
XftFontOpenXlfd [LSB]	XftFontUnloadGlyphs [LSB]	XftGetVersion [LSB]
XftGlyphExtents [LSB]	XftGlyphFontSpecRender [LSB]	XftGlyphRender [LSB]
XftGlyphSpecRender [LSB]	XftInit [LSB]	XftInitFtLibrary [LSB]
XftListFonts [LSB]	XftLockFace [LSB]	XftNameParse [LSB]
XftTextExtents16 [LSB]	XftTextExtents32 [LSB]	XftTextExtents8 [LSB]
XftTextExtentsUtf16 [LSB]	XftTextExtentsUtf8 [LSB]	XftTextRender16 [LSB]
XftTextRender16BE [LSB]	XftTextRender16LE [LSB]	XftTextRender32 [LSB]
XftTextRender32BE [LSB]	XftTextRender32LE [LSB]	XftTextRender8 [LSB]
XftTextRenderUtf16 [LSB]	XftTextRenderUtf8 [LSB]	XftUnlockFace [LSB]
XftXlfdParse [LSB]		

13.2 Data Definitions for libXft

This section defines global identifiers and their values that are associated with interfaces contained in libXft. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

13.2.1 X11/Xft/Xft.h

```

#define _XFT_H_
#define XFT_VERSION \
    ((XFT_MAJOR * 10000) + (XFT_MINOR * 100) + (XFT_REVISION))
#define XFT_CORE "core"
#define XFT_MAX_GLYPH_MEMORY "maxglyphmemory"
#define XFT_MAX_UNREF_FONTS "maxunreffonts"
#define XFT_RENDER "render"
#define XFT_XLFD "xlfd"
#define XFT_MINOR 1
#define XFT_MAJOR 2
#define XFT_NMISSING 256
#define XFT_REVISION 8
#define XftVersion XFT_VERSION

typedef struct _XftDraw XftDraw;
typedef struct _XftColor {
    unsigned long int pixel;
    XRenderColor color;
} XftColor;
typedef struct _XftFont {
    int ascent;
    int descent;
    int height;
    int max_advance_width;
    FcCharSet *charset;
    FcPattern **pattern;
} XftFont;
typedef struct _XftGlyphSpec {
    FT_UInt glyph;
    short x;
    short y;
} XftGlyphSpec;
typedef struct _XftCharSpec {
    FcChar32 ucs4;
    short x;
    short y;
} XftCharSpec;
typedef struct _XftFtFile XftFtFile;
typedef struct _XftFontInfo XftFontInfo;
typedef struct _XftCharFontSpec {
    XftFont *font;
    FcChar32 ucs4;
    short x;
    short y;
} XftCharFontSpec;
typedef struct _XftGlyphFontSpec {
    XftFont *font;
    FT_UInt glyph;
    short x;
    short y;
} XftGlyphFontSpec;
extern FcBool XftCharExists(Display *, XftFont *, FcChar32);
extern void XftCharFontSpecRender(Display *, int, Picture, Picture,
int, int, const XftCharFontSpec *, int);
extern FT_UInt XftCharIndex(Display *, XftFont *, FcChar32);
extern void XftCharSpecRender(Display *, int, Picture, XftFont *,
Picture,
int, int, const XftCharSpec *, int);
extern int XftColorAllocName(Display *, const Visual *, Colormap,
const char *, XftColor *);
extern int XftColorAllocValue(Display *, Visual *, Colormap,
const XRenderColor *, XftColor *);

```

```

extern void XftColorFree(Display *, Visual *, Colormap, XftColor
*);
extern int XftDefaultHasRender(Display *);
extern int XftDefaultSet(Display *, FcPattern *);
extern void XftDefaultSubstitute(Display *, int, FcPattern *);
extern void XftDrawChange(XftDraw *, Drawable);
extern void XftDrawCharFontSpec(XftDraw *, const XftColor *,
                                const XftCharFontSpec *, int);
extern void XftDrawCharSpec(XftDraw *, const XftColor *, XftFont *,
                             const XftCharSpec *, int);
extern Colormap XftDrawColormap(XftDraw *);
extern XftDraw *XftDrawCreate(Display *, Drawable, Visual *,
Colormap);
extern XftDraw *XftDrawCreateAlpha(Display *, Pixmap, int);
extern XftDraw *XftDrawCreateBitmap(Display *, Pixmap);
extern void XftDrawDestroy(XftDraw *);
extern Display *XftDrawDisplay(XftDraw *);
extern Drawable XftDrawDrawable(XftDraw *);
extern void XftDrawGlyphFontSpec(XftDraw *, const XftColor *,
                                const XftGlyphFontSpec *, int);
extern void XftDrawGlyphSpec(XftDraw *, const XftColor *, XftFont
*,
                             const XftGlyphSpec *, int);
extern void XftDrawGlyphs(XftDraw *, const XftColor *, XftFont *,
int, int,
                             const FT_UInt *, int);
extern Picture XftDrawPicture(XftDraw *);
extern void XftDrawRect(XftDraw *, const XftColor *, int, int,
                        unsigned int, unsigned int);
extern int XftDrawSetClip(XftDraw *, Region);
extern int XftDrawSetClipRectangles(XftDraw *, int, int,
                                    const XRectangle *, int);
extern void XftDrawSetSubwindowMode(XftDraw *, int);
extern Picture XftDrawSrcPicture(XftDraw *, const XftColor *);
extern void XftDrawString16(XftDraw *, const XftColor *, XftFont *,
int,
                            int, const FcChar16 *, int);
extern void XftDrawString32(XftDraw *, const XftColor *, XftFont *,
int,
                            int, const FcChar32 *, int);
extern void XftDrawString8(XftDraw *, const XftColor *, XftFont *,
int,
                            int, const FcChar8 *, int);
extern void XftDrawStringUtf16(XftDraw *, const XftColor *, XftFont
*, int,
                              int, const FcChar8 *, FcEndian, int);
extern void XftDrawStringUtf8(XftDraw *, const XftColor *, XftFont
*, int,
                              int, const FcChar8 *, int);
extern Visual *XftDrawVisual(XftDraw *);
extern FcBool XftFontCheckGlyph(Display *, XftFont *, FcBool,
FT_UInt,
                            FT_UInt *, int *);
extern void XftFontClose(Display *, XftFont *);
extern XftFont *XftFontCopy(Display *, XftFont *);
extern XftFontInfo *XftFontInfoCreate(Display *, const FcPattern
*);
extern void XftFontInfoDestroy(Display *, XftFontInfo *);
extern FcBool XftFontInfoEqual(const XftFontInfo *, const
XftFontInfo *);
extern FcChar32 XftFontInfoHash(const XftFontInfo *);
extern void XftFontLoadGlyphs(Display *, XftFont *, FcBool,
                              const FT_UInt *, int);
extern FcPattern *XftFontMatch(Display *, int, const FcPattern *,
                              FcResult *);
extern XftFont *XftFontOpen(Display *, int, ...);

```

```

extern XftFont *XftFontOpenInfo(Display *, FcPattern *, XftFontInfo
*);
extern XftFont *XftFontOpenName(Display *, int, const char *);
extern XftFont *XftFontOpenPattern(Display *, FcPattern *);
extern XftFont *XftFontOpenXlfd(Display *, int, const char *);
extern void XftFontUnloadGlyphs(Display *, XftFont *, const FT_UInt
*,
                                int);
extern int XftGetVersion(void);
extern void XftGlyphExtents(Display *, XftFont *, const FT_UInt *,
int,
                                XGlyphInfo *);
extern void XftGlyphFontSpecRender(Display *, int, Picture, Picture,
int,
                                int, const XftGlyphFontSpec *, int);
extern void XftGlyphRender(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FT_UInt *, int);
extern void XftGlyphSpecRender(Display *, int, Picture, XftFont *,
Picture,
                                int, int, const XftGlyphSpec *, int);
extern int XftInit(const char *);
extern FcBool XftInitFtLibrary(void);
extern FcFontSet *XftListFonts(Display *, int, ...);
extern FT_Face XftLockFace(XftFont *);
extern FcPattern *XftNameParse(const char *);
extern void XftTextExtents16(Display *, XftFont *, const FcChar16
*, int,
                                XGlyphInfo *);
extern void XftTextExtents32(Display *, XftFont *, const FcChar32
*, int,
                                XGlyphInfo *);
extern void XftTextExtents8(Display *, XftFont *, const FcChar8 *,
int,
                                XGlyphInfo *);
extern void XftTextExtentsUtf16(Display *, XftFont *, const FcChar8
*,
                                FcEndian, int, XGlyphInfo *);
extern void XftTextExtentsUtf8(Display *, XftFont *, const FcChar8
*, int,
                                XGlyphInfo *);
extern void XftTextRender16(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar16 *, int);
extern void XftTextRender16BE(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *, int);
extern void XftTextRender16LE(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *, int);
extern void XftTextRender32(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar32 *, int);
extern void XftTextRender32BE(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *, int);
extern void XftTextRender32LE(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *, int);
extern void XftTextRender8(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *, int);
extern void XftTextRenderUtf16(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *,
                                FcEndian, int);

```

```

extern void XftTextRenderUtf8(Display *, int, Picture, XftFont *,
Picture,
                                int, int, int, int, const FcChar8 *, int);
extern void XftUnlockFace(XftFont *);
extern FcPattern *XftXlfdParse(const char *, FcBool, FcBool);

```

13.2.2 X11/Xft/XftCompat.h

```

#define XFT_ENCODING      "encoding"
#define XftConfigSubstitute(p)      FcConfigSubstitute      (0, p,
FcMatchPattern)
#define XftFontSetDestroy      FcFontSetDestroy
#define XftFontSetMatch      FcFontSetMatch
#define XftMatrixEqual      FcMatrixEqual
#define XftMatrixInit(m)      FcMatrixInit(m)
#define XftMatrixMultiply      FcMatrixMultiply
#define XftMatrixRotate      FcMatrixRotate
#define XftMatrixScale      FcMatrixScale
#define XftMatrixShear      FcMatrixShear
#define XftObjectSetAdd      FcObjectSetAdd
#define XftObjectSetBuild      FcObjectSetBuild
#define XftObjectSetCreate      FcObjectSetCreate
#define XftObjectSetDestroy      FcObjectSetDestroy
#define XftObjectSetVaBuild      FcObjectSetVaBuild
#define XftPatternAdd      FcPatternAdd
#define XftPatternAddBool      FcPatternAddBool
#define XftPatternAddDouble      FcPatternAddDouble
#define XftPatternAddInteger      FcPatternAddInteger
#define XftPatternAddMatrix      FcPatternAddMatrix
#define XftPatternAddString(p,e,s)      FcPatternAddString(p,e,s)
FcPatternAddString(p,e,(FcChar8 *)s))
#define XftPatternBuild      FcPatternBuild
#define XftPatternCreate      FcPatternCreate
#define XftPatternDel      FcPatternDel
#define XftPatternDestroy      FcPatternDestroy
#define XftPatternDuplicate      FcPatternDuplicate
#define XftPatternGet      FcPatternGet
#define XftPatternGetBool      FcPatternGetBool
#define XftPatternGetDouble      FcPatternGetDouble
#define XftPatternGetInteger      FcPatternGetInteger
#define XftPatternGetMatrix      FcPatternGetMatrix
#define XftPatternGetString(p,e,i,n)      FcPatternGetString(p,e,i,n)
FcPatternGetString(p,e,i,(FcChar8 **) (n))
#define XftPatternVaBuild      FcPatternVaBuild
#define XftResultMatch      FcResultMatch
#define XftResultNoId      FcResultNoId
#define XftResultNoMatch      FcResultNoMatch
#define XftResultTypeMismatch      FcResultTypeMismatch
#define XftTypeBool      FcTypeBool
#define XftTypeDouble      FcTypeDouble
#define XftTypeInteger      FcTypeInteger
#define XftTypeMatrix      FcTypeMatrix
#define XftTypeString      FcTypeString
#define XftTypeVoid      FcTypeVoid
#define XftUtf8Len      FcUtf8Len
#define XftUtf8ToUcs4      FcUtf8ToUcs4
#define XftValueDestroy      FcValueDestroy
#define XFT_ANTIALIAS      FC_ANTIALIAS
#define XFT_CHARCELL      FC_CHARCELL
#define XFT_CHAR_HEIGHT      FC_CHAR_HEIGHT
#define XFT_CHAR_WIDTH      FC_CHAR_WIDTH
#define XFT_DPI      FC_DPI
#define XFT_FAMILY      FC_FAMILY
#define XFT_FILE      FC_FILE
#define XFT_FOUNDRY      FC_FOUNDRY

```

```

#define XFT_INDEX          FC_INDEX
#define XFT_MATRIX         FC_MATRIX
#define XFT_MINSIZE        FC_MINSIZE
#define XFT_MONO           FC_MONO
#define XFT_OUTLINE        FC_OUTLINE
#define XFT_PIXEL_SIZE     FC_PIXEL_SIZE
#define XFT_PROPORTIONAL   FC_PROPORTIONAL
#define XFT_RASTERIZER     FC_RASTERIZER
#define XFT_RGBA           FC_RGBA
#define XFT_RGBA_BGR       FC_RGBA_BGR
#define XFT_RGBA_NONE      FC_RGBA_NONE
#define XFT_RGBA_RGB       FC_RGBA_RGB
#define XFT_RGBA_UNKNOWN   FC_RGBA_UNKNOWN
#define XFT_RGBA_VBGR      FC_RGBA_VBGR
#define XFT_RGBA_VRGB      FC_RGBA_VRGB
#define XFT_SCALABLE        FC_SCALABLE
#define XFT_SCALE           FC_SCALE
#define XFT_SIZE            FC_SIZE
#define XFT_SLANT           FC_SLANT
#define XFT_SLANT_ITALIC    FC_SLANT_ITALIC
#define XFT_SLANT_OBLIQUE   FC_SLANT_OBLIQUE
#define XFT_SLANT_ROMAN     FC_SLANT_ROMAN
#define XFT_SPACING         FC_SPACING
#define XFT_STYLE           FC_STYLE
#define XFT_WEIGHT          FC_WEIGHT
#define XFT_WEIGHT_BLACK    FC_WEIGHT_BLACK
#define XFT_WEIGHT_BOLD     FC_WEIGHT_BOLD
#define XFT_WEIGHT_DEMIBOLD FC_WEIGHT_DEMIBOLD
#define XFT_WEIGHT_LIGHT    FC_WEIGHT_LIGHT
#define XFT_WEIGHT_MEDIUM   FC_WEIGHT_MEDIUM
#define XftGlyphExists      XftCharExists

typedef unsigned char XftChar8;
typedef unsigned short XftChar16;
typedef unsigned int XftChar32;
typedef enum _FcType XftType;
typedef struct _FcMatrix XftMatrix;
typedef enum _FcResult XftResult;
typedef struct _FcValue XftValue;
typedef struct _FcPattern XftPattern;
typedef struct _FcFontSet XftFontSet;
typedef struct _FcObjectSet XftObjectSet;

```

13.3 Interface Definitions for libXft

The interfaces defined on the following pages are included in libXft and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 13.1 shall behave as described in the referenced base document.

IX Xrender library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

14 Libraries

14.1 Interfaces for libXrender

Table 14-1 defines the library name and shared object name for the libXrender library

Table 14-1 libXrender Definition

Library:	libXrender
SONAME:	libXrender.so.1

The behavior of the interfaces in this library is specified by the following specifications:

[LSB] This Specification

14.1.1 X Rendering Extension

14.1.1.1 Interfaces for X Rendering Extension

An LSB conforming implementation shall provide the generic functions for X Rendering Extension specified in Table 14-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 14-2 libXrender - X Rendering Extension Function Interfaces

XRenderAddGlyphs [LSB]	XRenderAddTraps [LSB]	XRenderChangePicture [LSB]
XRenderComposite [LSB]	XRenderCompositeDoublePoly [LSB]	XRenderCompositeString16 [LSB]
XRenderCompositeString32 [LSB]	XRenderCompositeString8 [LSB]	XRenderCompositeText16 [LSB]
XRenderCompositeText32 [LSB]	XRenderCompositeText8 [LSB]	XRenderCompositeTrapezoids [LSB]
XRenderCompositeTriFan [LSB]	XRenderCompositeTriStrip [LSB]	XRenderCompositeTriangles [LSB]
XRenderCreateAnimCursor [LSB]	XRenderCreateConicalGradient [LSB]	XRenderCreateCursor [LSB]
XRenderCreateGlyphSet [LSB]	XRenderCreateLinearGradient [LSB]	XRenderCreatePicture [LSB]
XRenderCreateRadialGradient [LSB]	XRenderCreateSolidFill [LSB]	XRenderFillRectangle [LSB]
XRenderFillRectangles [LSB]	XRenderFindFormat [LSB]	XRenderFindStandardFormat [LSB]
XRenderFindVisualFormat [LSB]	XRenderFreeGlyphSet [LSB]	XRenderFreeGlyphs [LSB]
XRenderFreePicture [LSB]	XRenderParseColor [LSB]	XRenderQueryExtension [LSB]

XRenderQueryFilters [LSB]	XRenderQueryFormats [LSB]	XRenderQueryPictIndexValues [LSB]
XRenderQuerySubpixelOrder [LSB]	XRenderQueryVersion [LSB]	XRenderReferenceGlyphSet [LSB]
XRenderSetPictureClipRectangles [LSB]	XRenderSetPictureClipRegion [LSB]	XRenderSetPictureFilter [LSB]
XRenderSetPictureTransform [LSB]	XRenderSetSubpixelOrder [LSB]	

14.2 Data Definitions for libXrender

This section defines global identifiers and their values that are associated with interfaces contained in libXrender. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

14.2.1 X11/extensions/Xrender.h

```
#define XFixedToDouble(f)      (((XDouble) (f)) / 65536)
#define XDoubleToFixed(f)     ((XFixed) ((f) * 65536))
#define PictFormatID          (1 << 0)
#define PictFormatType        (1 << 1)
#define PictFormatAlphaMask    (1 << 10)
#define PictFormatColormap     (1 << 11)
#define PictFormatDepth        (1 << 2)
#define PictFormatRed          (1 << 3)
#define PictFormatRedMask      (1 << 4)
#define PictFormatGreen        (1 << 5)
#define PictFormatGreenMask    (1 << 6)
#define PictFormatBlue         (1 << 7)
#define PictFormatBlueMask     (1 << 8)
#define PictFormatAlpha        (1 << 9)
#define PictStandardARGB32     0
#define PictStandardRGB24      1
#define PictStandardA8         2
#define PictStandardA4         3
#define PictStandardA1         4
#define PictStandardNUM        5

typedef long unsigned int GlyphSet;
typedef int XFixed;
typedef struct _XPointFixed {
    XFixed x;
    XFixed y;
} XPointFixed;
typedef struct _XCircle {
```

```

        XFixed x;
        XFixed y;
        XFixed radius;
    } XCircle;
typedef double XDouble;
typedef struct _XSpanFix {
    XFixed left;
    XFixed right;
    XFixed y;
} XSpanFix;
typedef struct _XLineFixed {
    XPointFixed p1;
    XPointFixed p2;
} XLineFixed;

typedef struct _XTrapezoid {
    XFixed top;
    XFixed bottom;
    XLineFixed left;
    XLineFixed right;
} XTrapezoid;
typedef struct {
    unsigned short red;
    unsigned short green;
    unsigned short blue;
    unsigned short alpha;
} XRenderColor;
typedef long unsigned int Glyph;
typedef long unsigned int PictFormat;
typedef struct {
    short red;
    short redMask;
    short green;
    short greenMask;
    short blue;
    short blueMask;
    short alpha;
    short alphaMask;
} XRenderDirectFormat;
typedef struct {
    PictFormat id;
    int type;
    int depth;
    XRenderDirectFormat direct;
    Colormap colormap;
} XRenderPictFormat;
typedef struct _XGlyphInfo {
    unsigned short width;
    unsigned short height;
    short x;
    short y;
    short xOff;
    short yOff;
} XGlyphInfo;
typedef struct _XFilters {
    int nfilter;
    char **filter;
    int nalias;
    short int *alias;
} XFilters;
typedef struct _XTransform {
    XFixed matrix[3][3];
} XTransform;
typedef struct _XAnimCursor {
    Cursor cursor;
    long unsigned int delay;

```

```

    } XAnimCursor;
typedef struct _XRadialGradient {
    XCircle inner;
    XCircle outer;
} XRadialGradient;
typedef struct _XGlyphElt32 {
    GlyphSet glyphset;
    const unsigned int *chars;
    int nchars;
    int xOff;
    int yOff;
} XGlyphElt32;
typedef struct _XGlyphElt8 {
    GlyphSet glyphset;
    const char *chars;
    int nchars;
    int xOff;
    int yOff;
} XGlyphElt8;
typedef struct _XConicalGradient {
    XPointFixed center;
    XFixed angle;
} XConicalGradient;
typedef struct _XLinearGradient {
    XPointFixed p1;
    XPointFixed p2;
} XLinearGradient;
typedef struct _XPointDouble {
    XDouble x;
    XDouble y;
} XPointDouble;
typedef struct _XRenderPictureAttributes {
    int repeat;
    Picture alpha_map;
    int alpha_x_origin;
    int alpha_y_origin;
    int clip_x_origin;
    int clip_y_origin;
    Pixmap clip_mask;
    int graphics_exposures;
    int subwindow_mode;
    int poly_edge;
    int poly_mode;
    Atom dither;
    int component_alpha;
} XRenderPictureAttributes;
typedef struct _XTrap {
    XSpanFix top;
    XSpanFix bottom;
} XTrap;
typedef struct _XIndexValue {
    long unsigned int pixel;
    unsigned short red;
    unsigned short green;
    unsigned short blue;
    unsigned short alpha;
} XIndexValue;
typedef struct _XGlyphElt16 {
    GlyphSet glyphset;
    const short unsigned int *chars;
    int nchars;
    int xOff;
    int yOff;
} XGlyphElt16;
typedef struct _XTriangle {
    XPointFixed p1;

```

```

    XPointFixed p2;
    XPointFixed p3;
} XTriangle;
extern void XRenderAddGlyphs(Display * dpy, GlyphSet glyphset,
                             const Glyph * gids, const XGlyphInfo *
glyphs,
                             int nglyphs, const char *images,
                             int nbyte_images);
extern void XRenderAddTraps(Display * dpy, Picture picture, int
xOff,
                             int yOff, const XTrap * traps, int ntrap);
extern void XRenderChangePicture(Display * dpy, Picture picture,
                                 long unsigned int valuemask,
                                 const XRenderPictureAttributes *
attributes);
extern void XRenderComposite(Display * dpy, int op, Picture src,
                             Picture mask, Picture dst, int src_x,
                             int src_y, int mask_x, int mask_y, int
dst_x,
                             int dst_y, unsigned int width,
                             unsigned int height);
extern void XRenderCompositeDoublePoly(Display * dpy, int op,
Picture src,
                                     Picture dst,
                                     const XRenderPictFormat *
maskFormat, int xSrc, int ySrc,
                                     int xDst, int yDst,
                                     const XPointDouble * fpoints,
                                     int npoints, int winding);
extern void XRenderCompositeString16(Display * dpy, int op, Picture
src,
                                     Picture dst,
                                     const XRenderPictFormat *
maskFormat,
                                     GlyphSet glyphset, int xSrc, int
ySrc,
                                     int xDst, int yDst,
                                     const short unsigned int *string,
                                     int nchar);
extern void XRenderCompositeString32(Display * dpy, int op, Picture
src,
                                     Picture dst,
                                     const XRenderPictFormat *
maskFormat,
                                     GlyphSet glyphset, int xSrc, int
ySrc,
                                     int xDst, int yDst,
                                     const unsigned int *string,
                                     int nchar);
extern void XRenderCompositeString8(Display * dpy, int op, Picture
src,
                                     Picture dst,
                                     const XRenderPictFormat * maskFormat,
                                     GlyphSet glyphset, int xSrc, int ySrc,
                                     int xDst, int yDst, const char
*string,
                                     int nchar);
extern void XRenderCompositeText16(Display * dpy, int op, Picture
src,
                                     Picture dst,
                                     const XRenderPictFormat * maskFormat,
                                     int xSrc, int ySrc, int xDst, int yDst,
                                     const XGlyphElt16 * elts, int nelt);
extern void XRenderCompositeText32(Display * dpy, int op, Picture
src,
                                     Picture dst,

```

```

        const XRenderPictFormat * maskFormat,
        int xSrc, int ySrc, int xDst, int yDst,
        const XGlyphElt32 * elts, int nelt);
extern void XRenderCompositeText8(Display * dpy, int op, Picture
src,
        Picture dst,
        const XRenderPictFormat * maskFormat,
        int xSrc, int ySrc, int xDst, int yDst,
        const XGlyphElt8 * elts, int nelt);
extern void XRenderCompositeTrapezoids(Display * dpy, int op,
Picture src,
        Picture dst,
        const XRenderPictFormat *
        maskFormat, int xSrc, int ySrc,
        const XTrapezoid * traps,
        int ntrap);
extern void XRenderCompositeTriFan(Display * dpy, int op, Picture
src,
        Picture dst,
        const XRenderPictFormat * maskFormat,
        int xSrc, int ySrc,
        const XPointFixed * points, int
npoint);
extern void XRenderCompositeTriStrip(Display * dpy, int op, Picture
src,
        Picture dst,
        const XRenderPictFormat *
maskFormat,
        int xSrc, int ySrc,
        const XPointFixed * points,
        int npoint);
extern void XRenderCompositeTriangles(Display * dpy, int op,
Picture src,
        Picture dst,
        const XRenderPictFormat *
maskFormat,
        int xSrc, int ySrc,
        const XTriangle * triangles,
        int ntriangle);
extern Cursor XRenderCreateAnimCursor(Display * dpy, int ncursor,
XAnimCursor * cursors);
extern Picture XRenderCreateConicalGradient(Display * dpy,
        const XConicalGradient *
        gradient, const XFixed * stops,
        const XRenderColor * colors,
        int nstops);
extern Cursor XRenderCreateCursor(Display * dpy, Picture source,
        unsigned int x, unsigned int y);
extern GlyphSet XRenderCreateGlyphSet(Display * dpy,
        const XRenderPictFormat * format);
extern Picture XRenderCreateLinearGradient(Display * dpy,
        const XLinearGradient *
        gradient, const XFixed * stops,
        const XRenderColor * colors,
        int nstops);
extern Picture XRenderCreatePicture(Display * dpy, Drawable
drawable,
        const XRenderPictFormat * format,
        long unsigned int valuemask,
        const XRenderPictureAttributes *
        attributes);
extern Picture XRenderCreateRadialGradient(Display * dpy,
        const XRadialGradient *
        gradient, const XFixed * stops,
        const XRenderColor * colors,
        int nstops);

```

```

extern Picture XRenderCreateSolidFill(Display * dpy,
                                     const XRenderColor * color);
extern void XRenderFillRectangle(Display * dpy, int op, Picture dst,
                                const XRenderColor * color, int x, int
Y,
                                unsigned int width, unsigned int
height);
extern void XRenderFillRectangles(Display * dpy, int op, Picture
dst,
                                const XRenderColor * color,
                                const XRectangle * rectangles,
                                int n_rects);
extern XRenderPictFormat *XRenderFindFormat(Display * dpy,
                                             long unsigned int mask,
                                             const XRenderPictFormat *
templ, int count);
extern XRenderPictFormat *XRenderFindStandardFormat(Display * dpy,
                                                    int format);
extern XRenderPictFormat *XRenderFindVisualFormat(Display * dpy,
                                                  const Visual * visual);
extern void XRenderFreeGlyphSet(Display * dpy, GlyphSet glyphset);
extern void XRenderFreeGlyphs(Display * dpy, GlyphSet glyphset,
                              const Glyph * gids, int nglyphs);
extern void XRenderFreePicture(Display * dpy, Picture picture);
extern int XRenderParseColor(Display * dpy, char *spec,
                             XRenderColor * def);
extern int XRenderQueryExtension(Display * dpy, int *event_basep,
                                int *error_basep);
extern XFilters *XRenderQueryFilters(Display * dpy, Drawable
drawable);
extern int XRenderQueryFormats(Display * dpy);
extern XIndexValue *XRenderQueryPictIndexValues(Display * dpy,
                                                const XRenderPictFormat *
format, int *num);
extern int XRenderQuerySubpixelOrder(Display * dpy, int screen);
extern int XRenderQueryVersion(Display * dpy, int *major_versionp,
                               int *minor_versionp);
extern GlyphSet XRenderReferenceGlyphSet(Display * dpy, GlyphSet
existing);
extern void XRenderSetPictureClipRectangles(Display * dpy, Picture
picture,
                                             int xOrigin, int yOrigin,
                                             const XRectangle * rects,
                                             int n);
extern void XRenderSetPictureClipRegion(Display * dpy, Picture
picture,
                                         Region r);
extern void XRenderSetPictureFilter(Display * dpy, Picture picture,
                                   const char *filter, XFixed * params,
                                   int nparams);
extern void XRenderSetPictureTransform(Display * dpy, Picture
picture,
                                       XTransform * transform);
extern int XRenderSetSubpixelOrder(Display * dpy, int screen,
                                   int subpixel);

```

14.2.2 X11/extensions/render.h

```

#define FilterBest      "best"
#define FilterBilinear  "bilinear"
#define FilterConvolution "convolution"
#define FilterFast      "fast"
#define FilterGood      "good"
#define FilterNearest   "nearest"
#define RENDER_NAME     "RENDER"

```

```

#define CPRepeat          (1 << 0)
#define CPAlphaMap        (1 << 1)
#define CPPolyMode        (1 << 10)
#define CPDither          (1 << 11)
#define CPComponentAlpha  (1 << 12)
#define CPAlphaXOrigin    (1 << 2)
#define CPAlphaYOrigin    (1 << 3)
#define CPClipXOrigin     (1 << 4)
#define CPClipYOrigin     (1 << 5)
#define CPClipMask        (1 << 6)
#define CPGraphicsExposure (1 << 7)
#define CPSubwindowMode   (1 << 8)
#define CPPolyEdge        (1 << 9)
#define RenderNumberErrors (BadGlyph+1)
#define RenderNumberRequests (X_RenderCreateConicalGradient+1)
#define FilterAliasNone -1
#define BadPictFormat 0
#define PictOpClear 0
#define PictOpMinimum 0
#define PictTypeIndexed 0
#define PolyEdgeSharp 0
#define PolyModePrecise 0
#define RENDER_MAJOR 0
#define RepeatNone 0
#define SubPixelUnknown 0
#define X_RenderQueryVersion 0
#define PictOpDisjointClear 0x10
#define PictOpDisjointMinimum 0x10
#define PictOpDisjointSrc 0x11
#define PictOpDisjointDst 0x12
#define PictOpDisjointOver 0x13
#define PictOpDisjointOverReverse 0x14
#define PictOpDisjointIn 0x15
#define PictOpDisjointInReverse 0x16
#define PictOpDisjointOut 0x17
#define PictOpDisjointOutReverse 0x18
#define PictOpDisjointAtop 0x19
#define PictOpDisjointAtopReverse 0x1a
#define PictOpDisjointMaximum 0x1b
#define PictOpDisjointXor 0x1b
#define PictOpConjointClear 0x20
#define PictOpConjointMinimum 0x20
#define PictOpConjointSrc 0x21
#define PictOpConjointDst 0x22
#define PictOpConjointOver 0x23
#define PictOpConjointOverReverse 0x24
#define PictOpConjointIn 0x25
#define PictOpConjointInReverse 0x26
#define PictOpConjointOut 0x27
#define PictOpConjointOutReverse 0x28
#define PictOpConjointAtop 0x29
#define PictOpConjointAtopReverse 0x2a
#define PictOpConjointMaximum 0x2b
#define PictOpConjointXor 0x2b
#define BadPicture 1
#define PictOpSrc 1
#define PictTypeDirect 1
#define PolyEdgeSmooth 1
#define PolyModeImprecise 1
#define RepeatNormal 1
#define SubPixelHorizontalRGB 1
#define X_RenderQueryPictFormats 1
#define PictOpAtopReverse 10
#define RENDER_MINOR 10
#define X_RenderTrapezoids 10
#define PictOpXor 11

```



```

#define X_RenderTriangles      11
#define CPLastBit              12
#define PictOpAdd              12
#define X_RenderTriStrip      12
#define PictOpMaximum          13
#define PictOpSaturate         13
#define X_RenderTriFan         13
#define X_RenderColorTrapezoids 14
#define X_RenderColorTriangles 15
#define X_RenderCreateGlyphSet 17
#define X_RenderReferenceGlyphSet 18
#define X_RenderFreeGlyphSet   19
#define BadPictOp              2
#define PictOpDst              2
#define RepeatPad              2
#define SubPixelHorizontalBGR  2
#define X_RenderQueryPictIndexValues 2
#define X_RenderAddGlyphs      20
#define X_RenderAddGlyphsFromPicture 21
#define X_RenderFreeGlyphs     22
#define X_RenderCompositeGlyphs8 23
#define X_RenderCompositeGlyphs16 24
#define X_RenderCompositeGlyphs32 25
#define X_RenderFillRectangles 26
#define X_RenderCreateCursor   27
#define X_RenderSetPictureTransform 28
#define X_RenderQueryFilters   29
#define BadGlyphSet            3
#define PictOpOver             3
#define RepeatReflect          3
#define SubPixelVerticalRGB    3
#define X_RenderQueryDithers   3
#define X_RenderSetPictureFilter 30
#define X_RenderCreateAnimCursor 31
#define X_RenderAddTraps       32
#define X_RenderCreateSolidFill 33
#define X_RenderCreateLinearGradient 34
#define X_RenderCreateRadialGradient 35
#define X_RenderCreateConicalGradient 36
#define BadGlyph              4
#define PictOpOverReverse      4
#define SubPixelVerticalBGR    4
#define X_RenderCreatePicture  4
#define PictOpIn               5
#define SubPixelNone           5
#define X_RenderChangePicture  5
#define PictOpInReverse        6
#define X_RenderSetPictureClipRectangles 6
#define PictOpOut              7
#define X_RenderFreePicture     7
#define PictOpOutReverse       8
#define X_RenderComposite      8
#define PictOpAtop             9
#define X_RenderScale          9

typedef long unsigned int Picture;

```

14.3 Interface Definitions for libXrender

The interfaces defined on the following pages are included in libXrender and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 14.1 shall behave as described in the referenced base document.

X Cairo Vector Graphics library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

15 Libraries

15.1 Interfaces for libcairo

Table 15-1 defines the library name and shared object name for the libcairo library

Table 15-1 libcairo Definition

Library:	libcairo
SONAME:	libcairo.so.2

The behavior of the interfaces in this library is specified by the following specifications:

[Cairo 1.12.4] Cairo API Reference

[LSB] This Specification

15.1.1 Cairo Vector Graphics Library

15.1.1.1 Interfaces for Cairo Vector Graphics Library

An LSB conforming implementation shall provide the generic functions for Cairo Vector Graphics Library specified in Table 15-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 15-2 libcairo - Cairo Vector Graphics Library Function Interfaces

cairo_append_path [Cairo 1.12.4]	cairo_arc [Cairo 1.12.4]	cairo_arc_negative [Cairo 1.12.4]
cairo_clip [Cairo 1.12.4]	cairo_clip_preserve [Cairo 1.12.4]	cairo_close_path [Cairo 1.12.4]
cairo_copy_page [Cairo 1.12.4]	cairo_copy_path [Cairo 1.12.4]	cairo_copy_path_flat [Cairo 1.12.4]
cairo_create [Cairo 1.12.4]	cairo_curve_to [Cairo 1.12.4]	cairo_destroy [Cairo 1.12.4]
cairo_device_acquire [Cairo 1.12.4]	cairo_device_destroy [Cairo 1.12.4]	cairo_device_finish [Cairo 1.12.4]
cairo_device_flush [Cairo 1.12.4]	cairo_device_get_refere nce_count [Cairo 1.12.4]	cairo_device_get_type [Cairo 1.12.4]
cairo_device_get_user_ data [Cairo 1.12.4]	cairo_device_observer_ elapsed [LSB]	cairo_device_observer_f ill_elapsed [LSB]
cairo_device_observer_ glyphs_elapsed [LSB]	cairo_device_observer_ mask_elapsed [LSB]	cairo_device_observer_ paint_elapsed [LSB]
cairo_device_observer_ print [LSB]	cairo_device_observer_ stroke_elapsed [LSB]	cairo_device_reference [Cairo 1.12.4]
cairo_device_release [Cairo 1.12.4]	cairo_device_set_user_ data [Cairo 1.12.4]	cairo_device_status [Cairo 1.12.4]
cairo_device_to_user [Cairo 1.12.4]	cairo_device_to_user_di stance [Cairo 1.12.4]	cairo_fill [Cairo 1.12.4]

cairo_fill_extents [Cairo 1.12.4]	cairo_fill_preserve [Cairo 1.12.4]	cairo_font_extents [Cairo 1.12.4]
cairo_font_face_destroy [Cairo 1.12.4]	cairo_font_face_get_type [Cairo 1.12.4]	cairo_font_face_get_user_data [Cairo 1.12.4]
cairo_font_face_reference [Cairo 1.12.4]	cairo_font_face_set_user_data [Cairo 1.12.4]	cairo_font_face_status [Cairo 1.12.4]
cairo_font_options_copy [Cairo 1.12.4]	cairo_font_options_create [Cairo 1.12.4]	cairo_font_options_destroy [Cairo 1.12.4]
cairo_font_options_equal [Cairo 1.12.4]	cairo_font_options_get_antialias [Cairo 1.12.4]	cairo_font_options_get_hint_metrics [Cairo 1.12.4]
cairo_font_options_get_hint_style [Cairo 1.12.4]	cairo_font_options_get_subpixel_order [Cairo 1.12.4]	cairo_font_options_has_h [Cairo 1.12.4]
cairo_font_options_merge [Cairo 1.12.4]	cairo_font_options_set_antialias [Cairo 1.12.4]	cairo_font_options_set_hint_metrics [Cairo 1.12.4]
cairo_font_options_set_hint_style [Cairo 1.12.4]	cairo_font_options_set_subpixel_order [Cairo 1.12.4]	cairo_font_options_status [Cairo 1.12.4]
cairo_format_stride_for_width [Cairo 1.12.4]	cairo_ft_font_face_create_for_ft_face [Cairo 1.12.4]	cairo_ft_font_face_create_for_pattern [Cairo 1.12.4]
cairo_ft_font_face_get_synthesize [Cairo 1.12.4]	cairo_ft_font_face_set_synthesize [Cairo 1.12.4]	cairo_ft_font_face_unset_synthesize [Cairo 1.12.4]
cairo_ft_font_options_substitute [Cairo 1.12.4]	cairo_ft_scaled_font_lock_face [Cairo 1.12.4]	cairo_ft_scaled_font_unlock_face [Cairo 1.12.4]
cairo_get_antialias [Cairo 1.12.4]	cairo_get_current_point [Cairo 1.12.4]	cairo_get_fill_rule [Cairo 1.12.4]
cairo_get_font_face [Cairo 1.12.4]	cairo_get_font_matrix [Cairo 1.12.4]	cairo_get_font_options [Cairo 1.12.4]
cairo_get_group_target [Cairo 1.12.4]	cairo_get_line_cap [Cairo 1.12.4]	cairo_get_line_join [Cairo 1.12.4]
cairo_get_line_width [Cairo 1.12.4]	cairo_get_matrix [Cairo 1.12.4]	cairo_get_miter_limit [Cairo 1.12.4]
cairo_get_operator [Cairo 1.12.4]	cairo_get_source [Cairo 1.12.4]	cairo_get_target [Cairo 1.12.4]
cairo_get_tolerance [Cairo 1.12.4]	cairo_glyph_allocate [Cairo 1.12.4]	cairo_glyph_extents [Cairo 1.12.4]
cairo_glyph_free [Cairo 1.12.4]	cairo_glyph_path [Cairo 1.12.4]	cairo_has_current_point [Cairo 1.12.4]

cairo_identity_matrix [Cairo 1.12.4]	cairo_image_surface_create [Cairo 1.12.4]	cairo_image_surface_create_for_data [Cairo 1.12.4]
cairo_image_surface_create_from_png [Cairo 1.12.4]	cairo_image_surface_create_from_png_stream [Cairo 1.12.4]	cairo_image_surface_get_data [Cairo 1.12.4]
cairo_image_surface_get_format [Cairo 1.12.4]	cairo_image_surface_get_height [Cairo 1.12.4]	cairo_image_surface_get_stride [Cairo 1.12.4]
cairo_image_surface_get_width [Cairo 1.12.4]	cairo_in_clip [Cairo 1.12.4]	cairo_in_fill [Cairo 1.12.4]
cairo_in_stroke [Cairo 1.12.4]	cairo_line_to [Cairo 1.12.4]	cairo_mask [Cairo 1.12.4]
cairo_mask_surface [Cairo 1.12.4]	cairo_matrix_init [Cairo 1.12.4]	cairo_matrix_init_identity [Cairo 1.12.4]
cairo_matrix_init_rotate [Cairo 1.12.4]	cairo_matrix_init_scale [Cairo 1.12.4]	cairo_matrix_init_translate [Cairo 1.12.4]
cairo_matrix_invert [Cairo 1.12.4]	cairo_matrix_multiply [Cairo 1.12.4]	cairo_matrix_rotate [Cairo 1.12.4]
cairo_matrix_scale [Cairo 1.12.4]	cairo_matrix_transform_distance [Cairo 1.12.4]	cairo_matrix_transform_point [Cairo 1.12.4]
cairo_matrix_translate [Cairo 1.12.4]	cairo_mesh_pattern_begin_patch [Cairo 1.12.4]	cairo_mesh_pattern_curve_to [Cairo 1.12.4]
cairo_mesh_pattern_end_patch [Cairo 1.12.4]	cairo_mesh_pattern_get_control_point [Cairo 1.12.4]	cairo_mesh_pattern_get_corner_color_rgba [Cairo 1.12.4]
cairo_mesh_pattern_get_patch_count [Cairo 1.12.4]	cairo_mesh_pattern_get_path [Cairo 1.12.4]	cairo_mesh_pattern_line_to [Cairo 1.12.4]
cairo_mesh_pattern_move_to [Cairo 1.12.4]	cairo_mesh_pattern_set_control_point [Cairo 1.12.4]	cairo_mesh_pattern_set_corner_color_rgb [Cairo 1.12.4]
cairo_mesh_pattern_set_corner_color_rgba [Cairo 1.12.4]	cairo_move_to [Cairo 1.12.4]	cairo_new_path [Cairo 1.12.4]
cairo_new_sub_path [Cairo 1.12.4]	cairo_paint [Cairo 1.12.4]	cairo_paint_with_alpha [Cairo 1.12.4]
cairo_path_destroy [Cairo 1.12.4]	cairo_path_extents [Cairo 1.12.4]	cairo_pattern_add_color_stop_rgb [Cairo 1.12.4]
cairo_pattern_add_color_stop_rgba [Cairo 1.12.4]	cairo_pattern_create_for_surface [Cairo 1.12.4]	cairo_pattern_create_linear [Cairo 1.12.4]

cairo_pattern_create_mesh [Cairo 1.12.4]	cairo_pattern_create_radial [Cairo 1.12.4]	cairo_pattern_create_raster_source [Cairo 1.12.4]
cairo_pattern_create_rgb [Cairo 1.12.4]	cairo_pattern_create_rgba [Cairo 1.12.4]	cairo_pattern_destroy [Cairo 1.12.4]
cairo_pattern_get_extents [Cairo 1.12.4]	cairo_pattern_get_filter [Cairo 1.12.4]	cairo_pattern_get_matrix [Cairo 1.12.4]
cairo_pattern_get_type [Cairo 1.12.4]	cairo_pattern_reference [Cairo 1.12.4]	cairo_pattern_set_extents [Cairo 1.12.4]
cairo_pattern_set_filter [Cairo 1.12.4]	cairo_pattern_set_matrix [Cairo 1.12.4]	cairo_pattern_status [Cairo 1.12.4]
cairo_pdf_get_versions [Cairo 1.12.4]	cairo_pdf_surface_create [Cairo 1.12.4]	cairo_pdf_surface_create_for_stream [Cairo 1.12.4]
cairo_pdf_surface_restrict_to_version [Cairo 1.12.4]	cairo_pdf_surface_set_size [Cairo 1.12.4]	cairo_pdf_version_to_string [Cairo 1.12.4]
cairo_pop_group [Cairo 1.12.4]	cairo_pop_group_to_source [Cairo 1.12.4]	cairo_ps_get_levels [Cairo 1.12.4]
cairo_ps_level_to_string [Cairo 1.12.4]	cairo_ps_surface_create [Cairo 1.12.4]	cairo_ps_surface_create_for_stream [Cairo 1.12.4]
cairo_ps_surface_dsc_begin_page_setup [Cairo 1.12.4]	cairo_ps_surface_dsc_begin_setup [Cairo 1.12.4]	cairo_ps_surface_dsc_comment [Cairo 1.12.4]
cairo_ps_surface_get_eps [Cairo 1.12.4]	cairo_ps_surface_restrict_to_level [Cairo 1.12.4]	cairo_ps_surface_set_eps [Cairo 1.12.4]
cairo_ps_surface_set_size [Cairo 1.12.4]	cairo_push_group [Cairo 1.12.4]	cairo_push_group_with_content [Cairo 1.12.4]
cairo_raster_source_pattern_get_acquire [Cairo 1.12.4]	cairo_raster_source_pattern_get_callback_data [Cairo 1.12.4]	cairo_raster_source_pattern_get_copy [Cairo 1.12.4]
cairo_raster_source_pattern_get_finish [Cairo 1.12.4]	cairo_raster_source_pattern_get_snapshot [Cairo 1.12.4]	cairo_raster_source_pattern_set_acquire [Cairo 1.12.4]
cairo_raster_source_pattern_set_callback_data [Cairo 1.12.4]	cairo_raster_source_pattern_set_copy [Cairo 1.12.4]	cairo_raster_source_pattern_set_finish [Cairo 1.12.4]
cairo_raster_source_pattern_set_snapshot [Cairo 1.12.4]	cairo_recording_surface_create [Cairo 1.12.4]	cairo_recording_surface_get_extents [Cairo 1.12.4]
cairo_recording_surface_ink_extents [Cairo 1.12.4]	cairo_rectangle [Cairo 1.12.4]	cairo_reference [Cairo 1.12.4]

cairo_region_contains_point [Cairo 1.12.4]	cairo_region_contains_rectangle [Cairo 1.12.4]	cairo_region_copy [Cairo 1.12.4]
cairo_region_create [Cairo 1.12.4]	cairo_region_create_rectangle [Cairo 1.12.4]	cairo_region_create_rectangles [Cairo 1.12.4]
cairo_region_destroy [Cairo 1.12.4]	cairo_region_equal [Cairo 1.12.4]	cairo_region_get_extents [Cairo 1.12.4]
cairo_region_get_rectangle [Cairo 1.12.4]	cairo_region_intersect [Cairo 1.12.4]	cairo_region_intersect_rectangle [Cairo 1.12.4]
cairo_region_is_empty [Cairo 1.12.4]	cairo_region_num_rectangles [Cairo 1.12.4]	cairo_region_reference [Cairo 1.12.4]
cairo_region_status [Cairo 1.12.4]	cairo_region_subtract [Cairo 1.12.4]	cairo_region_subtract_rectangle [Cairo 1.12.4]
cairo_region_translate [Cairo 1.12.4]	cairo_region_union [Cairo 1.12.4]	cairo_region_union_rectangle [Cairo 1.12.4]
cairo_region_xor [Cairo 1.12.4]	cairo_region_xor_rectangle [Cairo 1.12.4]	cairo_rel_curve_to [Cairo 1.12.4]
cairo_rel_line_to [Cairo 1.12.4]	cairo_rel_move_to [Cairo 1.12.4]	cairo_reset_clip [Cairo 1.12.4]
cairo_restore [Cairo 1.12.4]	cairo_rotate [Cairo 1.12.4]	cairo_save [Cairo 1.12.4]
cairo_scale [Cairo 1.12.4]	cairo_scaled_font_create [Cairo 1.12.4]	cairo_scaled_font_destroy [Cairo 1.12.4]
cairo_scaled_font_extents [Cairo 1.12.4]	cairo_scaled_font_get_ctm [Cairo 1.12.4]	cairo_scaled_font_get_font_face [Cairo 1.12.4]
cairo_scaled_font_get_font_matrix [Cairo 1.12.4]	cairo_scaled_font_get_font_options [Cairo 1.12.4]	cairo_scaled_font_get_scale_matrix [Cairo 1.12.4]
cairo_scaled_font_get_type [Cairo 1.12.4]	cairo_scaled_font_glyph_extents [Cairo 1.12.4]	cairo_scaled_font_reference [Cairo 1.12.4]
cairo_scaled_font_statuses [Cairo 1.12.4]	cairo_scaled_font_text_extents [Cairo 1.12.4]	cairo_scaled_font_text_to_glyphs [Cairo 1.12.4]
cairo_script_create [Cairo 1.12.4]	cairo_script_create_for_stream [Cairo 1.12.4]	cairo_script_from_recording_surface [Cairo 1.12.4]
cairo_script_get_mode [Cairo 1.12.4]	cairo_script_set_mode [Cairo 1.12.4]	cairo_script_surface_create [Cairo 1.12.4]
cairo_script_surface_create_for_target [Cairo 1.12.4]	cairo_script_write_comment [Cairo 1.12.4]	cairo_select_font_face [Cairo 1.12.4]
cairo_set_antialias [Cairo 1.12.4]	cairo_set_dash [Cairo 1.12.4]	cairo_set_fill_rule [Cairo 1.12.4]
cairo_set_font_face [Cairo 1.12.4]	cairo_set_font_matrix [Cairo 1.12.4]	cairo_set_font_options [Cairo 1.12.4]

cairo_set_font_size [Cairo 1.12.4]	cairo_set_line_cap [Cairo 1.12.4]	cairo_set_line_join [Cairo 1.12.4]
cairo_set_line_width [Cairo 1.12.4]	cairo_set_matrix [Cairo 1.12.4]	cairo_set_miter_limit [Cairo 1.12.4]
cairo_set_operator [Cairo 1.12.4]	cairo_set_scaled_font [Cairo 1.12.4]	cairo_set_source [Cairo 1.12.4]
cairo_set_source_rgb [Cairo 1.12.4]	cairo_set_source_rgba [Cairo 1.12.4]	cairo_set_source_surfac e [Cairo 1.12.4]
cairo_set_tolerance [Cairo 1.12.4]	cairo_show_glyphs [Cairo 1.12.4]	cairo_show_page [Cairo 1.12.4]
cairo_show_text [Cairo 1.12.4]	cairo_show_text_glyphs [Cairo 1.12.4]	cairo_status [Cairo 1.12.4]
cairo_status_to_string [Cairo 1.12.4]	cairo_stroke [Cairo 1.12.4]	cairo_stroke_extents [Cairo 1.12.4]
cairo_stroke_preserve [Cairo 1.12.4]	cairo_surface_copy_pag e [Cairo 1.12.4]	cairo_surface_create_fo r_rectangle [Cairo 1.12.4]
cairo_surface_create_ob server [LSB]	cairo_surface_create_si milar [Cairo 1.12.4]	cairo_surface_create_si milar_image [Cairo 1.12.4]
cairo_surface_destroy [Cairo 1.12.4]	cairo_surface_finish [Cairo 1.12.4]	cairo_surface_flush [Cairo 1.12.4]
cairo_surface_get_cont ent [Cairo 1.12.4]	cairo_surface_get_devic e [Cairo 1.12.4]	cairo_surface_get_devic e_offset [Cairo 1.12.4]
cairo_surface_get_fallb ack_resolution [Cairo 1.12.4]	cairo_surface_get_font_ options [Cairo 1.12.4]	cairo_surface_get_mim e_data [Cairo 1.12.4]
cairo_surface_get_type [Cairo 1.12.4]	cairo_surface_get_user_ data [Cairo 1.12.4]	cairo_surface_has_sho w_text_glyphs [Cairo 1.12.4]
cairo_surface_map_to_i mage [Cairo 1.12.4]	cairo_surface_mark_dir ty [Cairo 1.12.4]	cairo_surface_mark_dir ty_rectangle [Cairo 1.12.4]
cairo_surface_observer_ add_fill_callback [LSB]	cairo_surface_observer_ add_finish_callback [LSB]	cairo_surface_observer_ add_flush_callback [LSB]
cairo_surface_observer_ add_glyphs_callback [LSB]	cairo_surface_observer_ add_mask_callback [LSB]	cairo_surface_observer_ add_paint_callback [LSB]
cairo_surface_observer_ add_stroke_callback [LSB]	cairo_surface_observer_ elapsed [LSB]	cairo_surface_observer_ print [LSB]

cairo_surface_reference [Cairo 1.12.4]	cairo_surface_set_device_offset [Cairo 1.12.4]	cairo_surface_set_fallback_resolution [Cairo 1.12.4]
cairo_surface_set_mime_data [Cairo 1.12.4]	cairo_surface_set_user_data [Cairo 1.12.4]	cairo_surface_show_page [Cairo 1.12.4]
cairo_surface_status [Cairo 1.12.4]	cairo_surface_supports_mime_type [Cairo 1.12.4]	cairo_surface_unmap_image [Cairo 1.12.4]
cairo_surface_write_to_png [Cairo 1.12.4]	cairo_surface_write_to_png_stream [Cairo 1.12.4]	cairo_svg_get_versions [Cairo 1.12.4]
cairo_svg_surface_create [Cairo 1.12.4]	cairo_svg_surface_create_for_stream [Cairo 1.12.4]	cairo_svg_surface_restrict_to_version [Cairo 1.12.4]
cairo_svg_version_to_string [Cairo 1.12.4]	cairo_text_cluster_allocate [Cairo 1.12.4]	cairo_text_cluster_free [Cairo 1.12.4]
cairo_text_extents [Cairo 1.12.4]	cairo_text_path [Cairo 1.12.4]	cairo_toy_font_face_create [Cairo 1.12.4]
cairo_toy_font_face_get_family [Cairo 1.12.4]	cairo_toy_font_face_get_slant [Cairo 1.12.4]	cairo_toy_font_face_get_weight [Cairo 1.12.4]
cairo_transform [Cairo 1.12.4]	cairo_translate [Cairo 1.12.4]	cairo_user_font_face_create [Cairo 1.12.4]
cairo_user_font_face_get_init_func [Cairo 1.12.4]	cairo_user_font_face_get_render_glyph_func [Cairo 1.12.4]	cairo_user_font_face_get_text_to_glyphs_func [Cairo 1.12.4]
cairo_user_font_face_get_unicode_to_glyph_func [Cairo 1.12.4]	cairo_user_font_face_set_init_func [Cairo 1.12.4]	cairo_user_font_face_set_render_glyph_func [Cairo 1.12.4]
cairo_user_font_face_set_text_to_glyphs_func [Cairo 1.12.4]	cairo_user_font_face_set_unicode_to_glyph_func [Cairo 1.12.4]	cairo_user_to_device [Cairo 1.12.4]
cairo_user_to_device_distance [Cairo 1.12.4]	cairo_version [Cairo 1.12.4]	cairo_version_string [Cairo 1.12.4]
cairo_xcb_device_get_connection [Cairo 1.12.4]	cairo_xcb_surface_create [Cairo 1.12.4]	cairo_xcb_surface_create_for_bitmap [Cairo 1.12.4]
cairo_xcb_surface_set_drawable [Cairo 1.12.4]	cairo_xcb_surface_set_size [Cairo 1.12.4]	cairo_xlib_surface_create [Cairo 1.12.4]
cairo_xlib_surface_create_for_bitmap [Cairo 1.12.4]	cairo_xlib_surface_create_with_xrender_format [Cairo 1.12.4]	cairo_xlib_surface_get_depth [Cairo 1.12.4]
cairo_xlib_surface_get_display [Cairo 1.12.4]	cairo_xlib_surface_get_drawable [Cairo 1.12.4]	cairo_xlib_surface_get_height [Cairo 1.12.4]

cairo_xlib_surface_get_screen [Cairo 1.12.4]	cairo_xlib_surface_get_visual [Cairo 1.12.4]	cairo_xlib_surface_get_width [Cairo 1.12.4]
cairo_xlib_surface_get_xrender_format [Cairo 1.12.4]	cairo_xlib_surface_set_drawable [Cairo 1.12.4]	cairo_xlib_surface_set_size [Cairo 1.12.4]

15.2 Data Definitions for libcairo

This section defines global identifiers and their values that are associated with interfaces contained in libcairo. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

15.2.1 cairo/cairo-features.h

```
#define CAIRO_HAS_FT_FONT 1
#define CAIRO_HAS_PDF_SURFACE 1
#define CAIRO_HAS_PNG_FUNCTIONS 1
#define CAIRO_HAS_PS_SURFACE 1
#define CAIRO_HAS_SVG_SURFACE 1
#define CAIRO_HAS_XLIB_SURFACE 1
#define CAIRO_HAS_XLIB_XRENDER_SURFACE 1
#define CAIRO_VERSION_MAJOR 1
#define CAIRO_VERSION_MINOR 12
#define CAIRO_VERSION_MICRO 4
#define CAIRO_VERSION_STRING CAIRO_VERSION_STRINGIZE( \
    CAIRO_VERSION_MAJOR, \
    CAIRO_VERSION_MINOR, \
    CAIRO_VERSION_MICRO)
```

15.2.2 cairo/cairo-ft.h

```
typedef enum {
    CAIRO_FT_SYNTHESIZE_BOLD = 1 << 0,
    CAIRO_FT_SYNTHESIZE_OBLIQUE = 1 << 1
} cairo_ft_synthesize_t;

extern cairo_font_face_t
*cairo_ft_font_face_create_for_ft_face(FT_Face
    face,
    int
    load_flags);

extern cairo_font_face_t
*cairo_ft_font_face_create_for_pattern(FcPattern *
    pattern);

extern unsigned int
cairo_ft_font_face_get_synthesize(cairo_font_face_t *
    font_face);
```

```

extern void cairo_ft_font_face_set_synthesize(cairo_font_face_t *
                                              font_face,
                                              unsigned int synth_flags);
extern void cairo_ft_font_face_unset_synthesize(cairo_font_face_t
*
                                              font_face,
                                              unsigned int synth_flags);
extern void cairo_ft_font_options_substitute(const
cairo_font_options_t *
                                              options, FcPattern * pattern);
extern FT_Face cairo_ft_scaled_font_lock_face(cairo_scaled_font_t
*
                                              scaled_font);
extern void cairo_ft_scaled_font_unlock_face(cairo_scaled_font_t *
                                              scaled_font);

```

15.2.3 cairo/cairo-gobject.h

```

#define CAIRO_GOBJECT_TYPE_ANTIALIAS
cairo_gobject_antialias_get_type ()
#define CAIRO_GOBJECT_TYPE_CLUSTER_FLAGS
cairo_gobject_cluster_flags_get_type ()
#define CAIRO_GOBJECT_TYPE_CONTENT
cairo_gobject_content_get_type ()
#define CAIRO_GOBJECT_TYPE_CONTEXT
cairo_gobject_context_get_type ()
#define CAIRO_GOBJECT_TYPE_DEVICE
cairo_gobject_device_get_type ()
#define CAIRO_GOBJECT_TYPE_DEVICE_TYPE
cairo_gobject_device_type_get_type ()
#define CAIRO_GOBJECT_TYPE_EXTEND
cairo_gobject_extend_get_type ()
#define CAIRO_GOBJECT_TYPE_FILL_RULE
cairo_gobject_fill_rule_get_type ()
#define CAIRO_GOBJECT_TYPE_FILTER
cairo_gobject_filter_get_type ()
#define CAIRO_GOBJECT_TYPE_FONT_FACE
cairo_gobject_font_face_get_type ()
#define CAIRO_GOBJECT_TYPE_FONT_OPTIONS
cairo_gobject_font_options_get_type ()
#define CAIRO_GOBJECT_TYPE_FONT_SLANT
cairo_gobject_font_slant_get_type ()
#define CAIRO_GOBJECT_TYPE_FONT_TYPE
cairo_gobject_font_type_get_type ()
#define CAIRO_GOBJECT_TYPE_FONT_WEIGHT
cairo_gobject_font_weight_get_type ()
#define CAIRO_GOBJECT_TYPE_FORMAT
cairo_gobject_format_get_type ()
#define CAIRO_GOBJECT_TYPE_HNT_METRICS
cairo_gobject_hint_metrics_get_type ()
#define CAIRO_GOBJECT_TYPE_HINT_STYLE
cairo_gobject_hint_style_get_type ()
#define CAIRO_GOBJECT_TYPE_LINE_CAP
cairo_gobject_line_cap_get_type ()
#define CAIRO_GOBJECT_TYPE_LINE_JOIN
cairo_gobject_line_join_get_type ()
#define CAIRO_GOBJECT_TYPE_OPERATOR
cairo_gobject_operator_get_type ()
#define CAIRO_GOBJECT_TYPE_PATH_DATA_TYPE
cairo_gobject_path_data_type_get_type ()
#define CAIRO_GOBJECT_TYPE_PATTERN
cairo_gobject_pattern_get_type ()
#define CAIRO_GOBJECT_TYPE_PATTERN_TYPE
cairo_gobject_pattern_type_get_type ()

```

```

#define CAIRO_GOBJECT_TYPE_RECTANGLE
cairo_gobject_rectangle_get_type ()
#define CAIRO_GOBJECT_TYPE_RECTANGLE_INT
cairo_gobject_rectangle_int_get_type ()
#define CAIRO_GOBJECT_TYPE_REGION
cairo_gobject_region_get_type ()
#define CAIRO_GOBJECT_TYPE_REGION_OVERLAP
cairo_gobject_region_overlap_get_type ()
#define CAIRO_GOBJECT_TYPE_SCALED_FONT
cairo_gobject_scaled_font_get_type ()
#define CAIRO_GOBJECT_TYPE_STATUS
cairo_gobject_status_get_type ()
#define CAIRO_GOBJECT_TYPE_SUBPIXEL_ORDER
cairo_gobject_subpixel_order_get_type ()
#define CAIRO_GOBJECT_TYPE_SURFACE
cairo_gobject_surface_get_type ()
#define CAIRO_GOBJECT_TYPE_SURFACE_TYPE
cairo_gobject_surface_type_get_type ()

extern GType cairo_gobject_antialias_get_type(void);
extern GType cairo_gobject_content_get_type(void);
extern GType cairo_gobject_context_get_type(void);
extern GType cairo_gobject_device_get_type(void);
extern GType cairo_gobject_device_type_get_type(void);
extern GType cairo_gobject_extend_get_type(void);
extern GType cairo_gobject_fill_rule_get_type(void);
extern GType cairo_gobject_filter_get_type(void);
extern GType cairo_gobject_font_face_get_type(void);
extern GType cairo_gobject_font_options_get_type(void);
extern GType cairo_gobject_font_slant_get_type(void);
extern GType cairo_gobject_font_type_get_type(void);
extern GType cairo_gobject_font_weight_get_type(void);
extern GType cairo_gobject_format_get_type(void);
extern GType cairo_gobject_hint_metrics_get_type(void);
extern GType cairo_gobject_hint_style_get_type(void);
extern GType cairo_gobject_line_cap_get_type(void);
extern GType cairo_gobject_line_join_get_type(void);
extern GType cairo_gobject_operator_get_type(void);
extern GType cairo_gobject_path_data_type_get_type(void);
extern GType cairo_gobject_pattern_get_type(void);
extern GType cairo_gobject_pattern_type_get_type(void);
extern GType cairo_gobject_rectangle_get_type(void);
extern GType cairo_gobject_rectangle_int_get_type(void);
extern GType cairo_gobject_region_get_type(void);
extern GType cairo_gobject_region_overlap_get_type(void);
extern GType cairo_gobject_scaled_font_get_type(void);
extern GType cairo_gobject_status_get_type(void);
extern GType cairo_gobject_subpixel_order_get_type(void);
extern GType cairo_gobject_surface_get_type(void);
extern GType cairo_gobject_surface_type_get_type(void);
extern GType cairo_gobject_text_cluster_flags_get_type(void);

```

15.2.4 cairo/cairo-pdf.h

```

typedef enum _cairo_pdf_version {
    CAIRO_PDF_VERSION_1_4,
    CAIRO_PDF_VERSION_1_5
} cairo_pdf_version_t;
extern void cairo_pdf_get_versions(const cairo_pdf_version_t *
*versions,
                                int *num_versions);
extern cairo_surface_t *cairo_pdf_surface_create(const char
*filename,
                                                double width_in_points,
                                                double height_in_points);

```

```

extern cairo_surface_t
    *cairo_pdf_surface_create_for_stream(cairo_write_func_t
write_func,
                                void *closure,
                                double width_in_points,
                                double height_in_points);
extern void cairo_pdf_surface_restrict_to_version(cairo_surface_t
*
                                surface,
                                cairo_pdf_version_t
                                version);
extern void cairo_pdf_surface_set_size(cairo_surface_t * surface,
                                double width_in_points,
                                double height_in_points);
extern const char *cairo_pdf_version_to_string(cairo_pdf_version_t
version);

```

15.2.5 cairo/cairo-ps.h

```

typedef enum _cairo_ps_level {
    CAIRO_PS_LEVEL_2,
    CAIRO_PS_LEVEL_3
} cairo_ps_level_t;
extern void cairo_ps_get_levels(const cairo_ps_level_t * *levels,
                                int *num_levels);
extern const char *cairo_ps_level_to_string(cairo_ps_level_t
level);
extern cairo_surface_t *cairo_ps_surface_create(const char
*filename,
                                double width_in_points,
                                double height_in_points);
extern cairo_surface_t
    *cairo_ps_surface_create_for_stream(cairo_write_func_t
write_func,
                                void *closure,
                                double width_in_points,
                                double height_in_points);
extern void cairo_ps_surface_dsc_begin_page_setup(cairo_surface_t
*
                                surface);
extern void cairo_ps_surface_dsc_begin_setup(cairo_surface_t *
surface);
extern void cairo_ps_surface_dsc_comment(cairo_surface_t * surface,
                                const char *comment);
extern cairo_bool_t cairo_ps_surface_get_eps(cairo_surface_t *
surface);
extern void cairo_ps_surface_restrict_to_level(cairo_surface_t *
surface,
                                cairo_ps_level_t level);
extern void cairo_ps_surface_set_eps(cairo_surface_t * surface,
                                cairo_bool_t eps);
extern void cairo_ps_surface_set_size(cairo_surface_t * surface,
                                double width_in_points,
                                double height_in_points);

```

15.2.6 cairo/cairo-script-interpreter.h

```

typedef struct _cairo_script_interpreter
cairo_script_interpreter_t;
typedef void (*csi_destroy_func_t) (void *, void *);
typedef cairo_surface_t *(*csi_surface_create_func_t) (void *,
                                cairo_content_t,
                                double, double,

```

```

                                long int);
typedef      cairo_t      *(*csi_context_create_func_t)      (void      *,
cairo_surface_t *);
typedef void (*csi_show_page_func_t) (void *, cairo_t *);
typedef void (*csi_copy_page_func_t) (void *, cairo_t *);
typedef struct _cairo_script_interpreter_hooks {
    void *closure;
    csi_surface_create_func_t surface_create;
    csi_destroy_func_t surface_destroy;
    csi_context_create_func_t context_create;
    csi_destroy_func_t context_destroy;
    csi_show_page_func_t show_page;
    csi_copy_page_func_t copy_page;
} cairo_script_interpreter_hooks_t;
extern      cairo_script_interpreter_t      cairo_script_interpreter_t
*cairo_script_interpreter_create(void);
extern cairo_status_t
cairo_script_interpreter_destroy(cairo_script_interpreter_t * ctx);
extern cairo_status_t
cairo_script_interpreter_feed_stream(cairo_script_interpreter_t *
ctx,
                                FILE * stream);

extern cairo_status_t
cairo_script_interpreter_feed_string(cairo_script_interpreter_t *
ctx,
                                const char *line, int len);

extern cairo_status_t
cairo_script_interpreter_finish(cairo_script_interpreter_t * ctx);
extern unsigned int
cairo_script_interpreter_get_line_number(cairo_script_interpreter
_t * ctx);
extern void
cairo_script_interpreter_install_hooks(cairo_script_interpreter_t
* ctx,
                                const
                                cairo_script_interpreter_hooks_t
*
                                hooks);
extern cairo_script_interpreter_t
*cairo_script_interpreter_reference(cairo_script_interpreter_t
* ctx);
extern cairo_status_t
cairo_script_interpreter_run(cairo_script_interpreter_t * ctx,
                                const char *filename);

extern      cairo_status_t      cairo_script_interpreter_translate_stream(FILE *
                                stream,
                                cairo_write_func_t
                                write_func,
                                void
                                *closure);

```

15.2.7 cairo/cairo-script.h

```

typedef enum {
    CAIRO_SCRIPT_MODE_ASCII,
    CAIRO_SCRIPT_MODE_BINARY
} cairo_script_mode_t;
extern cairo_device_t *cairo_script_create(const char *filename);
extern      cairo_device_t      *cairo_script_create_for_stream(cairo_write_func_t
                                write_func,
                                void *closure);

```

```

extern                                cairo_status_t
cairo_script_from_recording_surface(cairo_device_t *
                                script,
                                cairo_surface_t *
                                recording_surface);
extern cairo_script_mode_t cairo_script_get_mode(cairo_device_t *
script);
extern void cairo_script_set_mode(cairo_device_t * script,
                                cairo_script_mode_t mode);
extern cairo_surface_t *cairo_script_surface_create(cairo_device_t
*
                                script,
                                cairo_content_t
                                content, double width,
                                double height);

extern cairo_surface_t
    *cairo_script_surface_create_for_target(cairo_device_t *
script,
                                cairo_surface_t * target);
extern void cairo_script_write_comment(cairo_device_t * script,
                                const char *comment, int len);

```

15.2.8 cairo/cairo-svg.h

```

typedef enum _cairo_svg_version {
    CAIRO_SVG_VERSION_1_1,
    CAIRO_SVG_VERSION_1_2 = 1
} cairo_svg_version_t;
extern void cairo_svg_get_versions(const cairo_svg_version_t *
*versions,
                                int *num_versions);
extern cairo_surface_t *cairo_svg_surface_create(const char
*filename,
                                double width_in_points,
                                double height_in_points);
extern cairo_surface_t
    *cairo_svg_surface_create_for_stream(cairo_write_func_t
write_func,
                                void *closure,
                                double width_in_points,
                                double height_in_points);
extern void cairo_svg_surface_restrict_to_version(cairo_surface_t
*
                                surface,
                                cairo_svg_version_t
                                version);
extern const char *cairo_svg_version_to_string(cairo_svg_version_t
version);

```

15.2.9 cairo/cairo-xcb.h

```

extern                                xcb_connection_t
*cairo_xcb_device_get_connection(cairo_device_t *
                                device);
extern cairo_surface_t *cairo_xcb_surface_create(xcb_connection_t
*
                                connection,
                                xcb_drawable_t drawable,
                                xcb_visualtype_t * visual,
                                int width, int height);
extern cairo_surface_t

```

```

    *cairo_xcb_surface_create_for_bitmap(xcb_connection_t
connection,
                                xcb_screen_t * screen,
                                xcb_pixmap_t bitmap, int width,
                                int height);
extern void cairo_xcb_surface_set_drawable(cairo_surface_t *
surface,
                                xcb_drawable_t drawable,
                                int width, int height);
extern void cairo_xcb_surface_set_size(cairo_surface_t * surface,
                                int width, int height);

```

15.2.10 cairo/cairo-xlib-xrender.h

```

extern cairo_surface_t
    *cairo_xlib_surface_create_with_xrender_format(Display * dpy,
                                Drawable drawable,
                                Screen * screen,
                                XRenderPictFormat *
                                format, int width,
                                int height);
extern XRenderPictFormat
    *cairo_xlib_surface_get_xrender_format(cairo_surface_t
surface);

```

15.2.11 cairo/cairo-xlib.h

```

extern cairo_surface_t *cairo_xlib_surface_create(Display * dpy,
                                Drawable drawable,
                                Visual * visual,
                                int width, int height);
extern
    cairo_surface_t
    *cairo_xlib_surface_create_for_bitmap(Display * dpy,
                                Pixmap bitmap,
                                Screen *
                                screen,
                                int width,
                                int height);
extern int cairo_xlib_surface_get_depth(cairo_surface_t * surface);
extern Display *cairo_xlib_surface_get_display(cairo_surface_t *
surface);
extern Drawable cairo_xlib_surface_get_drawable(cairo_surface_t *
surface);
extern int cairo_xlib_surface_get_height(cairo_surface_t *
surface);
extern Screen *cairo_xlib_surface_get_screen(cairo_surface_t *
surface);
extern Visual *cairo_xlib_surface_get_visual(cairo_surface_t *
surface);
extern int cairo_xlib_surface_get_width(cairo_surface_t * surface);
extern void cairo_xlib_surface_set_drawable(cairo_surface_t *
surface,
                                Drawable drawable, int width,
                                int height);
extern void cairo_xlib_surface_set_size(cairo_surface_t * surface,
                                int width, int height);

```

15.2.12 cairo/cairo.h

```

#define CAIRO_VERSION \
    CAIRO_VERSION_ENCODE( CAIRO_VERSION_MAJOR, CAIRO_VERSION_MINOR,
\

```



```

    CAIRO_VERSION_MICRO)
#define CAIRO_VERSION_ENCODE(major, minor, micro)      (\
    ((major) * 10000) \
+ ((minor) * 100) \
+ ((micro) * 1))
#define CAIRO_MIME_TYPE_UNIQUE_ID      "application/x-cairo.uuid"
#define CAIRO_MIME_TYPE_JP2            "image/jp2"
#define CAIRO_MIME_TYPE_JPEG           "image/jpeg"
#define CAIRO_MIME_TYPE_PNG            "image/png"
#define CAIRO_MIME_TYPE_URI            "text/x-uri"

typedef int cairo_bool_t;
typedef struct _cairo cairo_t;
typedef struct _cairo_surface cairo_surface_t;
typedef struct _cairo_matrix {
    double xx;
    double yx;
    double xy;
    double yy;
    double x0;
    double y0;
} cairo_matrix_t;
typedef struct _cairo_pattern cairo_pattern_t;
typedef void (*cairo_destroy_func_t) (void *);
typedef struct _cairo_user_data_key {
    int unused;
} cairo_user_data_key_t;
typedef enum _cairo_status {
    CAIRO_STATUS_SUCCESS,
    CAIRO_STATUS_NO_MEMORY = 1,
    CAIRO_STATUS_INVALID_RESTORE = 2,
    CAIRO_STATUS_INVALID_POP_GROUP = 3,
    CAIRO_STATUS_NO_CURRENT_POINT = 4,
    CAIRO_STATUS_INVALID_MATRIX = 5,
    CAIRO_STATUS_INVALID_STATUS = 6,
    CAIRO_STATUS_NULL_POINTER = 7,
    CAIRO_STATUS_INVALID_STRING = 8,
    CAIRO_STATUS_INVALID_PATH_DATA = 9,
    CAIRO_STATUS_READ_ERROR = 10,
    CAIRO_STATUS_WRITE_ERROR = 11,
    CAIRO_STATUS_SURFACE_FINISHED = 12,
    CAIRO_STATUS_SURFACE_TYPE_MISMATCH = 13,
    CAIRO_STATUS_PATTERN_TYPE_MISMATCH = 14,
    CAIRO_STATUS_INVALID_CONTENT = 15,
    CAIRO_STATUS_INVALID_FORMAT = 16,
    CAIRO_STATUS_INVALID_VISUAL = 17,
    CAIRO_STATUS_FILE_NOT_FOUND = 18,
    CAIRO_STATUS_INVALID_DASH = 19,
    CAIRO_STATUS_INVALID_DSC_COMMENT = 20,
    CAIRO_STATUS_INVALID_INDEX = 21,
    CAIRO_STATUS_CLIP_NOT_REPRESENTABLE = 22,
    CAIRO_STATUS_TEMP_FILE_ERROR = 23,
    CAIRO_STATUS_INVALID_STRIDE = 24,
    CAIRO_STATUS_FONT_TYPE_MISMATCH = 25,
    CAIRO_STATUS_USER_FONT_IMMUTABLE = 26,
    CAIRO_STATUS_USER_FONT_ERROR = 27,
    CAIRO_STATUS_NEGATIVE_COUNT = 28,
    CAIRO_STATUS_INVALID_CLUSTERS = 29,
    CAIRO_STATUS_INVALID_SLANT = 30,
    CAIRO_STATUS_INVALID_WEIGHT = 31,
    CAIRO_STATUS_INVALID_SIZE = 32,
    CAIRO_STATUS_USER_FONT_NOT_IMPLEMENTED = 33,
    CAIRO_STATUS_DEVICE_TYPE_MISMATCH = 34,
    CAIRO_STATUS_DEVICE_ERROR = 35,
    CAIRO_STATUS_INVALID_MESH_CONSTRUCTION = 36,
    CAIRO_STATUS_DEVICE_FINISHED = 37,

```

```

        CAIRO_STATUS_LAST_STATUS = 38
    } cairo_status_t;
typedef enum _cairo_content {
    CAIRO_CONTENT_COLOR = 4096,
    CAIRO_CONTENT_ALPHA = 8192,
    CAIRO_CONTENT_COLOR_ALPHA = 12288
} cairo_content_t;
typedef cairo_status_t(*cairo_write_func_t) (void *, const unsigned
char *,
                                         unsigned int);
typedef cairo_status_t(*cairo_read_func_t) (void *, unsigned char
*,
                                         unsigned int);

typedef enum _cairo_operator {
    CAIRO_OPERATOR_CLEAR,
    CAIRO_OPERATOR_SOURCE = 1,
    CAIRO_OPERATOR_OVER = 2,
    CAIRO_OPERATOR_IN = 3,
    CAIRO_OPERATOR_OUT = 4,
    CAIRO_OPERATOR_ATOP = 5,
    CAIRO_OPERATOR_DEST = 6,
    CAIRO_OPERATOR_DEST_OVER = 7,
    CAIRO_OPERATOR_DEST_IN = 8,
    CAIRO_OPERATOR_DEST_OUT = 9,
    CAIRO_OPERATOR_DEST_ATOP = 10,
    CAIRO_OPERATOR_XOR = 11,
    CAIRO_OPERATOR_ADD = 12,
    CAIRO_OPERATOR_SATURATE = 13,
    CAIRO_OPERATOR_MULTIPLY = 14,
    CAIRO_OPERATOR_SCREEN = 15,
    CAIRO_OPERATOR_OVERLAY = 16,
    CAIRO_OPERATOR_DARKEN = 17,
    CAIRO_OPERATOR_LIGHTEN = 18,
    CAIRO_OPERATOR_COLOR_DODGE = 19,
    CAIRO_OPERATOR_COLOR_BURN = 20,
    CAIRO_OPERATOR_HARD_LIGHT = 21,
    CAIRO_OPERATOR_SOFT_LIGHT = 22,
    CAIRO_OPERATOR_DIFFERENCE = 23,
    CAIRO_OPERATOR_EXCLUSION = 24,
    CAIRO_OPERATOR_HSL_HUE = 25,
    CAIRO_OPERATOR_HSL_SATURATION = 26,
    CAIRO_OPERATOR_HSL_COLOR = 27,
    CAIRO_OPERATOR_HSL_LUMINOSITY = 28
} cairo_operator_t;
typedef enum _cairo_antialias {
    CAIRO_ANTIALIAS_DEFAULT,
    CAIRO_ANTIALIAS_NONE = 1,
    CAIRO_ANTIALIAS_GRAY = 2,
    CAIRO_ANTIALIAS_SUBPIXEL = 3,
    CAIRO_ANTIALIAS_FAST = 4,
    CAIRO_ANTIALIAS_GOOD = 5,
    CAIRO_ANTIALIAS_BEST = 6
} cairo_antialias_t;
typedef enum _cairo_fill_rule {
    CAIRO_FILL_RULE_WINDING,
    CAIRO_FILL_RULE_EVEN_ODD = 1
} cairo_fill_rule_t;
typedef enum _cairo_line_cap {
    CAIRO_LINE_CAP_BUTT,
    CAIRO_LINE_CAP_ROUND = 1,
    CAIRO_LINE_CAP_SQUARE = 2
} cairo_line_cap_t;
typedef enum _cairo_line_join {
    CAIRO_LINE_JOIN_MITER,
    CAIRO_LINE_JOIN_ROUND = 1,
    CAIRO_LINE_JOIN_BEVEL = 2
}

```

```

} cairo_line_join_t;
typedef struct _cairo_rectangle {
    double x;
    double y;
    double width;
    double height;
} cairo_rectangle_t;
typedef struct _cairo_rectangle_list {
    cairo_status_t status;
    cairo_rectangle_t *rectangles;
    int num_rectangles;
} cairo_rectangle_list_t;
typedef struct _cairo_scaled_font cairo_scaled_font_t;
typedef struct _cairo_font_face cairo_font_face_t;
typedef struct {
    long unsigned int index;
    double x;
    double y;
} cairo_glyph_t;
typedef struct {
    double x_bearing;
    double y_bearing;
    double width;
    double height;
    double x_advance;
    double y_advance;
} cairo_text_extents_t;
typedef struct {
    double ascent;
    double descent;
    double height;
    double max_x_advance;
    double max_y_advance;
} cairo_font_extents_t;
typedef enum _cairo_font_slant {
    CAIRO_FONT_SLANT_NORMAL,
    CAIRO_FONT_SLANT_ITALIC = 1,
    CAIRO_FONT_SLANT_OBLIQUE = 2
} cairo_font_slant_t;
typedef enum _cairo_font_weight {
    CAIRO_FONT_WEIGHT_NORMAL,
    CAIRO_FONT_WEIGHT_BOLD = 1
} cairo_font_weight_t;
typedef enum _cairo_subpixel_order {
    CAIRO_SUBPIXEL_ORDER_DEFAULT,
    CAIRO_SUBPIXEL_ORDER_RGB = 1,
    CAIRO_SUBPIXEL_ORDER_BGR = 2,
    CAIRO_SUBPIXEL_ORDER_VRGB = 3,
    CAIRO_SUBPIXEL_ORDER_VBGR = 4
} cairo_subpixel_order_t;
typedef enum _cairo_hint_style {
    CAIRO_HINT_STYLE_DEFAULT,
    CAIRO_HINT_STYLE_NONE = 1,
    CAIRO_HINT_STYLE_SLIGHT = 2,
    CAIRO_HINT_STYLE_MEDIUM = 3,
    CAIRO_HINT_STYLE_FULL = 4
} cairo_hint_style_t;
typedef enum _cairo_hint_metrics {
    CAIRO_HINT_METRICS_DEFAULT,
    CAIRO_HINT_METRICS_OFF = 1,
    CAIRO_HINT_METRICS_ON = 2
} cairo_hint_metrics_t;
typedef struct _cairo_font_options cairo_font_options_t;
typedef enum _cairo_font_type {
    CAIRO_FONT_TYPE_TOY,
    CAIRO_FONT_TYPE_FT = 1,

```

```

        CAIRO_FONT_TYPE_WIN32 = 2,
        CAIRO_FONT_TYPE_QUARTZ = 3,
        CAIRO_FONT_TYPE_USER = 4
    } cairo_font_type_t;
typedef enum _cairo_path_data_type {
    CAIRO_PATH_MOVE_TO,
    CAIRO_PATH_LINE_TO = 1,
    CAIRO_PATH_CURVE_TO = 2,
    CAIRO_PATH_CLOSE_PATH = 3
} cairo_path_data_type_t;
typedef union _cairo_path_data_t {
    struct {
        cairo_path_data_type_t type;
        int length;
    } header;
    struct {
        double x;
        double y;
    } point;
} cairo_path_data_t;
typedef struct cairo_path {
    cairo_status_t status;
    cairo_path_data_t *data;
    int num_data;
} cairo_path_t;
typedef enum _cairo_surface_type {
    CAIRO_SURFACE_TYPE_IMAGE,
    CAIRO_SURFACE_TYPE_PDF = 1,
    CAIRO_SURFACE_TYPE_PS = 2,
    CAIRO_SURFACE_TYPE_XLIB = 3,
    CAIRO_SURFACE_TYPE_XCB = 4,
    CAIRO_SURFACE_TYPE_GLITZ = 5,
    CAIRO_SURFACE_TYPE_QUARTZ = 6,
    CAIRO_SURFACE_TYPE_WIN32 = 7,
    CAIRO_SURFACE_TYPE_BEOS = 8,
    CAIRO_SURFACE_TYPE_DIRECTFB = 9,
    CAIRO_SURFACE_TYPE_SVG = 10,
    CAIRO_SURFACE_TYPE_QS2 = 11,
    CAIRO_SURFACE_TYPE_WIN32_PRINTING = 12,
    CAIRO_SURFACE_TYPE_QUARTZ_IMAGE = 13,
    CAIRO_SURFACE_TYPE_SCRIPT = 14,
    CAIRO_SURFACE_TYPE_QT = 15,
    CAIRO_SURFACE_TYPE_RECORDING = 16,
    CAIRO_SURFACE_TYPE_VG = 17,
    CAIRO_SURFACE_TYPE_GL = 18,
    CAIRO_SURFACE_TYPE_DRM = 19,
    CAIRO_SURFACE_TYPE_TEE = 20,
    CAIRO_SURFACE_TYPE_XML = 21,
    CAIRO_SURFACE_TYPE_SKIA = 22,
    CAIRO_SURFACE_TYPE_SUBSURFACE = 23,
    CAIRO_SURFACE_TYPE_COGL = 24
} cairo_surface_type_t;
typedef enum _cairo_format {
    CAIRO_FORMAT_ARGB32,
    CAIRO_FORMAT_RGB24 = 1,
    CAIRO_FORMAT_A8 = 2,
    CAIRO_FORMAT_A1 = 3,
    CAIRO_FORMAT_RGB16_565 = 4,
    CAIRO_FORMAT_RGB30 = 5
} cairo_format_t;
typedef enum _cairo_pattern_type {
    CAIRO_PATTERN_TYPE_SOLID,
    CAIRO_PATTERN_TYPE_SURFACE = 1,
    CAIRO_PATTERN_TYPE_LINEAR = 2,
    CAIRO_PATTERN_TYPE_RADIAL = 3,
    CAIRO_PATTERN_TYPE_MESH = 4,

```

```

        CAIRO_PATTERN_TYPE_RASTER_SOURCE = 5
    } cairo_pattern_type_t;
typedef enum _cairo_extend {
    CAIRO_EXTEND_NONE,
    CAIRO_EXTEND_REPEAT = 1,
    CAIRO_EXTEND_REFLECT = 2,
    CAIRO_EXTEND_PAD = 3
} cairo_extend_t;
typedef enum _cairo_filter {
    CAIRO_FILTER_FAST,
    CAIRO_FILTER_GOOD = 1,
    CAIRO_FILTER_BEST = 2,
    CAIRO_FILTER_NEAREST = 3,
    CAIRO_FILTER_BILINEAR = 4,
    CAIRO_FILTER_GAUSSIAN = 5
} cairo_filter_t;
typedef struct _cairo_device cairo_device_t;
typedef struct _cairo_rectangle_int {
    int x;
    int y;
    int width;
    int height;
} cairo_rectangle_int_t;
typedef enum _cairo_text_cluster_flags {
    CAIRO_TEXT_CLUSTER_FLAG_BACKWARD = 0x00000001
} cairo_text_cluster_flags_t;
typedef
cairo_status_t(*cairo_user_scaled_font_init_func_t)
(cairo_scaled_font_t *,
                                     cairo_t *,
                                     cairo_font_extents_t
                                     *);

typedef
cairo_status_t(*cairo_user_scaled_font_render_glyph_func_t)
(cairo_scaled_font_t *, unsigned long int, cairo_t *,
cairo_text_extents_t *);
typedef
cairo_status_t(*cairo_user_scaled_font_text_to_glyphs_func_t)
(cairo_scaled_font_t *, const char *, int, cairo_glyph_t ** , int
*,
cairo_text_cluster_t ** , int *, cairo_text_cluster_flags_t *);
typedef
cairo_status_t(*cairo_user_scaled_font_unicode_to_glyph_func_t)
(cairo_scaled_font_t *, unsigned long int, unsigned long int *);
typedef enum _cairo_device_type {
    CAIRO_DEVICE_TYPE_DRM,
    CAIRO_DEVICE_TYPE_GL,
    CAIRO_DEVICE_TYPE_SCRIPT,
    CAIRO_DEVICE_TYPE_XCB,
    CAIRO_DEVICE_TYPE_XLIB,
    CAIRO_DEVICE_TYPE_XML,
    CAIRO_DEVICE_TYPE_COGL,
    CAIRO_DEVICE_TYPE_WIN32,
    CAIRO_DEVICE_TYPE_INVALID = -1
} cairo_device_type_t;
typedef enum {
    CAIRO_SURFACE_OBSERVER_NORMAL = 0,
    CAIRO_SURFACE_OBSERVER_RECORD_OPERATIONS = 0x01
} cairo_surface_observer_mode_t;
typedef void (*cairo_surface_observer_callback_t) (cairo_surface_t
*,
                                     cairo_surface_t *,
                                     void *);

typedef cairo_surface_t
*(*cairo_raster_source_acquire_func_t) (cairo_pattern_t *, void
*,

```

```

cairo_surface_t *,
const cairo_rectangle_int_t
*);
typedef void (*cairo_raster_source_release_func_t)
(cairo_pattern_t *,
void *,
cairo_surface_t *);
typedef
cairo_status_t(*cairo_raster_source_snapshot_func_t)
(cairo_pattern_t *,
void *);
typedef cairo_status_t(*cairo_raster_source_copy_func_t)
(cairo_pattern_t
*, void *,
const
cairo_pattern_t
*);
typedef void (*cairo_raster_source_finish_func_t) (cairo_pattern_t
*,
void *);
typedef struct _cairo_region cairo_region_t;
typedef enum _cairo_region_overlap {
CAIRO_REGION_OVERLAP_IN,
CAIRO_REGION_OVERLAP_OUT,
CAIRO_REGION_OVERLAP_PART
} cairo_region_overlap_t;
extern void cairo_append_path(cairo_t * cr, const cairo_path_t *
path);
extern void cairo_arc(cairo_t * cr, double xc, double yc, double
radius,
double angle1, double angle2);
extern void cairo_arc_negative(cairo_t * cr, double xc, double yc,
double radius, double angle1,
double angle2);
extern void cairo_clip(cairo_t * cr);
extern void cairo_clip_extents(cairo_t * cr, double *x1, double *y1,
double *x2, double *y2);
extern void cairo_clip_preserve(cairo_t * cr);
extern void cairo_close_path(cairo_t * cr);
extern
cairo_rectangle_list_t
*cairo_copy_clip_rectangle_list(cairo_t *
cr);
extern void cairo_copy_page(cairo_t * cr);
extern cairo_path_t *cairo_copy_path(cairo_t * cr);
extern cairo_path_t *cairo_copy_path_flat(cairo_t * cr);
extern cairo_t *cairo_create(cairo_surface_t * target);
extern void cairo_curve_to(cairo_t * cr, double x1, double y1,
double x2,
double y2, double x3, double y3);
extern void cairo_destroy(cairo_t * cr);
extern cairo_status_t cairo_device_acquire(cairo_device_t *
device);
extern void cairo_device_destroy(cairo_device_t * device);
extern void cairo_device_finish(cairo_device_t * device);
extern void cairo_device_flush(cairo_device_t * device);
extern
unsigned
int
cairo_device_get_reference_count(cairo_device_t *
device);
extern cairo_device_type_t cairo_device_get_type(cairo_device_t *
device);
extern void *cairo_device_get_user_data(cairo_device_t * device,
const cairo_user_data_key_t *
key);
extern double cairo_device_observer_elapsed(cairo_device_t *
device);

```

```

extern double cairo_device_observer_fill_elapsed(cairo_device_t *
device);
extern double cairo_device_observer_glyphs_elapsed(cairo_device_t
*
device);
extern double cairo_device_observer_mask_elapsed(cairo_device_t *
device);
extern double cairo_device_observer_paint_elapsed(cairo_device_t *
device);
extern cairo_status_t cairo_device_observer_print(cairo_device_t *
device,
cairo_write_func_t
write_func,
void *closure);
extern double cairo_device_observer_stroke_elapsed(cairo_device_t
*
device);
extern cairo_device_t *cairo_device_reference(cairo_device_t *
device);
extern void cairo_device_release(cairo_device_t * device);
extern cairo_status_t cairo_device_set_user_data(cairo_device_t *
device,
const
cairo_user_data_key_t *
key, void *user_data,
cairo_destroy_func_t
destroy);
extern cairo_status_t cairo_device_status(cairo_device_t * device);
extern void cairo_device_to_user(cairo_t * cr, double *x, double
*y);
extern void cairo_device_to_user_distance(cairo_t * cr, double *dx,
double *dy);
extern void cairo_fill(cairo_t * cr);
extern void cairo_fill_extents(cairo_t * cr, double *x1, double *y1,
double *x2, double *y2);
extern void cairo_fill_preserve(cairo_t * cr);
extern void cairo_font_extents(cairo_t * cr,
cairo_font_extents_t * extents);
extern void cairo_font_face_destroy(cairo_font_face_t * font_face);
extern unsigned int
cairo_font_face_get_reference_count(cairo_font_face_t *
font_face);
extern cairo_font_type_t
cairo_font_face_get_type(cairo_font_face_t *
font_face);
extern void *cairo_font_face_get_user_data(cairo_font_face_t *
font_face,
const cairo_user_data_key_t *
key);
extern cairo_font_face_t
*cairo_font_face_reference(cairo_font_face_t *
font_face);
extern cairo_status_t
cairo_font_face_set_user_data(cairo_font_face_t *
font_face,
const
cairo_user_data_key_t
*
key, void *user_data,
cairo_destroy_func_t
destroy);
extern cairo_status_t cairo_font_face_status(cairo_font_face_t *
font_face);
extern cairo_font_options_t *cairo_font_options_copy(const
cairo_font_options_t
*

```

```

original);
extern cairo_font_options_t *cairo_font_options_create(void);
extern void cairo_font_options_destroy(cairo_font_options_t *
options);
extern cairo_bool_t cairo_font_options_equal(const
cairo_font_options_t *
options,
const cairo_font_options_t *
other);
extern cairo_antialias_t cairo_font_options_get_antialias(const
cairo_font_options_t
* options);
extern cairo_hint_metrics_t
cairo_font_options_get_hint_metrics(const
cairo_font_options_t
* options);
extern cairo_hint_style_t cairo_font_options_get_hint_style(const
cairo_font_options_t
* options);
extern cairo_subpixel_order_t
cairo_font_options_get_subpixel_order(const
cairo_font_options_t
*
options);
extern long unsigned int cairo_font_options_hash(const
cairo_font_options_t
* options);
extern void cairo_font_options_merge(cairo_font_options_t *
options,
const cairo_font_options_t * other);
extern void cairo_font_options_set_antialias(cairo_font_options_t
*
options,
cairo_antialias_t antialias);
extern void
cairo_font_options_set_hint_metrics(cairo_font_options_t *
options,
cairo_hint_metrics_t
hint_metrics);
extern void cairo_font_options_set_hint_style(cairo_font_options_t
*
options,
cairo_hint_style_t
hint_style);
extern void
cairo_font_options_set_subpixel_order(cairo_font_options_t *
options,
cairo_subpixel_order_t
subpixel_order);
extern cairo_status_t
cairo_font_options_status(cairo_font_options_t *
options);
extern int cairo_format_stride_for_width(cairo_format_t format,
int width);
extern cairo_antialias_t cairo_get_antialias(cairo_t * cr);
extern void cairo_get_current_point(cairo_t * cr, double *x, double
*y);
extern void cairo_get_dash(cairo_t * cr, double *dashes, double
*offset);
extern int cairo_get_dash_count(cairo_t * cr);
extern cairo_fill_rule_t cairo_get_fill_rule(cairo_t * cr);
extern cairo_font_face_t *cairo_get_font_face(cairo_t * cr);

```



```

extern void cairo_get_font_matrix(cairo_t * cr, cairo_matrix_t *
matrix);
extern void cairo_get_font_options(cairo_t * cr,
    cairo_font_options_t * options);
extern cairo_surface_t *cairo_get_group_target(cairo_t * cr);
extern cairo_line_cap_t cairo_get_line_cap(cairo_t * cr);
extern cairo_line_join_t cairo_get_line_join(cairo_t * cr);
extern double cairo_get_line_width(cairo_t * cr);
extern void cairo_get_matrix(cairo_t * cr, cairo_matrix_t * matrix);
extern double cairo_get_miter_limit(cairo_t * cr);
extern cairo_operator_t cairo_get_operator(cairo_t * cr);
extern unsigned int cairo_get_reference_count(cairo_t * cr);
extern cairo_scaled_font_t *cairo_get_scaled_font(cairo_t * cr);
extern cairo_pattern_t *cairo_get_source(cairo_t * cr);
extern cairo_surface_t *cairo_get_target(cairo_t * cr);
extern double cairo_get_tolerance(cairo_t * cr);
extern void *cairo_get_user_data(cairo_t * cr,
    const cairo_user_data_key_t * key);
extern cairo_glyph_t *cairo_glyph_allocate(int num_glyphs);
extern void cairo_glyph_extents(cairo_t * cr, const cairo_glyph_t
* glyphs,
    int num_glyphs,
    cairo_text_extents_t * extents);
extern void cairo_glyph_free(cairo_glyph_t * glyphs);
extern void cairo_glyph_path(cairo_t * cr, const cairo_glyph_t *
glyphs,
    int num_glyphs);
extern cairo_bool_t cairo_has_current_point(cairo_t * cr);
extern void cairo_identity_matrix(cairo_t * cr);
extern cairo_surface_t *cairo_image_surface_create(cairo_format_t
format,
    int width, int height);
extern
    cairo_surface_t
*cairo_image_surface_create_for_data(unsigned char
    *data,
    cairo_format_t
    format,
    int width,
    int height,
    int stride);
extern cairo_surface_t *cairo_image_surface_create_from_png(const
char
    *filename);
extern cairo_surface_t
    *cairo_image_surface_create_from_png_stream(cairo_read_func_t
    read_func, void *closure);
extern unsigned char *cairo_image_surface_get_data(cairo_surface_t
*
    surface);
extern
    cairo_format_t
cairo_image_surface_get_format(cairo_surface_t *
    surface);
extern int
    cairo_image_surface_get_height(cairo_surface_t *
    surface);
extern int
    cairo_image_surface_get_stride(cairo_surface_t *
    surface);
extern int
    cairo_image_surface_get_width(cairo_surface_t *
    surface);
extern cairo_bool_t cairo_in_clip(cairo_t * cr, double x, double
y);
extern cairo_bool_t cairo_in_fill(cairo_t * cr, double x, double
y);
extern cairo_bool_t cairo_in_stroke(cairo_t * cr, double x, double
y);
extern void cairo_line_to(cairo_t * cr, double x, double y);
extern void cairo_mask(cairo_t * cr, cairo_pattern_t * pattern);

```

```

extern void cairo_mask_surface(cairo_t * cr, cairo_surface_t *
surface,
                                double surface_x, double surface_y);
extern void cairo_matrix_init(cairo_matrix_t * matrix, double xx,
                                double yx, double xy, double yy, double
x0,
                                double y0);
extern void cairo_matrix_init_identity(cairo_matrix_t * matrix);
extern void cairo_matrix_init_rotate(cairo_matrix_t * matrix,
                                double radians);
extern void cairo_matrix_init_scale(cairo_matrix_t * matrix, double
sx,
                                double sy);
extern void cairo_matrix_init_translate(cairo_matrix_t * matrix,
double tx,
                                double ty);
extern cairo_status_t cairo_matrix_invert(cairo_matrix_t * matrix);
extern void cairo_matrix_multiply(cairo_matrix_t * result,
                                const cairo_matrix_t * a,
                                const cairo_matrix_t * b);
extern void cairo_matrix_rotate(cairo_matrix_t * matrix, double
radians);
extern void cairo_matrix_scale(cairo_matrix_t * matrix, double sx,
double sy);
extern void cairo_matrix_transform_distance(const cairo_matrix_t *
matrix,
                                double *dx, double *dy);
extern void cairo_matrix_transform_point(const cairo_matrix_t *
matrix,
                                double *x, double *y);
extern void cairo_matrix_translate(cairo_matrix_t * matrix, double
tx,
                                double ty);
extern void cairo_mesh_pattern_begin_patch(cairo_pattern_t *
pattern);
extern void cairo_mesh_pattern_curve_to(cairo_pattern_t * pattern,
double x1, double y1, double x2,
double y2, double x3, double y3);
extern void cairo_mesh_pattern_end_patch(cairo_pattern_t *
pattern);
extern
                                cairo_status_t
cairo_mesh_pattern_get_control_point(cairo_pattern_t
                                * pattern,
                                unsigned int
                                patch_num,
                                unsigned int
                                point_num,
                                double *x,
                                double *y);
extern cairo_status_t
cairo_mesh_pattern_get_corner_color_rgba(cairo_pattern_t * pattern,
                                unsigned int patch_num,
                                unsigned int corner_num,
                                double *red, double *green,
                                double *blue, double *alpha);
extern
                                cairo_status_t
cairo_mesh_pattern_get_patch_count(cairo_pattern_t *
                                pattern,
                                unsigned int
                                *count);
extern cairo_path_t *cairo_mesh_pattern_get_path(cairo_pattern_t *
pattern,
                                unsigned int patch_num);
extern void cairo_mesh_pattern_line_to(cairo_pattern_t * pattern,
double x,
                                double y);

```

```

extern void cairo_mesh_pattern_move_to(cairo_pattern_t * pattern,
double x,
double y);
extern void cairo_mesh_pattern_set_control_point(cairo_pattern_t *
pattern,
unsigned int point_num,
double x, double y);
extern
cairo_mesh_pattern_set_corner_color_rgb(cairo_pattern_t *
pattern,
unsigned int
corner_num, double red,
double green,
double blue);
extern
cairo_mesh_pattern_set_corner_color_rgba(cairo_pattern_t *
pattern,
unsigned int
corner_num,
double red,
double green,
double blue,
double alpha);
extern void cairo_move_to(cairo_t * cr, double x, double y);
extern void cairo_new_path(cairo_t * cr);
extern void cairo_new_sub_path(cairo_t * cr);
extern void cairo_paint(cairo_t * cr);
extern void cairo_paint_with_alpha(cairo_t * cr, double alpha);
extern void cairo_path_destroy(cairo_path_t * path);
extern void cairo_path_extents(cairo_t * cr, double *x1, double *y1,
double *x2, double *y2);
extern void cairo_pattern_add_color_stop_rgb(cairo_pattern_t *
pattern,
double offset, double red,
double green, double blue);
extern void cairo_pattern_add_color_stop_rgba(cairo_pattern_t *
pattern,
double offset, double red,
double green, double blue,
double alpha);
extern
cairo_pattern_t
*cairo_pattern_create_for_surface(cairo_surface_t *
surface);
extern cairo_pattern_t *cairo_pattern_create_linear(double x0,
double y0,
double x1, double y1);
extern cairo_pattern_t *cairo_pattern_create_mesh(void);
extern cairo_pattern_t *cairo_pattern_create_radial(double cx0,
double cy0,
double radius0,
double cx1, double cy1,
double radius1);
extern cairo_pattern_t *cairo_pattern_create_raster_source(void
*user_data,
cairo_content_t
content,
int width,
int height);
extern cairo_pattern_t *cairo_pattern_create_rgb(double red,
double green,
double blue);
extern cairo_pattern_t *cairo_pattern_create_rgba(double red,
double green,
double blue,
double alpha);
extern void cairo_pattern_destroy(cairo_pattern_t * pattern);

```

```

extern                                cairo_status_t
cairo_pattern_get_color_stop_count(cairo_pattern_t *
                                pattern,
                                int *count);

extern                                cairo_status_t
cairo_pattern_get_color_stop_rgba(cairo_pattern_t *
                                pattern, int index,
                                double *offset,
                                double *red,
                                double *green,
                                double *blue,
                                double *alpha);

extern cairo_extend_t  cairo_pattern_get_extend(cairo_pattern_t *
pattern);
extern cairo_filter_t  cairo_pattern_get_filter(cairo_pattern_t *
pattern);
extern                                cairo_status_t
cairo_pattern_get_linear_points(cairo_pattern_t *
                                pattern, double *x0,
                                double *y0,
                                double *x1,
                                double *y1);

extern void cairo_pattern_get_matrix(cairo_pattern_t * pattern,
                                cairo_matrix_t * matrix);

extern                                cairo_status_t
cairo_pattern_get_radial_circles(cairo_pattern_t *
                                pattern, double *x0,
                                double *y0,
                                double *r0,
                                double *x1,
                                double *y1,
                                double *r1);

extern                                unsigned int
cairo_pattern_get_reference_count(cairo_pattern_t *
                                pattern);

extern cairo_status_t  cairo_pattern_get_rgba(cairo_pattern_t *
pattern,
                                double *red, double *green,
                                double *blue, double *alpha);

extern cairo_status_t  cairo_pattern_get_surface(cairo_pattern_t *
pattern,
                                cairo_surface_t *
                                *surface);

extern cairo_pattern_type_t  cairo_pattern_get_type(cairo_pattern_t
*
                                pattern);

extern void *cairo_pattern_get_user_data(cairo_pattern_t * pattern,
                                const cairo_user_data_key_t *
                                key);

extern cairo_pattern_t *cairo_pattern_reference(cairo_pattern_t *
pattern);

extern void cairo_pattern_set_extend(cairo_pattern_t * pattern,
                                cairo_extend_t extend);

extern void cairo_pattern_set_filter(cairo_pattern_t * pattern,
                                cairo_filter_t filter);

extern void cairo_pattern_set_matrix(cairo_pattern_t * pattern,
                                const cairo_matrix_t * matrix);

extern cairo_status_t  cairo_pattern_set_user_data(cairo_pattern_t
*
                                pattern,
                                const
                                cairo_user_data_key_t *
                                key, void *user_data,
                                cairo_destroy_func_t
                                destroy);

```

```

extern cairo_status_t cairo_pattern_status(cairo_pattern_t *
pattern);
extern cairo_pattern_t *cairo_pop_group(cairo_t * cr);
extern void cairo_pop_group_to_source(cairo_t * cr);
extern void cairo_push_group(cairo_t * cr);
extern void cairo_push_group_with_content(cairo_t * cr,
                                         cairo_content_t content);
extern void
cairo_raster_source_pattern_get_acquire(cairo_pattern_t *
                                         pattern,

cairo_raster_source_acquire_func_t

                                         * acquire,

cairo_raster_source_release_func_t

                                         * release);
extern void
*cairo_raster_source_pattern_get_callback_data(cairo_pattern_t
* pattern);
extern cairo_raster_source_copy_func_t
cairo_raster_source_pattern_get_copy(cairo_pattern_t * pattern);
extern cairo_raster_source_finish_func_t
cairo_raster_source_pattern_get_finish(cairo_pattern_t * pattern);
extern cairo_raster_source_snapshot_func_t
cairo_raster_source_pattern_get_snapshot(cairo_pattern_t
* pattern);
extern void
cairo_raster_source_pattern_set_acquire(cairo_pattern_t *
                                         pattern,

cairo_raster_source_acquire_func_t

                                         acquire,

cairo_raster_source_release_func_t

                                         release);
extern void
cairo_raster_source_pattern_set_callback_data(cairo_pattern_t *
                                         pattern,
                                         void *data);
extern void cairo_raster_source_pattern_set_copy(cairo_pattern_t *
pattern,

cairo_raster_source_copy_func_t

                                         copy);
extern void cairo_raster_source_pattern_set_finish(cairo_pattern_t
*

                                         pattern,

cairo_raster_source_finish_func_t

                                         finish);
extern void
cairo_raster_source_pattern_set_snapshot(cairo_pattern_t *
                                         pattern,

cairo_raster_source_snapshot_func_t

                                         snapshot);
extern cairo_surface_t
*cairo_recording_surface_create(cairo_content_t
                                content,
                                const
                                cairo_rectangle_t *
                                extents);
extern cairo_bool_t
cairo_recording_surface_get_extents(cairo_surface_t *
                                    surface,
                                    cairo_rectangle_t *

```

```

                                extents);
extern void cairo_recording_surface_ink_extents(cairo_surface_t *
surface,
                                double *x0, double *y0,
                                double *width,
                                double *height);
extern void cairo_rectangle(cairo_t * cr, double x, double y, double
width,
                                double height);
extern void cairo_rectangle_list_destroy(cairo_rectangle_list_t *
rectangle_list);
extern cairo_t *cairo_reference(cairo_t * cr);
extern cairo_bool_t cairo_region_contains_point(const
cairo_region_t *
                                region, int x, int y);
extern
                                cairo_region_overlap_t
cairo_region_contains_rectangle(const
                                cairo_region_t
                                * region,
                                const
                                cairo_rectangle_int_t
                                * rectangle);
extern cairo_region_t *cairo_region_copy(const cairo_region_t *
original);
extern cairo_region_t *cairo_region_create(void);
extern cairo_region_t *cairo_region_create_rectangle(const
                                cairo_rectangle_int_t
                                * rectangle);
extern cairo_region_t *cairo_region_create_rectangles(const
                                cairo_rectangle_int_t
                                * rects, int count);
extern void cairo_region_destroy(cairo_region_t * region);
extern cairo_bool_t cairo_region_equal(const cairo_region_t * a,
                                const cairo_region_t * b);
extern void cairo_region_get_extents(const cairo_region_t * region,
                                cairo_rectangle_int_t * extents);
extern void cairo_region_get_rectangle(const cairo_region_t *
region,
                                int nth,
                                cairo_rectangle_int_t
                                *
                                rectangle);
extern cairo_status_t cairo_region_intersect(cairo_region_t * dst,
                                const cairo_region_t * other);
extern
                                cairo_status_t
cairo_region_intersect_rectangle(cairo_region_t *
                                dst,
                                const
                                cairo_rectangle_int_t
                                * rectangle);
extern cairo_bool_t cairo_region_is_empty(const cairo_region_t *
region);
extern int cairo_region_num_rectangles(const cairo_region_t *
region);
extern cairo_region_t *cairo_region_reference(cairo_region_t *
region);
extern cairo_status_t cairo_region_status(const cairo_region_t *
region);
extern cairo_status_t cairo_region_subtract(cairo_region_t * dst,
                                const cairo_region_t * other);
extern
                                cairo_status_t
cairo_region_subtract_rectangle(cairo_region_t * dst,
                                const
                                cairo_rectangle_int_t
                                * rectangle);

```

```

cairo_rectangle_int_t
                                * rectangle);
extern void cairo_region_translate(cairo_region_t * region, int dx,
                                int dy);
extern cairo_status_t cairo_region_union(cairo_region_t * dst,
                                const cairo_region_t * other);
extern cairo_status_t cairo_region_union_rectangle(cairo_region_t
* dst,
                                const
                                cairo_rectangle_int_t *
                                rectangle);
extern cairo_status_t cairo_region_xor(cairo_region_t * dst,
                                const cairo_region_t * other);
extern cairo_status_t cairo_region_xor_rectangle(cairo_region_t *
dst,
                                const
                                cairo_rectangle_int_t *
                                rectangle);
extern void cairo_rel_curve_to(cairo_t * cr, double dx1, double dy1,
                                double dx2, double dy2, double dx3,
                                double dy3);
extern void cairo_rel_line_to(cairo_t * cr, double dx, double dy);
extern void cairo_rel_move_to(cairo_t * cr, double dx, double dy);
extern void cairo_reset_clip(cairo_t * cr);
extern void cairo_restore(cairo_t * cr);
extern void cairo_rotate(cairo_t * cr, double angle);
extern void cairo_save(cairo_t * cr);
extern void cairo_scale(cairo_t * cr, double sx, double sy);
extern
                                cairo_scaled_font_t
*cairo_scaled_font_create(cairo_font_face_t *
                                font_face,
                                const cairo_matrix_t *
                                font_matrix,
                                const cairo_matrix_t *
                                ctm,
                                const
                                cairo_font_options_t
*
                                options);
extern void cairo_scaled_font_destroy(cairo_scaled_font_t *
scaled_font);
extern void cairo_scaled_font_extents(cairo_scaled_font_t *
scaled_font,
                                cairo_font_extents_t * extents);
extern void cairo_scaled_font_get_ctm(cairo_scaled_font_t *
scaled_font,
                                cairo_matrix_t * ctm);
extern cairo_font_face_t
*cairo_scaled_font_get_font_face(cairo_scaled_font_t
scaled_font);
extern void cairo_scaled_font_get_font_matrix(cairo_scaled_font_t
*
                                scaled_font,
                                cairo_matrix_t *
                                font_matrix);
extern void cairo_scaled_font_get_font_options(cairo_scaled_font_t
*
                                scaled_font,
                                cairo_font_options_t *
                                options);

extern unsigned int
cairo_scaled_font_get_reference_count(cairo_scaled_font_t *
scaled_font);
extern void cairo_scaled_font_get_scale_matrix(cairo_scaled_font_t
*

```

```

scaled_font,
cairo_matrix_t *
scale_matrix);
extern cairo_font_type_t
cairo_scaled_font_get_type(cairo_scaled_font_t *
scaled_font);
extern void *cairo_scaled_font_get_user_data(cairo_scaled_font_t *
scaled_font,
const cairo_user_data_key_t
*
key);
extern void cairo_scaled_font_glyph_extents(cairo_scaled_font_t *
scaled_font,
const cairo_glyph_t * glyphs,
int num_glyphs,
cairo_text_extents_t *
extents);
extern cairo_scaled_font_t
*cairo_scaled_font_reference(cairo_scaled_font_t
* scaled_font);
extern cairo_status_t
cairo_scaled_font_set_user_data(cairo_scaled_font_t *
scaled_font,
const
cairo_user_data_key_t
* key,
void *user_data,
cairo_destroy_func_t
destroy);
extern cairo_status_t cairo_scaled_font_status(cairo_scaled_font_t
*
scaled_font);
extern void cairo_scaled_font_text_extents(cairo_scaled_font_t *
scaled_font, const char *utf8,
cairo_text_extents_t
*
extents);
extern cairo_status_t
cairo_scaled_font_text_to_glyphs(cairo_scaled_font_t
* scaled_font,
double x, double y,
const char *utf8,
int utf8_len,
cairo_glyph_t *
*glyphs,
int *num_glyphs,
cairo_text_cluster_t
* *clusters,
int *num_clusters,
cairo_text_cluster_flags_t
* cluster_flags);
extern void cairo_select_font_face(cairo_t * cr, const char *family,
cairo_font_slant_t slant,
cairo_font_weight_t weight);
extern void cairo_set_antialias(cairo_t * cr, cairo_antialias_t
antialias);
extern void cairo_set_dash(cairo_t * cr, const double *dashes,
int num_dashes, double offset);
extern void cairo_set_fill_rule(cairo_t * cr, cairo_fill_rule_t
fill_rule);
extern void cairo_set_font_face(cairo_t * cr,
cairo_font_face_t * font_face);
extern void cairo_set_font_matrix(cairo_t * cr,
const cairo_matrix_t * matrix);

```



```

extern void cairo_set_font_options(cairo_t * cr,
                                   const cairo_font_options_t * options);
extern void cairo_set_font_size(cairo_t * cr, double size);
extern void cairo_set_line_cap(cairo_t * cr, cairo_line_cap_t
line_cap);
extern void cairo_set_line_join(cairo_t * cr, cairo_line_join_t
line_join);
extern void cairo_set_line_width(cairo_t * cr, double width);
extern void cairo_set_matrix(cairo_t * cr, const cairo_matrix_t *
matrix);
extern void cairo_set_miter_limit(cairo_t * cr, double limit);
extern void cairo_set_operator(cairo_t * cr, cairo_operator_t op);
extern void cairo_set_scaled_font(cairo_t * cr,
                                   const cairo_scaled_font_t *
scaled_font);
extern void cairo_set_source(cairo_t * cr, cairo_pattern_t *
source);
extern void cairo_set_source_rgb(cairo_t * cr, double red, double
green,
                                double blue);
extern void cairo_set_source_rgba(cairo_t * cr, double red, double
green,
                                double blue, double alpha);
extern void cairo_set_source_surface(cairo_t * cr,
                                     cairo_surface_t * surface, double x,
                                     double y);
extern void cairo_set_tolerance(cairo_t * cr, double tolerance);
extern cairo_status_t cairo_set_user_data(cairo_t * cr,
                                           const cairo_user_data_key_t *
                                           key, void *user_data,
                                           cairo_destroy_func_t destroy);
extern void cairo_show_glyphs(cairo_t * cr, const cairo_glyph_t *
glyphs,
                              int num_glyphs);
extern void cairo_show_page(cairo_t * cr);
extern void cairo_show_text(cairo_t * cr, const char *utf8);
extern void cairo_show_text_glyphs(cairo_t * cr, const char *utf8,
                                    int utf8_len,
                                    const cairo_glyph_t * glyphs,
                                    int num_glyphs,
                                    const cairo_text_cluster_t * clusters,
                                    int num_clusters,
                                    cairo_text_cluster_flags_t
                                    cluster_flags);
extern cairo_status_t cairo_status(cairo_t * cr);
extern const char *cairo_status_to_string(cairo_status_t status);
extern void cairo_stroke(cairo_t * cr);
extern void cairo_stroke_extents(cairo_t * cr, double *x1, double
*y1,
                                double *x2, double *y2);
extern void cairo_stroke_preserve(cairo_t * cr);
extern void cairo_surface_copy_page(cairo_surface_t * surface);
extern void cairo_surface_create_for_rectangle(cairo_surface_t
*target,
                                              double x,
                                              double y,
                                              double width,
                                              double height);
extern void cairo_surface_create_observer(cairo_surface_t *
target,
                                          cairo_surface_observer_mode_t
                                          mode);

```

```

extern cairo_surface_t
*cairo_surface_create_similar(cairo_surface_t *
                             other,
                             cairo_content_t
                             content, int width,
                             int height);

extern cairo_surface_t
*cairo_surface_create_similar_image(cairo_surface_t
                                   * other,
                                   cairo_format_t
                                   format,
                                   int width,
                                   int height);

extern void cairo_surface_destroy(cairo_surface_t * surface);
extern void cairo_surface_finish(cairo_surface_t * surface);
extern void cairo_surface_flush(cairo_surface_t * surface);
extern cairo_content_t cairo_surface_get_content(cairo_surface_t *
                                                surface);

extern cairo_device_t *cairo_surface_get_device(cairo_surface_t *
surface);
extern void cairo_surface_get_device_offset(cairo_surface_t *
surface,
                                           double *x_offset,
                                           double *y_offset);

extern void cairo_surface_get_fallback_resolution(cairo_surface_t
*
                                                surface,
                                                double
                                                *x_pixels_per_inch,
                                                double
                                                *y_pixels_per_inch);

extern void cairo_surface_get_font_options(cairo_surface_t *
surface,
                                           cairo_font_options_t
                                           *
                                           options);
extern void cairo_surface_get_mime_data(cairo_surface_t * surface,
                                       const char *mime_type,
                                       const unsigned char **data,
                                       unsigned long int *length);

extern unsigned int
cairo_surface_get_reference_count(cairo_surface_t *
surface);
extern cairo_surface_type_t cairo_surface_get_type(cairo_surface_t
*
                                                surface);

extern void *cairo_surface_get_user_data(cairo_surface_t * surface,
                                       const cairo_user_data_key_t *
                                       key);

extern cairo_bool_t
cairo_surface_has_show_text_glyphs(cairo_surface_t *
surface);
extern cairo_surface_t *cairo_surface_map_to_image(cairo_surface_t
*
                                                surface,
                                                const
                                                cairo_rectangle_int_t *
                                                extents);

extern void cairo_surface_mark_dirty(cairo_surface_t * surface);
extern void cairo_surface_mark_dirty_rectangle(cairo_surface_t *
surface,
                                               int x, int y, int width,
                                               int height);

extern cairo_status_t
cairo_surface_observer_add_fill_callback(cairo_surface_t *
abstract_surface,

```

```

cairo_surface_observer_callback_t
                                func, void *data);
extern cairo_status_t
cairo_surface_observer_add_finish_callback(cairo_surface_t *
                                abstract_surface,

cairo_surface_observer_callback_t
                                func, void *data);
extern cairo_status_t
cairo_surface_observer_add_flush_callback(cairo_surface_t *
                                abstract_surface,

cairo_surface_observer_callback_t
                                func, void *data);
extern cairo_status_t
cairo_surface_observer_add_glyphs_callback(cairo_surface_t *
                                abstract_surface,

cairo_surface_observer_callback_t
                                func, void *data);
extern cairo_status_t
cairo_surface_observer_add_mask_callback(cairo_surface_t *
                                abstract_surface,

cairo_surface_observer_callback_t
                                func, void *data);
extern cairo_status_t
cairo_surface_observer_add_paint_callback(cairo_surface_t *
                                abstract_surface,

cairo_surface_observer_callback_t
                                func, void *data);
extern cairo_status_t
cairo_surface_observer_add_stroke_callback(cairo_surface_t *
                                abstract_surface,

cairo_surface_observer_callback_t
                                func, void *data);
extern double cairo_surface_observer_elapsed(cairo_surface_t *
surface);
extern cairo_status_t cairo_surface_observer_print(cairo_surface_t
*
                                surface,
                                cairo_write_func_t
                                write_func,
                                void *closure);
extern cairo_surface_t *cairo_surface_reference(cairo_surface_t *
surface);
extern void cairo_surface_set_device_offset(cairo_surface_t *
surface,
                                double x_offset,
                                double y_offset);
extern void cairo_surface_set_fallback_resolution(cairo_surface_t
*
                                surface,
                                double x_pixels_per_inch,
                                double
                                y_pixels_per_inch);
extern cairo_status_t cairo_surface_set_mime_data(cairo_surface_t
*
                                surface,
                                const char *mime_type,
                                const unsigned char
                                *data,
                                unsigned long int length,

```

```

        cairo_destroy_func_t
        destroy, void *closure);
extern cairo_status_t cairo_surface_set_user_data(cairo_surface_t
*
        surface,
        const
        cairo_user_data_key_t *
        key, void *user_data,
        cairo_destroy_func_t
        destroy);
extern void cairo_surface_show_page(cairo_surface_t * surface);
extern cairo_status_t cairo_surface_status(cairo_surface_t *
surface);
extern
        cairo_bool_t
cairo_surface_supports_mime_type(cairo_surface_t *
        surface,
        const char
        *mime_type);
extern void cairo_surface_unmap_image(cairo_surface_t * surface,
        cairo_surface_t * image);
extern cairo_status_t cairo_surface_write_to_png(cairo_surface_t *
surface,
        const char *filename);
extern
        cairo_status_t
cairo_surface_write_to_png_stream(cairo_surface_t *
        surface,
        cairo_write_func_t
        write_func,
        void *closure);
extern
        cairo_text_cluster_t
        *cairo_text_cluster_allocate(int
num_clusters);
extern void
        cairo_text_cluster_free(cairo_text_cluster_t
clusters);
extern void cairo_text_extents(cairo_t * cr, const char *utf8,
        cairo_text_extents_t * extents);
extern void cairo_text_path(cairo_t * cr, const char *utf8);
extern
        cairo_font_face_t
        *cairo_font_face_create(const char
*family,
        cairo_font_slant_t
        slant,
        cairo_font_weight_t
        weight);
extern
        const
        char
*cairo_font_face_get_family(cairo_font_face_t *
        font_face);
extern
        cairo_font_slant_t
        *
        cairo_font_face_get_slant(cairo_font_face_t *
        font_face);
extern
        cairo_font_weight_t
        *
        cairo_font_face_get_weight(cairo_font_face_t
        * font_face);
extern void cairo_transform(cairo_t * cr, const cairo_matrix_t *
matrix);
extern void cairo_translate(cairo_t * cr, double tx, double ty);
extern
        cairo_font_face_t
        *cairo_user_font_face_create(void);
extern
        cairo_user_scaled_font_init_func_t
        *
        cairo_user_font_face_get_init_func(cairo_font_face_t * font_face);
extern
        cairo_user_scaled_font_render_glyph_func_t
        *
        cairo_user_font_face_get_render_glyph_func(cairo_font_face_t
        * font_face);
extern
        cairo_user_scaled_font_text_to_glyphs_func_t
        *
        cairo_user_font_face_get_text_to_glyphs_func(cairo_font_face_t *
        font_face);
extern
        cairo_user_scaled_font_unicode_to_glyph_func_t
        *
        cairo_user_font_face_get_unicode_to_glyph_func(cairo_font_face_t *
        font_face);

```

```

extern void cairo_user_font_face_set_init_func(cairo_font_face_t *
                                              font_face,

cairo_user_scaled_font_init_func_t
                                              init_func);
extern
cairo_user_font_face_set_render_glyph_func(cairo_font_face_t *
                                           font_face,
                                           void

cairo_user_scaled_font_render_glyph_func_t
                                           render_glyph_func);
extern
cairo_user_font_face_set_text_to_glyphs_func(cairo_font_face_t
                                             * font_face,
                                             void

cairo_user_scaled_font_text_to_glyphs_func_t

text_to_glyphs_func);
extern void
cairo_user_font_face_set_unicode_to_glyph_func(cairo_font_face_t *
                                              font_face,

cairo_user_scaled_font_unicode_to_glyph_func_t
                                              unicode_to_glyph_func);
extern void cairo_user_to_device(cairo_t * cr, double *x, double
*y);
extern void cairo_user_to_device_distance(cairo_t * cr, double *dx,
double *dy);
extern int cairo_version(void);
extern const char *cairo_version_string(void);

```

15.3 Interface Definitions for libcairo

The interfaces defined on the following pages are included in libcairo and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 15.1 shall behave as described in the referenced base document.

XI tiff library

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

16 Libraries

16.1 Interfaces for libtiff

Table 16-1 defines the library name and shared object name for the libtiff library

Table 16-1 libtiff Definition

Library:	libtiff
SONAME:	libtiff.so.5

The behavior of the interfaces in this library is specified by the following specifications:

[Libtiff 4.0.2] Libtiff 4.0.2 Reference Manual

16.1.1 libtiff interfaces

16.1.1.1 Interfaces for libtiff interfaces

An LSB conforming implementation shall provide the generic functions for libtiff interfaces specified in Table 16-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 16-2 libtiff - libtiff interfaces Function Interfaces

TIFFAccessTagMethods [Libtiff 4.0.2]	TIFFCIELabToRGBInit [Libtiff 4.0.2]	TIFFCIELabToXYZ [Libtiff 4.0.2]
TIFFCheckTile [Libtiff 4.0.2]	TIFFCheckpointDirectory [Libtiff 4.0.2]	TIFFCleanup [Libtiff 4.0.2]
TIFFClientOpen [Libtiff 4.0.2]	TIFFClientdata [Libtiff 4.0.2]	TIFFClose [Libtiff 4.0.2]
TIFFComputeStrip [Libtiff 4.0.2]	TIFFComputeTile [Libtiff 4.0.2]	TIFFCreateCustomDirectory [Libtiff 4.0.2]
TIFFCreateDirectory [Libtiff 4.0.2]	TIFFCreateEXIFDirectory [Libtiff 4.0.2]	TIFFCurrentDirOffset [Libtiff 4.0.2]
TIFFCurrentDirectory [Libtiff 4.0.2]	TIFFCurrentRow [Libtiff 4.0.2]	TIFFCurrentStrip [Libtiff 4.0.2]
TIFFCurrentTile [Libtiff 4.0.2]	TIFFDataWidth [Libtiff 4.0.2]	TIFFDefaultStripSize [Libtiff 4.0.2]
TIFFDefaultTileSize [Libtiff 4.0.2]	TIFFError [Libtiff 4.0.2]	TIFFErrorExt [Libtiff 4.0.2]
TIFFFdOpen [Libtiff 4.0.2]	TIFFFieldWithName [Libtiff 4.0.2]	TIFFFieldWithTag [Libtiff 4.0.2]
TIFFFileName [Libtiff 4.0.2]	TIFFFileno [Libtiff 4.0.2]	TIFFFindCODEC [Libtiff 4.0.2]
TIFFFindField [Libtiff 4.0.2]	TIFFFlush [Libtiff 4.0.2]	TIFFFlushData [Libtiff 4.0.2]

TIFFFreeDirectory [Libtiff 4.0.2]	TIFFGetBitRevTable [Libtiff 4.0.2]	TIFFGetClientInfo [Libtiff 4.0.2]
TIFFGetCloseProc [Libtiff 4.0.2]	TIFFGetConfiguredCO DECs [Libtiff 4.0.2]	TIFFGetField [Libtiff 4.0.2]
TIFFGetFieldDefaulted [Libtiff 4.0.2]	TIFFGetMapFileProc [Libtiff 4.0.2]	TIFFGetMode [Libtiff 4.0.2]
TIFFGetReadProc [Libtiff 4.0.2]	TIFFGetSeekProc [Libtiff 4.0.2]	TIFFGetSizeProc [Libtiff 4.0.2]
TIFFGetTagListCount [Libtiff 4.0.2]	TIFFGetTagListEntry [Libtiff 4.0.2]	TIFFGetUnmapFileProc [Libtiff 4.0.2]
TIFFGetVersion [Libtiff 4.0.2]	TIFFGetWriteProc [Libtiff 4.0.2]	TIFFIsBigEndian [Libtiff 4.0.2]
TIFFIsByteSwapped [Libtiff 4.0.2]	TIFFIsCODECConfigur ed [Libtiff 4.0.2]	TIFFIsMSB2LSB [Libtiff 4.0.2]
TIFFIsTiled [Libtiff 4.0.2]	TIFFIsUpSampled [Libtiff 4.0.2]	TIFFLastDirectory [Libtiff 4.0.2]
TIFFNumberOfDirector ies [Libtiff 4.0.2]	TIFFNumberOfStrips [Libtiff 4.0.2]	TIFFNumberOfTiles [Libtiff 4.0.2]
TIFFOpen [Libtiff 4.0.2]	TIFFPrintDirectory [Libtiff 4.0.2]	TIFFRGBAImageBegin [Libtiff 4.0.2]
TIFFRGBAImageEnd [Libtiff 4.0.2]	TIFFRGBAImageGet [Libtiff 4.0.2]	TIFFRGBAImageOK [Libtiff 4.0.2]
TIFFRasterScanlineSize [Libtiff 4.0.2]	TIFFRasterScanlineSize 64 [Libtiff 4.0.2]	TIFFRawStripSize [Libtiff 4.0.2]
TIFFRawStripSize64 [Libtiff 4.0.2]	TIFFReadBufferSetup [Libtiff 4.0.2]	TIFFReadCustomDirect ory [Libtiff 4.0.2]
TIFFReadDirectory [Libtiff 4.0.2]	TIFFReadEXIFDirectory [Libtiff 4.0.2]	TIFFReadEncodedStrip [Libtiff 4.0.2]
TIFFReadEncodedTile [Libtiff 4.0.2]	TIFFReadRGBAImage [Libtiff 4.0.2]	TIFFReadRGBAImageO riented [Libtiff 4.0.2]
TIFFReadRGBAStrip [Libtiff 4.0.2]	TIFFReadRGBATile [Libtiff 4.0.2]	TIFFReadRawStrip [Libtiff 4.0.2]
TIFFReadRawTile [Libtiff 4.0.2]	TIFFReadScanline [Libtiff 4.0.2]	TIFFReadTile [Libtiff 4.0.2]
TIFFRegisterCODEC [Libtiff 4.0.2]	TIFFReverseBits [Libtiff 4.0.2]	TIFFRewriteDirectory [Libtiff 4.0.2]
TIFFScanlineSize [Libtiff 4.0.2]	TIFFScanlineSize64 [Libtiff 4.0.2]	TIFFSetClientInfo [Libtiff 4.0.2]
TIFFSetClientdata [Libtiff 4.0.2]	TIFFSetDirectory [Libtiff 4.0.2]	TIFFSetErrorHandler [Libtiff 4.0.2]
TIFFSetErrorHandlerEx t [Libtiff 4.0.2]	TIFFSetField [Libtiff 4.0.2]	TIFFSetFileName [Libtiff 4.0.2]

TIFFSetFileno [Libtiff 4.0.2]	TIFFSetMode [Libtiff 4.0.2]	TIFFSetSubDirectory [Libtiff 4.0.2]
TIFFSetTagExtender [Libtiff 4.0.2]	TIFFSetWarningHandler [Libtiff 4.0.2]	TIFFSetWarningHandlerExt [Libtiff 4.0.2]
TIFFSetWriteOffset [Libtiff 4.0.2]	TIFFSetupStrips [Libtiff 4.0.2]	TIFFStripSize [Libtiff 4.0.2]
TIFFStripSize64 [Libtiff 4.0.2]	TIFFSwabArrayOfDouble [Libtiff 4.0.2]	TIFFSwabArrayOfFloat [Libtiff 4.0.2]
TIFFSwabArrayOfLong [Libtiff 4.0.2]	TIFFSwabArrayOfLong8 [Libtiff 4.0.2]	TIFFSwabArrayOfShort [Libtiff 4.0.2]
TIFFSwabArrayOfTriples [Libtiff 4.0.2]	TIFFSwabDouble [Libtiff 4.0.2]	TIFFSwabFloat [Libtiff 4.0.2]
TIFFSwabLong [Libtiff 4.0.2]	TIFFSwabLong8 [Libtiff 4.0.2]	TIFFSwabShort [Libtiff 4.0.2]
TIFFTileRowSize [Libtiff 4.0.2]	TIFFTileRowSize64 [Libtiff 4.0.2]	TIFFTileSize [Libtiff 4.0.2]
TIFFTileSize64 [Libtiff 4.0.2]	TIFFUnRegisterCODEC [Libtiff 4.0.2]	TIFFUnlinkDirectory [Libtiff 4.0.2]
TIFFUnsetField [Libtiff 4.0.2]	TIFFVGetField [Libtiff 4.0.2]	TIFFVGetFieldDefaulted [Libtiff 4.0.2]
TIFFVSetField [Libtiff 4.0.2]	TIFFVStripSize [Libtiff 4.0.2]	TIFFVStripSize64 [Libtiff 4.0.2]
TIFFVTileSize [Libtiff 4.0.2]	TIFFVTileSize64 [Libtiff 4.0.2]	TIFFWarning [Libtiff 4.0.2]
TIFFWarningExt [Libtiff 4.0.2]	TIFFWriteBufferSetup [Libtiff 4.0.2]	TIFFWriteCheck [Libtiff 4.0.2]
TIFFWriteCustomDirectory [Libtiff 4.0.2]	TIFFWriteDirectory [Libtiff 4.0.2]	TIFFWriteEncodedStrip [Libtiff 4.0.2]
TIFFWriteEncodedTile [Libtiff 4.0.2]	TIFFWriteRawStrip [Libtiff 4.0.2]	TIFFWriteRawTile [Libtiff 4.0.2]
TIFFWriteScanline [Libtiff 4.0.2]	TIFFWriteTile [Libtiff 4.0.2]	TIFFXYZToRGB [Libtiff 4.0.2]
TIFFYCbCrToRGBInit [Libtiff 4.0.2]	TIFFYCbCrToRGB [Libtiff 4.0.2]	

16.2 Data Definitions for libtiff

This section defines global identifiers and their values that are associated with interfaces contained in libtiff. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and

application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

16.2.1 tiff.h

```
#define CLEANFAXDATA_CLEAN      0
#define DCSIMAGERFILTER_IR      0
#define DCSIMAGERMODEL_M3      0
#define EXTRASAMPLE_UNSPECIFIED 0
#define PERSAMPLE_MERGED        0
#define PHOTOMETRIC_MINISWHITE  0
#define PIXARLOGDATAFORMAT_8BIT 0
#define SGILOGDATAFORMAT_FLOAT  0
#define SGILOGENCODE_NODITHER  0
#define DCSINTERPMODE_NORMAL    0x0
#define FAXMODE_CLASSIC 0x0000
#define JPEGCOLORMODE_RAW       0x0000
#define FAXMODE_NORTC 0x0001
#define JPEGCOLORMODE_RGB       0x0001
#define JPEGTABLESMODE_QUANT    0x0001
#define FAXMODE_NOEOL 0x0002
#define JPEGTABLESMODE_HUFF     0x0002
#define FAXMODE_BYTEALIGN      0x0004
#define FAXMODE_WORDALIGN      0x0008
#define DCSINTERPMODE_PREVIEW  0x1
#define FILETYPE_REDUCEDIMAGE  0x1
#define GROUP3OPT_2DENCODING    0x1
#define FILETYPE_PAGE 0x2
#define GROUP3OPT_UNCOMPRESSED  0x2
#define GROUP4OPT_UNCOMPRESSED  0x2
#define FILETYPE_MASK 0x4
#define GROUP3OPT_FILLBITS      0x4
#define MDI_BIGENDIAN 0x4550
#define TIFF_LITTLEENDIAN      0x4949
#define TIFF_BIGENDIAN 0x4d4d
#define MDI_LITTLEENDIAN      0x5045
#define CLEANFAXDATA_REGENERATED 1
#define COLORRESPONSEUNIT_10S  1
#define COMPRESSION_NONE        1
#define DCSIMAGERFILTER_MONO    1
#define DCSIMAGERMODEL_M5      1
#define EXTRASAMPLE_ASSOCALPHA  1
#define FILLORDER_MSB2LSB       1
#define GRAYRESPONSEUNIT_10S    1
#define INKSET_CMYK 1
#define JPEGPROC_BASELINE       1
#define OFILETYPE_IMAGE 1
#define ORIENTATION_TOPLEFT     1
#define PERSAMPLE_MULTI 1
#define PHOTOMETRIC_MINISBLACK  1
#define PIXARLOGDATAFORMAT_8BITABGR 1
#define PLANARCONFIG_CONTIG     1
#define PREDICTOR_NONE 1
#define RESUNIT_NONE 1
#define SAMPLEFORMAT_UINT       1
#define SGILOGDATAFORMAT_16BIT  1
#define SGILOGENCODE_RANDITHER  1
#define THRESHOLD_BILEVEL       1
#define YCBCRPOSITION_CENTERED  1
```

```

#define PHOTOMETRIC_ITULAB      10
#define JPEGPROC_LOSSLESS      14
#define CLEANFAXDATA_UNCLEAN    2
#define COLORRESPONSEUNIT_100S  2
#define COMPRESSION_CCITTRLE    2
#define DCSIMAGERFILTER_CFA      2
#define DCSIMAGERMODEL_M6        2
#define EXTRASAMPLE_UNASSALPHA   2
#define FILLORDER_LSB2MSB        2
#define GRAYRESPONSEUNIT_100S    2
#define INKSET_MULTINK           2
#define OFILETYPE_REDUCEDIMAGE   2
#define ORIENTATION_TOPRIGHT     2
#define PHOTOMETRIC_RGB          2
#define PIXARLOGDATAFMT_11BITLOG 2
#define PLANARCONFIG_SEPARATE     2
#define PREDICTOR_HORIZONTAL      2
#define RESUNIT_INCH             2
#define SAMPLEFORMAT_INT          2
#define SGILOGDATAFMT_RAW         2
#define THRESHHOLD_HALFTONE       2
#define YCBCRPOSITION_COSITED     2
#define TIFFTAG_SUBFILETYPE       254
#define TIFFTAG_OSUBFILETYPE      255
#define TIFFTAG_IMAGEWIDTH        256
#define TIFFTAG_IMAGELENGTH       257
#define TIFFTAG_BITSPERSAMPLE     258
#define TIFFTAG_COMPRESSION       259
#define TIFFTAG_PHOTOMETRIC       262
#define TIFFTAG_THRESHHOLDING     263
#define TIFFTAG_CELLWIDTH        264
#define TIFFTAG_CELLENGTH        265
#define TIFFTAG_FILLORDER         266
#define TIFFTAG_DOCUMENTNAME      269
#define TIFFTAG_IMAGEDESCRIPTION  270
#define TIFFTAG_MAKE              271
#define TIFFTAG_MODEL             272
#define TIFFTAG_STRIPOFFSETS      273
#define TIFFTAG_ORIENTATION       274
#define TIFFTAG_SAMPLESPERPIXEL   277
#define TIFFTAG_ROWSPERSTRIP      278
#define TIFFTAG_STRIPBYTECOUNTS  279
#define TIFFTAG_MINSAMPLEVALUE    280
#define TIFFTAG_MAXSAMPLEVALUE    281
#define TIFFTAG_XRESOLUTION       282
#define TIFFTAG_YRESOLUTION       283
#define TIFFTAG_PLANARCONFIG      284
#define TIFFTAG_PAGENAME          285
#define TIFFTAG_XPOSITION         286
#define TIFFTAG_YPOSITION         287
#define TIFFTAG_FREEOFFSETS       288
#define TIFFTAG_FREEBYTECOUNTS   289
#define TIFFTAG_GRAYRESPONSEUNIT  290
#define TIFFTAG_GRAYRESPONSECURVE 291
#define TIFFTAG_GROUP3OPTIONS     292
#define TIFFTAG_T4OPTIONS         292
#define TIFFTAG_GROUP4OPTIONS     293
#define TIFFTAG_T6OPTIONS         293
#define TIFFTAG_RESOLUTIONUNIT    296
#define TIFFTAG_PAGENUMBER        297
#define COLORRESPONSEUNIT_1000S  3
#define COMPRESSION_CCITTFAX3     3
#define COMPRESSION_CCITT_T4      3
#define DCSIMAGERFILTER_OTHER     3
#define GRAYRESPONSEUNIT_1000S    3
#define OFILETYPE_PAGE            3

```

```

#define ORIENTATION_BOTRIGHT      3
#define PHOTOMETRIC_PALETTE        3
#define PIXARLOGDATAFORMAT_12BITPCIO 3
#define PREDICTOR_FLOATINGPOINT    3
#define RESUNIT_CENTIMETER         3
#define SAMPLEFORMAT_IEEEFP        3
#define SGILOGDATAFORMAT_8BIT      3
#define THRESHHOLD_ERRORDIFFUSE    3
#define TIFFTAG_COLORRESPONSEUNIT  300
#define TIFFTAG_TRANSFERFUNCTION    301
#define TIFFTAG_SOFTWARE            305
#define TIFFTAG_DATETIME            306
#define TIFFTAG_ARTIST              315
#define TIFFTAG_HOSTCOMPUTER        316
#define TIFFTAG_PREDICTOR           317
#define TIFFTAG_WHITEPOINT          318
#define TIFFTAG_PRIMARYCHROMATICITIES 319
#define TIFFTAG_COLORMAP            320
#define TIFFTAG_HALFTONEHINTS       321
#define TIFFTAG_TILEWIDTH           322
#define TIFFTAG_TILELENGTH          323
#define TIFFTAG_TILEOFFSETS         324
#define TIFFTAG_TILEBYTECOUNTS     325
#define TIFFTAG_BADFAXLINES         326
#define TIFFTAG_CLEANFAXDATA        327
#define COMPRESSION_NEXT            32766
#define COMPRESSION_CCITTRLEW       32771
#define COMPRESSION_PACKBITS        32773
#define TIFFTAG_OPIIMAGEID          32781
#define TIFFTAG_CONSECUTIVEBADFAXLINES 328
#define COMPRESSION_THUNDERSCAN     32809
#define PHOTOMETRIC_LOGL            32844
#define PHOTOMETRIC_LOGLUV          32845
#define COMPRESSION_IT8CTPAD        32895
#define COMPRESSION_IT8LW           32896
#define COMPRESSION_IT8MP           32897
#define COMPRESSION_IT8BL           32898
#define COMPRESSION_PIXARFILM       32908
#define COMPRESSION_PIXARLOG        32909
#define COMPRESSION_DEFLATE         32946
#define COMPRESSION_DCS             32947
#define TIFFTAG_REFPTS              32953
#define TIFFTAG_REGIONTACKPOINT     32954
#define TIFFTAG_REGIONWARPCORNERS   32955
#define TIFFTAG_REGIONAFFINE        32956
#define TIFFTAG_MATTEING            32995
#define TIFFTAG_DATATYPE            32996
#define TIFFTAG_IMAGEDEPTH          32997
#define TIFFTAG_TILEDEPTH           32998
#define TIFFTAG_SUBIFD              330
#define TIFFTAG_INKSET              332
#define TIFFTAG_INKNAMES            333
#define TIFFTAG_PIXAR_IMAGEFULLWIDTH 33300
#define TIFFTAG_PIXAR_IMAGEFULLLENGTH 33301
#define TIFFTAG_PIXAR_TEXTUREFORMAT 33302
#define TIFFTAG_PIXAR_WRAPMODES     33303
#define TIFFTAG_PIXAR_FOVCOT        33304
#define TIFFTAG_PIXAR_MATRIX_WORLDTOSCREEN 33305
#define TIFFTAG_PIXAR_MATRIX_WORLDTOCAMERA 33306
#define TIFFTAG_NUMBEROFINKS        334
#define TIFFTAG_WRITERSERIALNUMBER  33405
#define TIFFTAG_COPYRIGHT           33432
#define EXIFTAG_EXPOSURETIME        33434
#define EXIFTAG_FNUMBER             33437
#define TIFFTAG_DOTRANGE            336
#define TIFFTAG_TARGETPRINTER       337

```

```

#define TIFFTAG_RICHTIFFIPTC      33723
#define TIFFTAG_EXTRASAMPLES      338
#define TIFFTAG_SAMPLEFORMAT      339
#define TIFFTAG_SMINSAMPLEVALUE   340
#define TIFFTAG_IT8SITE           34016
#define TIFFTAG_IT8COLORSEQUENCE  34017
#define TIFFTAG_IT8HEADER         34018
#define TIFFTAG_IT8RASTERPADDING  34019
#define TIFFTAG_IT8BITSPERRUNLENGTH 34020
#define TIFFTAG_IT8BITSPEREXTENDED RUNLENGTH 34021
#define TIFFTAG_IT8COLORTABLE     34022
#define TIFFTAG_IT8IMAGECOLORINDICATOR 34023
#define TIFFTAG_IT8BKGCOLORINDICATOR 34024
#define TIFFTAG_IT8IMAGECOLORVALUE 34025
#define TIFFTAG_IT8BKGCOLORVALUE  34026
#define TIFFTAG_IT8PIXELINTENSITYRANGE 34027
#define TIFFTAG_IT8TRANSPARENCYINDICATOR 34028
#define TIFFTAG_IT8COLORCHARACTERIZATION 34029
#define TIFFTAG_IT8HCUSAGE        34030
#define TIFFTAG_IT8TRAPINDICATOR  34031
#define TIFFTAG_IT8CMYKEQUIVALENT 34032
#define TIFFTAG_SMAXSAMPLEVALUE   341
#define TIFFTAG_FRAMECOUNT       34232
#define TIFFTAG_CLIPPATH          343
#define TIFFTAG_PHOTOSHOP        34377
#define TIFFTAG_XCLIPPATHUNITS    344
#define TIFFTAG_YCLIPPATHUNITS    345
#define TIFFTAG_INDEXED          346
#define COMPRESSION_JBIG         34661
#define TIFFTAG_EXIFIFD          34665
#define TIFFTAG_ICCPROFILE        34675
#define COMPRESSION_SGILOG        34676
#define COMPRESSION_SGILOG24      34677
#define TIFFTAG_JPEGTABLES        347
#define COMPRESSION_JP2000        34712
#define TIFFTAG_JBIGOPTIONS       34750
#define EXIFTAG_EXPOSUREPROGRAM  34850
#define EXIFTAG_SPECTRAISENSITIVITY 34852
#define TIFFTAG_GPSIFD           34853
#define EXIFTAG_ISOSPEEDRATINGS  34855
#define EXIFTAG_OECF             34856
#define TIFFTAG_FAXRECVPARAMS     34908
#define TIFFTAG_FAXSUBADDRESS      34909
#define TIFFTAG_FAXRCVTIME        34910
#define TIFFTAG_FAXDCS            34911
#define COMPRESSION_LZMA          34925
#define TIFFTAG_FEDEX_EDR          34929
#define TIFFTAG_OPIPROXY          351
#define EXIFTAG_EXIFVERSION       36864
#define EXIFTAG_DATETIMEORIGINAL  36867
#define EXIFTAG_DATETIMEDIGITIZED 36868
#define EXIFTAG_COMPONENTSCONFIGURATION 37121
#define EXIFTAG_COMPRESSEDBITS PERPIXEL 37122
#define EXIFTAG_SHUTTERSPEEDVALUE 37377
#define EXIFTAG_APERTUREVALUE     37378
#define EXIFTAG_BRIGHTNESSVALUE  37379
#define EXIFTAG_EXPOSUREBIASVALUE 37380
#define EXIFTAG_MAXAPERTUREVALUE  37381
#define EXIFTAG_SUBJECTDISTANCE  37382
#define EXIFTAG_METERINGMODE      37383
#define EXIFTAG_LIGHTSOURCE       37384
#define EXIFTAG_FLASH             37385
#define EXIFTAG_FOCALLENGTH       37386
#define EXIFTAG_SUBJECTAREA       37396
#define TIFFTAG_STONITS           37439
#define EXIFTAG_MAKERNOTE         37500

```

```

#define EXIFTAG_USERCOMMENT      37510
#define EXIFTAG_SUBSECTIME      37520
#define EXIFTAG_SUBSECTIMEORIGINAL 37521
#define EXIFTAG_SUBSECTIMEDIGITIZED 37522
#define COLORRESPONSEUNIT_10000S 4
#define COMPRESSION_CCITTFAX4 4
#define COMPRESSION_CCITT_T6 4
#define GRAYRESPONSEUNIT_10000S 4
#define ORIENTATION_BOTLEFT 4
#define PHOTOMETRIC_MASK 4
#define PIXARLOGDATAFORMAT_16BIT 4
#define SAMPLEFORMAT_VOID 4
#define EXIFTAG_FLASHPIXVERSION 40960
#define EXIFTAG_COLORSPACE 40961
#define EXIFTAG_PIXELXDIMENSION 40962
#define EXIFTAG_PIXELYDIMENSION 40963
#define EXIFTAG_RELATEDSOUNDFILE 40964
#define TIFFTAG_INTEROPERABILITYIFD 40965
#define EXIFTAG_FLASHENERGY 41483
#define EXIFTAG_SPATIALFREQUENCYRESPONSE 41484
#define EXIFTAG_FOCALPLANEXRESOLUTION 41486
#define EXIFTAG_FOCALPLANEYRESOLUTION 41487
#define EXIFTAG_FOCALPLANERESOLUTIONUNIT 41488
#define EXIFTAG_SUBJECTLOCATION 41492
#define EXIFTAG_EXPOSUREINDEX 41493
#define EXIFTAG_SENSINGMETHOD 41495
#define EXIFTAG_FILESOURCE 41728
#define EXIFTAG_SCENETYPE 41729
#define EXIFTAG_CFAPATTERN 41730
#define EXIFTAG_CUSTOMRENDERED 41985
#define EXIFTAG_EXPOSUREMODE 41986
#define EXIFTAG_WHITEBALANCE 41987
#define EXIFTAG_DIGITALZOOMRATIO 41988
#define EXIFTAG_FOCALLENGTHIN35MMFILM 41989
#define EXIFTAG_SCENECAPTURETYPE 41990
#define EXIFTAG_GAINCONTROL 41991
#define EXIFTAG_CONTRAST 41992
#define EXIFTAG_SATURATION 41993
#define EXIFTAG_SHARPNESS 41994
#define EXIFTAG_DEVICESETTINGDESCRIPTION 41995
#define EXIFTAG_SUBJECTDISTANCERANGE 41996
#define TIFF_VERSION_CLASSIC 42
#define EXIFTAG_IMAGEUNIQUEID 42016
#define TIFF_VERSION_BIG 43
#define COLORRESPONSEUNIT_100000S 5
#define COMPRESSION_LZW 5
#define GRAYRESPONSEUNIT_100000S 5
#define ORIENTATION_LEFTTOP 5
#define PHOTOMETRIC_SEPARATED 5
#define PIXARLOGDATAFORMAT_FLOAT 5
#define SAMPLEFORMAT_COMPLEXINT 5
#define TIFFTAG_DNGVERSION 50706
#define TIFFTAG_DNGBACKWARDVERSION 50707
#define TIFFTAG_UNIQUECAMERAMODEL 50708
#define TIFFTAG_LOCALIZEDCAMERAMODEL 50709
#define TIFFTAG_CFAPLANECOLOR 50710
#define TIFFTAG_CFA_LAYOUT 50711
#define TIFFTAG_LINEARIZATIONTABLE 50712
#define TIFFTAG_BLACKLEVELREPEATDIM 50713
#define TIFFTAG_BLACKLEVEL 50714
#define TIFFTAG_BLACKLEVELDELTAH 50715
#define TIFFTAG_BLACKLEVELDELTAV 50716
#define TIFFTAG_WHITELEVEL 50717
#define TIFFTAG_DEFAULTSCALE 50718
#define TIFFTAG_DEFAULTCROPOIGIN 50719
#define TIFFTAG_DEFAULTCROPSIZE 50720

```

```

#define TIFFTAG_COLORMATRIX1      50721
#define TIFFTAG_COLORMATRIX2      50722
#define TIFFTAG_CAMERACALIBRATION1 50723
#define TIFFTAG_CAMERACALIBRATION2 50724
#define TIFFTAG_REDUCTIONMATRIX1 50725
#define TIFFTAG_REDUCTIONMATRIX2 50726
#define TIFFTAG_ANALOGBALANCE      50727
#define TIFFTAG_ASSHOTNEUTRAL      50728
#define TIFFTAG_ASSHOTWHITEXY      50729
#define TIFFTAG_BASELINEEXPOSURE    50730
#define TIFFTAG_BASELINENOISE      50731
#define TIFFTAG_BASELINESHARPNESS  50732
#define TIFFTAG_BAYERGREENSPLIT    50733
#define TIFFTAG_LINEARRESPONSELIMIT 50734
#define TIFFTAG_CAMERA_SERIALNUMBER 50735
#define TIFFTAG_LENSINFO            50736
#define TIFFTAG_CHROMABLURRADIUS    50737
#define TIFFTAG_ANTIALIASSTRENGTH   50738
#define TIFFTAG_SHADOWSCALE         50739
#define TIFFTAG_DNGPRIVATEDATA      50740
#define TIFFTAG_MAKERNOTESAFETY     50741
#define TIFFTAG_CALIBRATIONILLUMINANT1 50778
#define TIFFTAG_CALIBRATIONILLUMINANT2 50779
#define TIFFTAG_BESTQUALITYSCALE    50780
#define TIFFTAG_RAWDATAUNIQUEID    50781
#define TIFFTAG_ORIGINALRAWFILENAME 50827
#define TIFFTAG_ORIGINALRAWFILEDATA 50828
#define TIFFTAG_ACTIVEAREA          50829
#define TIFFTAG_MASKEDAREAS         50830
#define TIFFTAG_ASSHOTICCPROFILE    50831
#define TIFFTAG_ASSHOTPREPROFILEMATRIX 50832
#define TIFFTAG_CURRENTICCPROFILE   50833
#define TIFFTAG_CURRENTPREPROFILEMATRIX 50834
#define TIFFTAG_JPEGPROC             512
#define TIFFTAG_JPEGIFOFFSET         513
#define TIFFTAG_JPEGIFBYTECOUNT    514
#define TIFFTAG_JPEGRESTARTINTERVAL 515
#define TIFFTAG_JPEGLOSSLESSPREDICTORS 517
#define TIFFTAG_JPEGPOINTTRANSFORM  518
#define TIFFTAG_JPEGQTABLES          519
#define TIFFTAG_JPEGDCTABLES         520
#define TIFFTAG_JPEGACTABLES         521
#define TIFFTAG_YCBCRCOEFFICIENTS    529
#define TIFFTAG_YCBCRSUBSAMPLING     530
#define TIFFTAG_YCBCRPOSITIONING     531
#define TIFFTAG_REFERENCEBLACKWHITE  532
#define COMPRESSION_OJPEG            6
#define ORIENTATION_RIGHTTOP          6
#define PHOTOMETRIC_YCBCR            6
#define SAMPLEFORMAT_COMPLEXIEEEFP   6
#define TIFFTAG_DCSHUESHIFTVALUES     65535
#define TIFFTAG_FAXMODE               65536
#define TIFFTAG_JPEGQUALITY           65537
#define TIFFTAG_JPEGCOLORMODE         65538
#define TIFFTAG_JPEGTABLESMODE       65539
#define TIFFTAG_FAXFILLFUNC           65540
#define TIFFTAG_PIXARLOGDATAFORMAT   65549
#define TIFFTAG_DCSIMAGERTYPE         65550
#define TIFFTAG_DCSINTERPMODE         65551
#define TIFFTAG_DCSBALANCEARRAY       65552
#define TIFFTAG_DCSCORRECTMATRIX     65553
#define TIFFTAG_DCSGAMMA              65554
#define TIFFTAG_DCSTOESHOULDERPTS     65555
#define TIFFTAG_DCSLIBRATIONFD        65556
#define TIFFTAG_ZIPQUALITY            65557
#define TIFFTAG_PIXARLOGQUALITY       65558

```



```

#define TIFFTAG_DCSCLIPRECTANGLE      65559
#define TIFFTAG_SGILOGDATAFMT      65560
#define TIFFTAG_SGILOGENCODE      65561
#define TIFFTAG_LZMAPRESET      65562
#define TIFFTAG_PERSAMPLE      65563
#define COMPRESSION_JPEG      7
#define ORIENTATION_RIGHTBOT      7
#define TIFFTAG_XMLPACKET      700
#define COMPRESSION_ADOBE_DEFLATE      8
#define ORIENTATION_LEFTBOT      8
#define PHOTOMETRIC_CIELAB      8
#define PHOTOMETRIC_ICCLAB      9
#define FAXMODE_CLASSF FAXMODE_NORTC

typedef TIFF_INT8_T int8;
typedef TIFF_INT16_T int16;
typedef TIFF_INT32_T int32;
typedef TIFF_UINT64_T uint64;
typedef int uint16_vap;
typedef TIFF_UINT8_T uint8;
typedef TIFF_UINT16_T uint16;
typedef TIFF_UINT32_T uint32;

typedef enum {
    TIFF_NOTYPE,
    TIFF_BYTE,
    TIFF_ASCII,
    TIFF_SHORT,
    TIFF_LONG,
    TIFF_RATIONAL,
    TIFF_SBYTE,
    TIFF_UNDEFINED,
    TIFF_SSHORT,
    TIFF_SLONG,
    TIFF_SRATIONAL,
    TIFF_FLOAT,
    TIFF_DOUBLE,
    TIFF_IFD,
    TIFF_LONG8,
    TIFF_SLONG8,
    TIFF_IFD8
} TIFFDataType;

```

16.2.2 tiffconf.h

```

#define CMYK_SUPPORT
#define COLORIMETRY_SUPPORT
#define ICC_SUPPORT
#define IPTC_SUPPORT
#define PHOTOSHOP_SUPPORT
#define YCBCR_SUPPORT
#define HOST_BIGENDIAN 0
#define CCITT_SUPPORT 1
#define CHECK_JPEG_YCBCR_SUBSAMPLING 1
#define DEFAULT_EXTRASAMPLE_AS_ALPHA 1
#define HAVE_IEEEFP 1
#define JBIG_SUPPORT 1
#define JPEG_SUPPORT 1
#define LOGLUV_SUPPORT 1
#define LZW_SUPPORT 1
#define MDI_SUPPORT 1
#define NEXT_SUPPORT 1
#define OJPEG_SUPPORT 1
#define PACKBITS_SUPPORT 1
#define PIXARLOG_SUPPORT 1

```



```

#define SUBIFD_SUPPORT 1
#define THUNDER_SUPPORT 1
#define ZIP_SUPPORT 1
#define HOST_FILLORDER FILLORDER_LSB2MSB
#define TIFF_PTRDIFF_T ptrdiff_t
#define TIFF_INT8_T signed char
#define TIFF_INT32_T signed int
#define TIFF_SSIZE_T signed long
#define TIFF_INT64_T signed long long
#define TIFF_INT16_T signed short
#define STRIPCHOP_DEFAULT TIFF_STRIPCHOP
#define TIFF_UINT8_T unsigned char
#define TIFF_UINT32_T unsigned int
#define TIFF_UINT64_T unsigned long long
#define TIFF_UINT16_T unsigned short

```

16.2.3 tiffio.h

```

#define TIFFGetB(abgr) (((abgr) >> 16) & 0xff)
#define TIFFGetA(abgr) (((abgr) >> 24) & 0xff)
#define TIFFGetG(abgr) (((abgr) >> 8) & 0xff)
#define TIFFGetR(abgr) ((abgr) & 0xff)
#define D50_Y0 (100.0F)
#define D65_Y0 (100.0F)
#define D65_Z0 (108.8827F)
#define D50_Z0 (82.4680F)
#define D65_X0 (95.0470F)
#define D50_X0 (96.4250F)
#define TIFF_VARIABLE -1
#define TIFF_SPP -2
#define TIFF_VARIABLE2 -3
#define U_NEU 0.210526316
#define V_NEU 0.473684211
#define TIFFPRINT_NONE 0x0
#define TIFFPRINT_STRIPS 0x1
#define TIFFPRINT_JPEGOTABLES 0x100
#define TIFFPRINT_CURVES 0x2
#define TIFFPRINT_JPEGACTABLES 0x200
#define TIFFPRINT_JPEGDCTABLES 0x200
#define TIFFPRINT_COLOMAP 0x4
#define LOGLUV_PUBLIC 1
#define CIELABTORGB_TABLE_RANGE 1500
#define UVSCALE 410.
#define FIELD_CUSTOM 65
#define TIFF_ANY TIFF_NOTYPE

```

```

typedef struct tiff TIFF;
typedef TIFF_SSIZE_T tmsize_t;
typedef uint64 toff_t;
typedef uint32 ttag_t;
typedef uint16 tdir_t;
typedef uint16 tsample_t;
typedef uint32 tstrile_t;
typedef tstrile_t tstrip_t;
typedef tstrile_t ttile_t;
typedef tmsize_t tsize_t;
typedef void *tdata_t;
typedef void *thandle_t;
typedef unsigned char TIFFRGBValue;
typedef struct {
    float d_mat[3][3];
    float d_YCR;
    float d_YCG;
    float d_YCB;
    uint32 d_Vrwr;
}

```

```

        uint32 d_Vrwg;
        uint32 d_Vrwb;
        float d_Y0R;
        float d_Y0G;
        float d_Y0B;
        float d_gammaR;
        float d_gammaG;
        float d_gammaB;
    } TIFFDisplay;
typedef struct {
    TIFFRGBValue *clamptab;
    int *Cr_r_tab;
    int *Cb_b_tab;
    int32 *Cr_g_tab;
    int32 *Cb_g_tab;
    int32 *Y_tab;
} TIFFYCbCrToRGB;
typedef struct {
    int range;
    float rstep;
    float gstep;
    float bstep;
    float X0;
    float Y0;
    float Z0;
    TIFFDisplay display;
    float Yr2r[1501];
    float Yg2g[1501];
    float Yb2b[1501];
} TIFFCIELabToRGB;
typedef struct _TIFFRGBAImage {
    TIFF *tif;
    int stoponerr;
    int isContig;
    int alpha;
    uint32 width;
    uint32 height;
    uint16 bitspersample;
    uint16 samplesperpixel;
    uint16 orientation;
    uint16 req_orientation;
    uint16 photometric;
    uint16 *redcmap;
    uint16 *greencmap;
    uint16 *bluecmap;
    int (*get) (TIFFRGBAImage *, uint32 *, uint32, uint32);
    union {
        void (*any) (TIFFRGBAImage *);
        tileContigRoutine contig;
        tileSeparateRoutine separate;
    } put;
    TIFFRGBValue *Map;
    uint32 **BWmap;
    uint32 **PALmap;
    TIFFYCbCrToRGB *ychcr;
    TIFFCIELabToRGB *cielab;
    uint8 *UaToAa;
    uint8 *Bitdepth16To8;
    int row_offset;
    int col_offset;
} TIFFRGBAImage;
typedef void (*tileContigRoutine) (TIFFRGBAImage *, uint32 *,
uint32,
uint32, uint32, uint32, int32, int32,
unsigned char *);

```

```

typedef void (*tileSeparateRoutine) (TIFFRGBAImage *, uint32 *,
uint32,
uint32, uint32, uint32, int32,
int32,
unsigned char *, unsigned char *,
unsigned char *, unsigned char *);
typedef int (*TIFFInitMethod) (TIFF *, int);
typedef struct {
    char *name;
    uint16 scheme;
    TIFFInitMethod init;
} TIFFCodec;
typedef void (*TIFFErrorHandler) (const char *, const char *,
va_list);
typedef void (*TIFFErrorHandlerExt) (thandle_t, const char *, const
char *,
va_list);
typedef tmsize_t (*TIFFReadWriteProc) (thandle_t, void *, tmsize_t);
typedef toff_t (*TIFFSeekProc) (thandle_t, toff_t, int);
typedef int (*TIFFCloseProc) (thandle_t);
typedef toff_t (*TIFFSizeProc) (thandle_t);
typedef int (*TIFFMapFileProc) (thandle_t, void **, toff_t *);
typedef void (*TIFFUnmapFileProc) (thandle_t, void *, toff_t);
typedef void (*TIFFExtendProc) (TIFF *);
typedef struct _TIFFField TIFFField;
typedef struct _TIFFFieldArray TIFFFieldArray;
typedef int (*TIFFVSetMethod) (TIFF *, uint32, va_list);
typedef int (*TIFFVGetMethod) (TIFF *, uint32, va_list);
typedef void (*TIFFPrintMethod) (TIFF *, FILE *, long int);
typedef struct {
    TIFFVSetMethod vsetfield;
    TIFFVGetMethod vgetfield;
    TIFFPrintMethod printdir;
} TIFFTagMethods;
extern int LogL10fromY(double, int);
extern double LogL10toY(int);
extern int LogL16fromY(double, int);
extern double LogL16toY(int);
extern uint32 LogLuv24fromXYZ(float *, int);
extern void LogLuv24toXYZ(uint32, float *);
extern uint32 LogLuv32fromXYZ(float *, int);
extern void LogLuv32toXYZ(uint32, float *);
extern TIFFTagMethods *TIFFAccessTagMethods(TIFF *);
extern int TIFFCIELabToRGBInit(TIFFCIELabToRGB *, const TIFFDisplay
*,
float *);
extern void TIFFCIELabToXYZ(TIFFCIELabToRGB *, uint32, int32, int32,
float *, float *, float *);
extern int TIFFCheckTile(TIFF * tif, uint32 x, uint32 y, uint32 z,
uint16 s);
extern int TIFFCheckpointDirectory(TIFF *);
extern void TIFFCleanup(TIFF * tif);
extern TIFF *TIFFClientOpen(const char *, const char *, thandle_t,
TIFFReadWriteProc, TIFFReadWriteProc,
TIFFSeekProc, TIFFCloseProc, TIFFSizeProc,
TIFFMapFileProc, TIFFUnmapFileProc);
extern thandle_t TIFFClientdata(TIFF *);
extern void TIFFClose(TIFF * tif);
extern uint32 TIFFComputeStrip(TIFF *, uint32, uint16);
extern uint32 TIFFComputeTile(TIFF * tif, uint32 x, uint32 y, uint32
z,
uint16 s);
extern int TIFFCreateCustomDirectory(TIFF *, const TIFFFieldArray
*);
extern int TIFFCreateDirectory(TIFF *);
extern int TIFFCreateEXIFDirectory(TIFF *);

```

```

extern uint64 TIFFCurrentDirOffset(TIFF *);
extern uint16 TIFFCurrentDirectory(TIFF *);
extern uint32 TIFFCurrentRow(TIFF *);
extern uint32 TIFFCurrentStrip(TIFF *);
extern uint32 TIFFCurrentTile(TIFF * tif);
extern int TIFFDataWidth(TIFFDataType);
extern uint32 TIFFDefaultStripSize(TIFF * tif, uint32 request);
extern void TIFFDefaultTileSize(TIFF *, uint32 *, uint32 *);
extern void TIFFError(const char *, const char *, ...);
extern void TIFFErrorExt(thandle_t, const char *, const char *, ...);
extern TIFF *TIFFFdOpen(int, const char *, const char *);
extern const TIFFField *TIFFFieldWithName(TIFF *, const char *);
extern const TIFFField *TIFFFieldWithTag(TIFF *, uint32);
extern const char *TIFFFileName(TIFF *);
extern int TIFFFileNo(TIFF *);
extern const TIFFCodec *TIFFFindCODEC(uint16);
extern const TIFFField *TIFFFindField(TIFF *, uint32, TIFFDataType);
extern int TIFFFlush(TIFF * tif);
extern int TIFFFlushData(TIFF * tif);
extern void TIFFFreeDirectory(TIFF *);
extern const unsigned char *TIFFGetBitRevTable(int);
extern void *TIFFGetClientInfo(TIFF *, const char *);
extern TIFFCloseProc TIFFGetCloseProc(TIFF *);
extern TIFFCodec *TIFFGetConfiguredCODECs(void);
extern int TIFFGetField(TIFF * tif, uint32 tag, ...);
extern int TIFFGetFieldDefaulted(TIFF * tif, uint32 tag, ...);
extern TIFFMapFileProc TIFFGetMapFileProc(TIFF *);
extern int TIFFGetMode(TIFF *);
extern TIFFReadWriteProc TIFFGetReadProc(TIFF *);
extern TIFFSeekProc TIFFGetSeekProc(TIFF *);
extern TIFFSizeProc TIFFGetSizeProc(TIFF *);
extern int TIFFGetTagListCount(TIFF *);
extern uint32 TIFFGetTagListEntry(TIFF *, int tag_index);
extern TIFFUnmapFileProc TIFFGetUnmapFileProc(TIFF *);
extern const char *TIFFGetVersion(void);
extern TIFFReadWriteProc TIFFGetWriteProc(TIFF *);
extern int TIFFIsBigEndian(TIFF *);
extern int TIFFIsByteSwapped(TIFF *);
extern int TIFFIsCODECConfigured(uint16);
extern int TIFFIsMSB2LSB(TIFF *);
extern int TIFFIsTiled(TIFF *);
extern int TIFFIsUpSampled(TIFF *);
extern int TIFFLastDirectory(TIFF *);
extern uint16 TIFFNumberOfDirectories(TIFF *);
extern uint32 TIFFNumberOfStrips(TIFF *);
extern uint32 TIFFNumberOfTiles(TIFF *);
extern TIFF *TIFFOpen(const char *, const char *);
extern void TIFFPrintDirectory(TIFF *, FILE *, long int);
extern int TIFFRGBAImageBegin(TIFFRGBAImage *, TIFF *, int, char *);
extern void TIFFRGBAImageEnd(TIFFRGBAImage *);
extern int TIFFRGBAImageGet(TIFFRGBAImage *, uint32 *, uint32, uint32);
extern int TIFFRGBAImageOK(TIFF *, char *);
extern tmsize_t TIFFRasterScanlineSize(TIFF * tif);
extern uint64 TIFFRasterScanlineSize64(TIFF * tif);
extern tmsize_t TIFFRawStripSize(TIFF * tif, uint32 strip);
extern uint64 TIFFRawStripSize64(TIFF * tif, uint32 strip);
extern int TIFFReadBufferSetup(TIFF * tif, void *bp, tmsize_t size);
extern int TIFFReadCustomDirectory(TIFF * tif, toff_t diroff, const TIFFFieldArray * infoarray);
extern int TIFFReadDirectory(TIFF * tif);
extern int TIFFReadEXIFDirectory(TIFF * tif, toff_t diroff);
extern tmsize_t TIFFReadEncodedStrip(TIFF * tif, uint32 strip, void *buf, tmsize_t size);

```

```

extern tmsize_t TIFFReadEncodedTile(TIFF * tif, uint32 tile, void
*buf,
                                tmsize_t size);
extern int TIFFReadRGBAIImage(TIFF * tif, uint32, uint32, uint32 *, int);
extern int TIFFReadRGBAIImageOriented(TIFF * tif, uint32, uint32, uint32
*, int,
                                int);
extern int TIFFReadRGBAStrip(TIFF * tif, uint32, uint32 *);
extern int TIFFReadRGBATile(TIFF * tif, uint32, uint32, uint32 *);
extern tmsize_t TIFFReadRawStrip(TIFF * tif, uint32 strip, void
*buf,
                                tmsize_t size);
extern tmsize_t TIFFReadRawTile(TIFF * tif, uint32 tile, void *buf,
                                tmsize_t size);
extern int TIFFReadScanline(TIFF * tif, void *buf, uint32 row,
                                uint16 sample);
extern tmsize_t TIFFReadTile(TIFF * tif, void *buf, uint32 x, uint32
y,
                                uint32 z, uint16 s);
extern TIFFCodec *TIFFRegisterCODEC(uint16, const char *,
TIFFInitMethod);
extern void TIFFReverseBits(uint8 * cp, tmsize_t n);
extern int TIFFRewriteDirectory(TIFF *);
extern tmsize_t TIFFScanlineSize(TIFF * tif);
extern uint64 TIFFScanlineSize64(TIFF * tif);
extern void TIFFSetClientInfo(TIFF * tif, void *, const char *);
extern thandle_t TIFFSetClientdata(TIFF * tif, thandle_t);
extern int TIFFSetDirectory(TIFF * tif, uint16);
extern TIFFErrorHandler TIFFSetErrorHandler(TIFFErrorHandler);
extern TIFFErrorHandlerExt TIFFSetErrorHandlerExt(TIFFErrorHandlerExt);
extern int TIFFSetField(TIFF * tif, uint32, ...);
extern const char *TIFFSetFileName(TIFF * tif, const char *);
extern int TIFFSetFileno(TIFF * tif, int);
extern int TIFFSetMode(TIFF * tif, int);
extern int TIFFSetSubDirectory(TIFF * tif, uint64);
extern TIFFExtendProc TIFFSetTagExtender(TIFFExtendProc);
extern TIFFErrorHandler TIFFSetWarningHandler(TIFFErrorHandler);
extern TIFFErrorHandlerExt TIFFSetWarningHandlerExt(TIFFErrorHandlerExt);
extern void TIFFSetWriteOffset(TIFF * tif, toff_t off);
extern int TIFFSetupStrips(TIFF * tif);
extern tmsize_t TIFFStripSize(TIFF * tif);
extern uint64 TIFFStripSize64(TIFF * tif);
extern void TIFFSwabArrayOfDouble(double *dp, tmsize_t n);
extern void TIFFSwabArrayOfFloat(float *fp, tmsize_t n);
extern void TIFFSwabArrayOfLong(uint32 * lp, tmsize_t n);
extern void TIFFSwabArrayOfLong8(uint64 * lp, tmsize_t n);
extern void TIFFSwabArrayOfShort(uint16 * wp, tmsize_t n);
extern void TIFFSwabArrayOfTriples(uint8 * tp, tmsize_t n);
extern void TIFFSwabDouble(double *);
extern void TIFFSwabFloat(float *);
extern void TIFFSwabLong(uint32 *);
extern void TIFFSwabLong8(uint64 *);
extern void TIFFSwabShort(uint16 *);
extern tmsize_t TIFFTileRowSize(TIFF * tif);
extern uint64 TIFFTileRowSize64(TIFF * tif);
extern tmsize_t TIFFTileSize(TIFF * tif);
extern uint64 TIFFTileSize64(TIFF * tif);
extern void TIFFUnRegisterCODEC(TIFFCodec *);
extern int TIFFUnlinkDirectory(TIFF * tif, uint16);
extern int TIFFUnsetField(TIFF * tif, uint32);
extern int TIFFVGetField(TIFF * tif, uint32 tag, va_list ap);
extern int TIFFVGetFieldDefaulted(TIFF * tif, uint32 tag, va_list
ap);
extern int TIFFVSetField(TIFF * tif, uint32, va_list);

```

```

extern tmsize_t TIFFVStripSize(TIFF * tif, uint32 nrows);
extern uint64 TIFFVStripSize64(TIFF * tif, uint32 nrows);
extern tmsize_t TIFFVTileSize(TIFF * tif, uint32 nrows);
extern uint64 TIFFVTileSize64(TIFF * tif, uint32 nrows);
extern void TIFFWarning(const char *, const char *, ...);
extern void TIFFWarningExt(thandle_t, const char *, const char
*, ...);
extern int TIFFWriteBufferSetup(TIFF * tif, void *bp, tmsize_t
size);
extern int TIFFWriteCheck(TIFF *, int, const char *);
extern int TIFFWriteCustomDirectory(TIFF *, uint64 *);
extern int TIFFWriteDirectory(TIFF *);
extern tmsize_t TIFFWriteEncodedStrip(TIFF * tif, uint32 strip,
void *data,
                                tmsize_t cc);
extern tmsize_t TIFFWriteEncodedTile(TIFF * tif, uint32 tile, void
*data,
                                tmsize_t cc);
extern tmsize_t TIFFWriteRawStrip(TIFF * tif, uint32 strip, void
*data,
                                tmsize_t cc);
extern tmsize_t TIFFWriteRawTile(TIFF * tif, uint32 tile, void
*data,
                                tmsize_t cc);
extern int TIFFWriteScanline(TIFF * tif, void *buf, uint32 row,
uint16 sample);
extern tmsize_t TIFFWriteTile(TIFF * tif, void *buf, uint32 x,
uint32 y,
                                uint32 z, uint16 s);
extern void TIFFXYZToRGB(TIFFCIELabToRGB *, float, float, float,
uint32 *,
                                uint32 *, uint32 *);
extern int TIFFYCbCrToRGBInit(TIFFYCbCrToRGB *, float *, float *);
extern void TIFFYCbCrToRGB(TIFFYCbCrToRGB *, uint32, int32, int32,
uint32 *, uint32 *, uint32 *);
extern void XYZtoRGB24(float *, uint8 *);
extern void _TIFFfree(void *p);
extern void *_TIFFmalloc(tmsize_t s);
extern int _TIFFmemcmp(const void *p1, const void *p2, tmsize_t c);
extern void _TIFFmemcpy(void *d, const void *s, tmsize_t c);
extern void _TIFFmemset(void *p, int v, tmsize_t c);
extern void *_TIFFrealloc(void *p, tmsize_t s);
extern int uv_decode(double *, double *, int);
extern int uv_encode(double, double, int);

```

16.2.4 tiffvers.h

```
#define TIFFLIB_VERSION 20120615
```

XII GTK+ Stack Libraries

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

17 Libraries

17.1 Introduction

A conforming implementation shall support the following libraries from the GTK+ stack which provide interfaces for creating rich graphical user interface applications.

GLib

A general-purpose utility library, not specific to graphical user interfaces. Glib is implemented as several libraries, providing a type system, a signal system, data conversion routines, string utilities, a filesystem abstraction and file utility routines, threading support and a main loop abstraction.

libglib-2.0

libgmodule-2.0

libgobject-2.0

libgthread-2.0

libgio-2.0

ATK

ATK is the Accessibility Toolkit. It provides a set of generic interfaces allowing accessibility technologies to interact with a graphical user interface. For example, a screen reader uses ATK to discover the text in an interface and read it to blind users. GTK+ widgets have built-in support for accessibility using the ATK framework.

libatk-1.0

Pango

Pango is a library for laying out and rendering text. It centers around the PangoLayout object, representing a paragraph of text. Pango provides the engine for GtkTextView, GtkLabel, GtkEntry, and other widgets that display text.

libpango-1.0

libpangocairo-1.0

libpangoft2-1.0

libpangoxft-1.0

GdkPixbuf

This is a small library which allows you to create GdkPixbuf ("pixel buffer") objects from image data or image files. Use a GdkPixbuf in combination with GtkImage to display images.

libgdk_pixbuf-2.0

libgdk_pixbuf_xlib-2.0

GDK

GDK is the abstraction layer that allows GTK+ to support multiple windowing systems. GDK provides drawing and window system facilities on X11, Windows, and the Linux framebuffer device.

libgdk-x11-2.0

GTK+

The GTK+ library contains widgets, that is, GUI components such as GtkButton or GtkTextView.

libgtk-x11-2.0

There are three main parts to the definition of each of these libraries.

The "Interfaces" section defines the required library name and version, and the required public symbols (interfaces and global data), as well as symbol versions, if any.

The "Interface Definitions" section provides complete or partial definitions of certain interfaces where either this specification is the source specification, or where there are variations from the source specification. If an interface definition requires one or more header files, one of those headers shall include the function prototype for the interface.

For source definitions of interfaces which include a reference to a header file, the contents of such header files form a part of the specification. The "Data Definitions" section provides the binary-level details for the header files from the source specifications, such as values for macros and enumerated types, as well as structure layouts, sizes and padding, etc. These data definitions, although presented in the form of header files for convenience, should not be taken as representing complete header files, as they are a supplement to the source specifications. Application developers should follow the guidelines of the source specifications when determining which header files need to be included to completely resolve all references.

Note: While the Data Definitions supplement the source specifications, this specification itself does not require conforming implementations to supply any header files.

17.2 Interfaces for libglib-2.0

Table 17-1 defines the library name and shared object name for the libglib-2.0 library

Table 17-1 libglib-2.0 Definition

Library:	libglib-2.0
SONAME:	libglib-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

- [Glib 2.32] Glib 2.32 Reference Manual
- [Gobject 2.32] Gobject 2.32 Reference Manual
- [LSB] This Specification

17.2.1 GTK General purpose utility library

17.2.1.1 Interfaces for GTK General purpose utility library

An LSB conforming implementation shall provide the generic functions for GTK General purpose utility library specified in Table 17-2, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-2 libglib-2.0 - GTK General purpose utility library Function Interfaces

g_array_get_element_size [Glib 2.32]	g_array_ref [Glib 2.32]
g_array_set_clear_func [Glib 2.32]	g_array_unref [Glib 2.32]
g_assert_warning [Glib 2.32]	g_assertion_message [LSB]
g_assertion_message_cmpnum [LSB]	g_assertion_message_cmpstr [LSB]
g_assertion_message_error [LSB]	g_assertion_message_expr [LSB]
g_async_queue_new_full [Glib 2.32]	g_async_queue_ref_unlocked [Glib 2.32]
g_async_queue_timeout_pop [Glib 2.32]	g_async_queue_timeout_pop_unlocked [Glib 2.32]
g_async_queue_unref_and_unlock [Glib 2.32]	g_atomic_int_and [Glib 2.32]
g_atomic_int_dec_and_test [Glib 2.32]	g_atomic_int_get [Glib 2.32]
g_atomic_int_inc [Glib 2.32]	g_atomic_int_or [Glib 2.32]
g_atomic_int_set [Glib 2.32]	g_atomic_int_xor [Glib 2.32]
g_atomic_pointer_add [Glib 2.32]	g_atomic_pointer_and [Glib 2.32]
g_atomic_pointer_get [Glib 2.32]	g_atomic_pointer_or [Glib 2.32]
g_atomic_pointer_set [Glib 2.32]	g_atomic_pointer_xor [Glib 2.32]
g_basename [Glib 2.32]	g_bit_lock [Glib 2.32]
g_bit_trylock [Glib 2.32]	g_bit_unlock [Glib 2.32]
g_byte_array_free_to_bytes [Glib 2.32]	g_byte_array_new_take [Glib 2.32]
g_byte_array_ref [Glib 2.32]	g_byte_array_unref [Glib 2.32]
g_bytes_compare [Glib 2.32]	g_bytes_equal [Glib 2.32]
g_bytes_get_data [Glib 2.32]	g_bytes_get_size [Glib 2.32]
g_bytes_hash [Glib 2.32]	g_bytes_new [Glib 2.32]
g_bytes_new_from_bytes [Glib 2.32]	g_bytes_new_static [Glib 2.32]
g_bytes_new_take [Glib 2.32]	g_bytes_new_with_free_func [Glib 2.32]
g_bytes_ref [Glib 2.32]	g_bytes_unref [Glib 2.32]
g_bytes_unref_to_array [Glib 2.32]	g_bytes_unref_to_data [Glib 2.32]
g_checksum_copy [Glib 2.32]	g_checksum_free [Glib 2.32]
g_checksum_get_digest [Glib 2.32]	g_checksum_get_string [Glib 2.32]
g_checksum_new [Glib 2.32]	g_checksum_reset [Glib 2.32]
g_checksum_type_get_length [Glib 2.32]	g_checksum_update [Glib 2.32]

g_compute_checksum_for_data [Glib 2.32]	g_compute_checksum_for_string [Glib 2.32]
g_compute_hmac_for_data [Glib 2.32]	g_compute_hmac_for_string [Glib 2.32]
g_cond_broadcast [Glib 2.32]	g_cond_clear [Glib 2.32]
g_cond_free [Glib 2.32]	g_cond_init [Glib 2.32]
g_cond_new [Glib 2.32]	g_cond_signal [Glib 2.32]
g_cond_timed_wait [Glib 2.32]	g_cond_wait [Glib 2.32]
g_cond_wait_until [Glib 2.32]	g_datalist_get_data [Glib 2.32]
g_date_time_add [Glib 2.32]	g_date_time_add_days [Glib 2.32]
g_date_time_add_full [Glib 2.32]	g_date_time_add_hours [Glib 2.32]
g_date_time_add_minutes [Glib 2.32]	g_date_time_add_months [Glib 2.32]
g_date_time_add_seconds [Glib 2.32]	g_date_time_add_weeks [Glib 2.32]
g_date_time_add_years [Glib 2.32]	g_date_time_compare [Glib 2.32]
g_date_time_difference [Glib 2.32]	g_date_time_equal [Glib 2.32]
g_date_time_format [Glib 2.32]	g_date_time_get_day_of_month [Glib 2.32]
g_date_time_get_day_of_week [Glib 2.32]	g_date_time_get_day_of_year [Glib 2.32]
g_date_time_get_hour [Glib 2.32]	g_date_time_get_microsecond [Glib 2.32]
g_date_time_get_minute [Glib 2.32]	g_date_time_get_month [Glib 2.32]
g_date_time_get_second [Glib 2.32]	g_date_time_get_seconds [Glib 2.32]
g_date_time_get_timezone_abbreviation [Glib 2.32]	g_date_time_get_utc_offset [Glib 2.32]
g_date_time_get_week_numbering_year [Glib 2.32]	g_date_time_get_week_of_year [Glib 2.32]
g_date_time_get_year [Glib 2.32]	g_date_time_get_ymd [Glib 2.32]
g_date_time_hash [Glib 2.32]	g_date_time_is_daylight_savings [Glib 2.32]
g_date_time_new [Glib 2.32]	g_date_time_new_from_timeval_local [Glib 2.32]
g_date_time_new_from_timeval_utc [Glib 2.32]	g_date_time_new_from_unix_local [Glib 2.32]
g_date_time_new_from_unix_utc [Glib 2.32]	g_date_time_new_local [Glib 2.32]
g_date_time_new_now [Glib 2.32]	g_date_time_new_now_local [Glib 2.32]

g_date_time_new_now_utc [Glib 2.32]	g_date_time_new_utc [Glib 2.32]
g_date_time_ref [Glib 2.32]	g_date_time_to_local [Glib 2.32]
g_date_time_to_timeval [Glib 2.32]	g_date_time_to_timezone [Glib 2.32]
g_date_time_to_unix [Glib 2.32]	g_date_time_to_utc [Glib 2.32]
g_date_time_unref [Glib 2.32]	g_dcgettext [Glib 2.32]
g_dgettext [Glib 2.32]	g_dir_make_tmp [Glib 2.32]
g_dngettext [Glib 2.32]	g_double_equal [Glib 2.32]
g_double_hash [Glib 2.32]	g_dpgettext [Glib 2.32]
g_dpgettext2 [Glib 2.32]	g_envron_getenv [Glib 2.32]
g_envron_setenv [Glib 2.32]	g_envron_unsetenv [Glib 2.32]
g_error_new_valist [Glib 2.32]	g_format_size [Glib 2.32]
g_format_size_for_display [Glib 2.32]	g_format_size_full [Glib 2.32]
g_get_codeset [Glib 2.32]	g_get_envron [Glib 2.32]
g_get_locale_variants [Glib 2.32]	g_get_monotonic_time [Glib 2.32]
g_get_real_time [Glib 2.32]	g_get_user_runtime_dir [Glib 2.32]
g_get_user_special_dir [Glib 2.32]	g_hash_table_add [Glib 2.32]
g_hash_table_contains [Glib 2.32]	g_hash_table_get_keys [Glib 2.32]
g_hash_table_get_values [Glib 2.32]	g_hash_table_iter_get_hash_table [Glib 2.32]
g_hash_table_iter_init [Glib 2.32]	g_hash_table_iter_next [Glib 2.32]
g_hash_table_iter_remove [Glib 2.32]	g_hash_table_iter_replace [Glib 2.32]
g_hash_table_iter_steal [Glib 2.32]	g_hmac_copy [Glib 2.32]
g_hmac_get_digest [Glib 2.32]	g_hmac_get_string [Glib 2.32]
g_hmac_new [Glib 2.32]	g_hmac_ref [Glib 2.32]
g_hmac_unref [Glib 2.32]	g_hmac_update [Glib 2.32]
g_hostname_is_ascii_encoded [Glib 2.32]	g_hostname_is_ip_address [Glib 2.32]
g_hostname_is_non_ascii [Glib 2.32]	g_hostname_to_ascii [Glib 2.32]
g_hostname_to_unicode [Glib 2.32]	g_int64_equal [Glib 2.32]
g_int64_hash [Glib 2.32]	g_io_channel_close [Glib 2.32]
g_io_channel_read [Glib 2.32]	g_io_channel_seek [Glib 2.32]
g_io_channel_write [Glib 2.32]	g_key_file_get_int64 [Glib 2.32]
g_key_file_get_uint64 [Glib 2.32]	g_key_file_load_from_dirs [Glib 2.32]
g_key_file_ref [Glib 2.32]	g_key_file_set_int64 [Glib 2.32]

g_key_file_set_uint64 [Glib 2.32]	g_key_file_unref [Glib 2.32]
g_list_free_full [Glib 2.32]	g_main_context_get_thread_default [Glib 2.32]
g_main_context_invoke [Glib 2.32]	g_main_context_invoke_full [Glib 2.32]
g_main_context_pop_thread_default [Glib 2.32]	g_main_context_push_thread_default [Glib 2.32]
g_main_context_ref_thread_default [Glib 2.32]	g_malloc0_n [Glib 2.32]
g_malloc_n [Glib 2.32]	g_mapped_file_new_from_fd [Glib 2.32]
g_mapped_file_ref [Glib 2.32]	g_mapped_file_unref [Glib 2.32]
g_markup_collect_attributes [Glib 2.32]	g_markup_parse_context_get_element_stack [Glib 2.32]
g_markup_parse_context_get_userdata [Glib 2.32]	g_markup_parse_context_pop [Glib 2.32]
g_markup_parse_context_push [Glib 2.32]	g_match_info_expand_references [Glib 2.32]
g_match_info_fetch [Glib 2.32]	g_match_info_fetch_all [Glib 2.32]
g_match_info_fetch_named [Glib 2.32]	g_match_info_fetch_named_pos [Glib 2.32]
g_match_info_fetch_pos [Glib 2.32]	g_match_info_free [Glib 2.32]
g_match_info_get_match_count [Glib 2.32]	g_match_info_get_regex [Glib 2.32]
g_match_info_get_string [Glib 2.32]	g_match_info_is_partial_match [Glib 2.32]
g_match_info_matches [Glib 2.32]	g_match_info_next [Glib 2.32]
g_match_info_ref [Glib 2.32]	g_match_info_unref [Glib 2.32]
g_mkdtemp [Glib 2.32]	g_mkdtemp_full [Glib 2.32]
g_mkstemp_full [Glib 2.32]	g_mutex_clear [Glib 2.32]
g_mutex_free [Glib 2.32]	g_mutex_init [Glib 2.32]
g_mutex_lock [Glib 2.32]	g_mutex_new [Glib 2.32]
g_mutex_trylock [Glib 2.32]	g_mutex_unlock [Glib 2.32]
g_once_impl [Glib 2.32]	g_once_init_enter [Glib 2.32]
g_once_init_enter_impl [LSB]	g_once_init_leave [Glib 2.32]
g_option_context_get_help [Glib 2.32]	g_pointer_bit_lock [Glib 2.32]
g_pointer_bit_trylock [Glib 2.32]	g_pointer_bit_unlock [Glib 2.32]
g_poll [Glib 2.32]	g_prefix_error [Glib 2.32]

g_private_get [Glib 2.32]	g_private_new [Glib 2.32]
g_private_replace [Glib 2.32]	g_private_set [Glib 2.32]
g_propagate_prefixed_error [Glib 2.32]	g_ptr_array_new_full [Glib 2.32]
g_ptr_array_new_with_free_func [Glib 2.32]	g_ptr_array_ref [Glib 2.32]
g_ptr_array_set_free_func [Glib 2.32]	g_ptr_array_unref [Glib 2.32]
g_queue_clear [Glib 2.32]	g_queue_free_full [Glib 2.32]
g_queue_init [Glib 2.32]	g_realloc_n [Glib 2.32]
g_rec_mutex_clear [Glib 2.32]	g_rec_mutex_init [Glib 2.32]
g_rec_mutex_lock [Glib 2.32]	g_rec_mutex_trylock [Glib 2.32]
g_rec_mutex_unlock [Glib 2.32]	g_regex_check_replacement [Glib 2.32]
g_regex_error_quark [LSB]	g_regex_escape_nul [Glib 2.32]
g_regex_escape_string [Glib 2.32]	g_regex_get_capture_count [Glib 2.32]
g_regex_get_compile_flags [Glib 2.32]	g_regex_get_match_flags [Glib 2.32]
g_regex_get_max_backref [Glib 2.32]	g_regex_get_pattern [Glib 2.32]
g_regex_get_string_number [Glib 2.32]	g_regex_match [Glib 2.32]
g_regex_match_all [Glib 2.32]	g_regex_match_all_full [Glib 2.32]
g_regex_match_full [Glib 2.32]	g_regex_match_simple [Glib 2.32]
g_regex_new [Glib 2.32]	g_regex_ref [Glib 2.32]
g_regex_replace [Glib 2.32]	g_regex_replace_eval [Glib 2.32]
g_regex_replace_literal [Glib 2.32]	g_regex_split [Glib 2.32]
g_regex_split_full [Glib 2.32]	g_regex_split_simple [Glib 2.32]
g_regex_unref [Glib 2.32]	g_reload_user_special_dirs_cache [Glib 2.32]
g_return_if_fail_warning [Glib 2.32]	g_rw_lock_clear [Glib 2.32]
g_rw_lock_init [Glib 2.32]	g_rw_lock_reader_lock [Glib 2.32]
g_rw_lock_reader_trylock [Glib 2.32]	g_rw_lock_reader_unlock [Glib 2.32]
g_rw_lock_writer_lock [Glib 2.32]	g_rw_lock_writer_trylock [Glib 2.32]
g_rw_lock_writer_unlock [Glib 2.32]	g_sequence_append [Glib 2.32]
g_sequence_foreach [Glib 2.32]	g_sequence_foreach_range [Glib 2.32]
g_sequence_free [Glib 2.32]	g_sequence_get [Glib 2.32]

<code>g_sequence_get_begin_iter</code> [Glib 2.32]	<code>g_sequence_get_end_iter</code> [Glib 2.32]
<code>g_sequence_get_iter_at_pos</code> [Glib 2.32]	<code>g_sequence_get_length</code> [Glib 2.32]
<code>g_sequence_insert_before</code> [Glib 2.32]	<code>g_sequence_insert_sorted</code> [Glib 2.32]
<code>g_sequence_insert_sorted_iter</code> [Glib 2.32]	<code>g_sequence_iter_compare</code> [Glib 2.32]
<code>g_sequence_iter_get_position</code> [Glib 2.32]	<code>g_sequence_iter_get_sequence</code> [Glib 2.32]
<code>g_sequence_iter_is_begin</code> [Glib 2.32]	<code>g_sequence_iter_is_end</code> [Glib 2.32]
<code>g_sequence_iter_move</code> [Glib 2.32]	<code>g_sequence_iter_next</code> [Glib 2.32]
<code>g_sequence_iter_prev</code> [Glib 2.32]	<code>g_sequence_lookup</code> [Glib 2.32]
<code>g_sequence_lookup_iter</code> [Glib 2.32]	<code>g_sequence_move</code> [Glib 2.32]
<code>g_sequence_move_range</code> [Glib 2.32]	<code>g_sequence_new</code> [Glib 2.32]
<code>g_sequence_prepend</code> [Glib 2.32]	<code>g_sequence_range_get_midpoint</code> [Glib 2.32]
<code>g_sequence_remove</code> [Glib 2.32]	<code>g_sequence_remove_range</code> [Glib 2.32]
<code>g_sequence_search</code> [Glib 2.32]	<code>g_sequence_search_iter</code> [Glib 2.32]
<code>g_sequence_set</code> [Glib 2.32]	<code>g_sequence_sort</code> [Glib 2.32]
<code>g_sequence_sort_changed</code> [Glib 2.32]	<code>g_sequence_sort_changed_iter</code> [Glib 2.32]
<code>g_sequence_sort_iter</code> [Glib 2.32]	<code>g_sequence_swap</code> [Glib 2.32]
<code>g_set_error_literal</code> [Glib 2.32]	<code>g_slist_free_full</code> [Glib 2.32]
<code>g_source_add_child_source</code> [Glib 2.32]	<code>g_source_get_name</code> [Glib 2.32]
<code>g_source_get_time</code> [Glib 2.32]	<code>g_source_remove_child_source</code> [Glib 2.32]
<code>g_source_set_name</code> [Glib 2.32]	<code>g_source_set_name_by_id</code> [Glib 2.32]
<code>g_static_mutex_get_mutex_impl</code> [Glib 2.32]	<code>g_strcasecmp</code> [Glib 2.32]
<code>g_strcmp0</code> [Glib 2.32]	<code>g_strdown</code> [Glib 2.32]
<code>g_string_append_uri_escaped</code> [Glib 2.32]	<code>g_string_append_vprintf</code> [Glib 2.32]
<code>g_string_chunk_clear</code> [Glib 2.32]	<code>g_string_down</code> [Glib 2.32]
<code>g_string_overwrite</code> [Glib 2.32]	<code>g_string_overwrite_len</code> [Glib 2.32]
<code>g_string_up</code> [Glib 2.32]	<code>g_string_vprintf</code> [Glib 2.32]
<code>g_strncasecmp</code> [Glib 2.32]	<code>g_strup</code> [Glib 2.32]

<code>g_test_add_data_func</code> [Glib 2.32]	<code>g_test_add_func</code> [Glib 2.32]
<code>g_test_add_vtable</code> [LSB]	<code>g_test_bug</code> [Glib 2.32]
<code>g_test_bug_base</code> [Glib 2.32]	<code>g_test_create_case</code> [Glib 2.32]
<code>g_test_create_suite</code> [Glib 2.32]	<code>g_test_fail</code> [Glib 2.32]
<code>g_test_get_root</code> [Glib 2.32]	<code>g_test_init</code> [Glib 2.32]
<code>g_test_log_buffer_free</code> [LSB]	<code>g_test_log_buffer_new</code> [LSB]
<code>g_test_log_buffer_pop</code> [LSB]	<code>g_test_log_buffer_push</code> [LSB]
<code>g_test_log_msg_free</code> [LSB]	<code>g_test_log_set_fatal_handler</code> [Glib 2.32]
<code>g_test_log_type_name</code> [LSB]	<code>g_test_maximized_result</code> [Glib 2.32]
<code>g_test_message</code> [Glib 2.32]	<code>g_test_minimized_result</code> [Glib 2.32]
<code>g_test_queue_destroy</code> [Glib 2.32]	<code>g_test_queue_free</code> [Glib 2.32]
<code>g_test_rand_double</code> [Glib 2.32]	<code>g_test_rand_double_range</code> [Glib 2.32]
<code>g_test_rand_int</code> [Glib 2.32]	<code>g_test_rand_int_range</code> [Glib 2.32]
<code>g_test_run</code> [Glib 2.32]	<code>g_test_run_suite</code> [Glib 2.32]
<code>g_test_suite_add</code> [Glib 2.32]	<code>g_test_suite_add_suite</code> [Glib 2.32]
<code>g_test_timer_elapsed</code> [Glib 2.32]	<code>g_test_timer_last</code> [Glib 2.32]
<code>g_test_timer_start</code> [Glib 2.32]	<code>g_test_trap_assertions</code> [LSB]
<code>g_test_trap_fork</code> [Glib 2.32]	<code>g_test_trap_has_passed</code> [Glib 2.32]
<code>g_test_trap_reached_timeout</code> [Glib 2.32]	<code>g_thread_create</code> [Glib 2.32]
<code>g_thread_get_initialized</code> [Glib 2.32]	<code>g_thread_gettime</code> [LSB]
<code>g_thread_new</code> [Glib 2.32]	<code>g_thread_ref</code> [Glib 2.32]
<code>g_thread_try_new</code> [Glib 2.32]	<code>g_thread_unref</code> [Glib 2.32]
<code>g_thread_yield</code> [Glib 2.32]	<code>g_time_zone_adjust_time</code> [Glib 2.32]
<code>g_time_zone_find_interval</code> [Glib 2.32]	<code>g_time_zone_get_abbreviation</code> [Glib 2.32]
<code>g_time_zone_get_offset</code> [Glib 2.32]	<code>g_time_zone_is_dst</code> [Glib 2.32]
<code>g_time_zone_new</code> [Glib 2.32]	<code>g_time_zone_new_local</code> [Glib 2.32]
<code>g_time_zone_new_utc</code> [Glib 2.32]	<code>g_time_zone_ref</code> [Glib 2.32]
<code>g_time_zone_unref</code> [Glib 2.32]	<code>g_timeout_add_seconds</code> [Glib 2.32]
<code>g_timeout_add_seconds_full</code> [Glib 2.32]	<code>g_timeout_source_new_seconds</code> [Glib 2.32]
<code>g_tree_ref</code> [Glib 2.32]	<code>g_tree_traverse</code> [Glib 2.32]
<code>g_tree_unref</code> [Glib 2.32]	<code>g_try_malloc0</code> [Glib 2.32]
<code>g_try_malloc0_n</code> [Glib 2.32]	<code>g_try_malloc_n</code> [Glib 2.32]

g_try_realloc_n [Glib 2.32]	g_unichar_combining_class [Glib 2.32]
g_unichar_compose [Glib 2.32]	g_unichar_decompose [Glib 2.32]
g_unichar_fully_decompose [Glib 2.32]	g_unichar_get_script [Glib 2.32]
g_unichar_ismark [Glib 2.32]	g_unichar_iszerowidth [Glib 2.32]
g_unicode_script_from_iso15924 [Glib 2.32]	g_unicode_script_to_iso15924 [Glib 2.32]
g_uri_escape_string [Glib 2.32]	g_uri_parse_scheme [Glib 2.32]
g_uri_unescape_segment [Glib 2.32]	g_uri_unescape_string [Glib 2.32]
g_utf8_substring [Glib 2.32]	g_variant_builder_add [Glib 2.32]
g_variant_builder_add_parsed [Glib 2.32]	g_variant_builder_add_value [Glib 2.32]
g_variant_builder_clear [Glib 2.32]	g_variant_builder_close [Glib 2.32]
g_variant_builder_end [Glib 2.32]	g_variant_builder_init [Glib 2.32]
g_variant_builder_new [Glib 2.32]	g_variant_builder_open [Glib 2.32]
g_variant_builder_ref [Glib 2.32]	g_variant_builder_unref [Glib 2.32]
g_variant_byteswap [Glib 2.32]	g_variant_classify [Glib 2.32]
g_variant_compare [Glib 2.32]	g_variant_dup_bytestring [Glib 2.32]
g_variant_dup_bytestring_array [Glib 2.32]	g_variant_dup_objv [Glib 2.32]
g_variant_dup_string [Glib 2.32]	g_variant_dup_strv [Glib 2.32]
g_variant_equal [Glib 2.32]	g_variant_get [Glib 2.32]
g_variant_get_boolean [Glib 2.32]	g_variant_get_byte [Glib 2.32]
g_variant_get_bytestring [Glib 2.32]	g_variant_get_bytestring_array [Glib 2.32]
g_variant_get_child [Glib 2.32]	g_variant_get_child_value [Glib 2.32]
g_variant_get_data [Glib 2.32]	g_variant_get_double [Glib 2.32]
g_variant_get_fixed_array [Glib 2.32]	g_variant_get_handle [Glib 2.32]
g_variant_get_int16 [Glib 2.32]	g_variant_get_int32 [Glib 2.32]
g_variant_get_int64 [Glib 2.32]	g_variant_get_maybe [Glib 2.32]
g_variant_get_normal_form [Glib 2.32]	g_variant_get_objv [Glib 2.32]
g_variant_get_size [Glib 2.32]	g_variant_get_string [Glib 2.32]
g_variant_get_strv [Glib 2.32]	g_variant_get_type [Gobject 2.32]
g_variant_get_type_string [Glib 2.32]	g_variant_get_uint16 [Glib 2.32]
g_variant_get_uint32 [Glib 2.32]	g_variant_get_uint64 [Glib 2.32]

g_variant_get_va [Glib 2.32]	g_variant_get_variant [Glib 2.32]
g_variant_hash [Glib 2.32]	g_variant_is_container [Glib 2.32]
g_variant_is_floating [Glib 2.32]	g_variant_is_normal_form [Glib 2.32]
g_variant_is_object_path [Glib 2.32]	g_variant_is_of_type [Glib 2.32]
g_variant_is_signature [Glib 2.32]	g_variant_iter_copy [Glib 2.32]
g_variant_iter_free [Glib 2.32]	g_variant_iter_init [Glib 2.32]
g_variant_iter_loop [Glib 2.32]	g_variant_iter_n_children [Glib 2.32]
g_variant_iter_new [Glib 2.32]	g_variant_iter_next [Glib 2.32]
g_variant_iter_next_value [Glib 2.32]	g_variant_lookup [Glib 2.32]
g_variant_lookup_value [Glib 2.32]	g_variant_n_children [Glib 2.32]
g_variant_new [Glib 2.32]	g_variant_new_array [Glib 2.32]
g_variant_new_boolean [Glib 2.32]	g_variant_new_byte [Glib 2.32]
g_variant_new_bytestring [Glib 2.32]	g_variant_new_bytestring_array [Glib 2.32]
g_variant_new_dict_entry [Glib 2.32]	g_variant_new_double [Glib 2.32]
g_variant_new_fixed_array [Glib 2.32]	g_variant_new_from_data [Glib 2.32]
g_variant_new_handle [Glib 2.32]	g_variant_new_int16 [Glib 2.32]
g_variant_new_int32 [Glib 2.32]	g_variant_new_int64 [Glib 2.32]
g_variant_new_maybe [Glib 2.32]	g_variant_new_object_path [Glib 2.32]
g_variant_new_objv [Glib 2.32]	g_variant_new_parsed [Glib 2.32]
g_variant_new_parsed_va [Glib 2.32]	g_variant_new_signature [Glib 2.32]
g_variant_new_string [Glib 2.32]	g_variant_new_strv [Glib 2.32]
g_variant_new_tuple [Glib 2.32]	g_variant_new_uint16 [Glib 2.32]
g_variant_new_uint32 [Glib 2.32]	g_variant_new_uint64 [Glib 2.32]
g_variant_new_va [Glib 2.32]	g_variant_new_variant [Glib 2.32]
g_variant_parse [Glib 2.32]	g_variant_parser_get_error_quark [LSB]
g_variant_print [Glib 2.32]	g_variant_print_string [Glib 2.32]
g_variant_ref [Glib 2.32]	g_variant_ref_sink [Glib 2.32]
g_variant_store [Glib 2.32]	g_variant_take_ref [Glib 2.32]
g_variant_type_checked_ [LSB]	g_variant_type_copy [Glib 2.32]
g_variant_type_dup_string [Glib 2.32]	g_variant_type_element [Glib 2.32]
g_variant_type_equal [Glib 2.32]	g_variant_type_first [Glib 2.32]

<code>g_variant_type_free</code> [Glib 2.32]	<code>g_variant_type_get_string_length</code> [Glib 2.32]
<code>g_variant_type_hash</code> [Glib 2.32]	<code>g_variant_type_is_array</code> [Glib 2.32]
<code>g_variant_type_is_basic</code> [Glib 2.32]	<code>g_variant_type_is_container</code> [Glib 2.32]
<code>g_variant_type_is_definite</code> [Glib 2.32]	<code>g_variant_type_is_dict_entry</code> [Glib 2.32]
<code>g_variant_type_is_maybe</code> [Glib 2.32]	<code>g_variant_type_is_subtype_of</code> [Glib 2.32]
<code>g_variant_type_is_tuple</code> [Glib 2.32]	<code>g_variant_type_is_variant</code> [Glib 2.32]
<code>g_variant_type_key</code> [Glib 2.32]	<code>g_variant_type_n_items</code> [Glib 2.32]
<code>g_variant_type_new</code> [Glib 2.32]	<code>g_variant_type_new_array</code> [Glib 2.32]
<code>g_variant_type_new_dict_entry</code> [Glib 2.32]	<code>g_variant_type_new_maybe</code> [Glib 2.32]
<code>g_variant_type_new_tuple</code> [Glib 2.32]	<code>g_variant_type_next</code> [Glib 2.32]
<code>g_variant_type_peek_string</code> [Glib 2.32]	<code>g_variant_type_string_is_valid</code> [Glib 2.32]
<code>g_variant_type_string_scan</code> [Glib 2.32]	<code>g_variant_type_value</code> [Glib 2.32]
<code>g_variant_unref</code> [Glib 2.32]	<code>g_warn_message</code> [LSB]

An LSB conforming implementation shall provide the generic deprecated functions for GTK General purpose utility library specified in Table 17-3, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-3 libglib-2.0 - GTK General purpose utility library Deprecated Function Interfaces

<code>g_assert_warning</code> [Glib 2.32]	<code>g_async_queue_ref_unlocked</code> [Glib 2.32]
<code>g_async_queue_unref_and_unlock</code> [Glib 2.32]	<code>g_basename</code> [Glib 2.32]
<code>g_cond_free</code> [Glib 2.32]	<code>g_cond_new</code> [Glib 2.32]
<code>g_cond_timed_wait</code> [Glib 2.32]	<code>g_format_size_for_display</code> [Glib 2.32]
<code>g_io_channel_close</code> [Glib 2.32]	<code>g_io_channel_read</code> [Glib 2.32]
<code>g_io_channel_seek</code> [Glib 2.32]	<code>g_io_channel_write</code> [Glib 2.32]
<code>g_mutex_free</code> [Glib 2.32]	<code>g_mutex_new</code> [Glib 2.32]
<code>g_private_new</code> [Glib 2.32]	<code>g_strcasecmp</code> [Glib 2.32]

<code>g_strdown</code> [Glib 2.32]	<code>g_string_down</code> [Glib 2.32]
<code>g_string_up</code> [Glib 2.32]	<code>g_strncasecmp</code> [Glib 2.32]
<code>g_strup</code> [Glib 2.32]	<code>g_thread_create</code> [Glib 2.32]
<code>g_tree_traverse</code> [Glib 2.32]	

An LSB conforming implementation shall provide the generic data interfaces for GTK General purpose utility library specified in Table 17-4, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-4 libglib-2.0 - GTK General purpose utility library Data Interfaces

<code>g_ascii_table</code> [Glib 2.32]	<code>g_child_watch_funcs</code> [Glib 2.32]
<code>g_idle_funcs</code> [Glib 2.32]	<code>g_io_watch_funcs</code> [Glib 2.32]
<code>g_test_config_vars</code> [LSB]	<code>g_thread_functions_for_glib_use</code> [Glib 2.32]
<code>g_thread_use_default_impl</code> [Glib 2.32]	<code>g_threads_got_initialized</code> [Glib 2.32]
<code>g_timeout_funcs</code> [Glib 2.32]	<code>g_utf8_skip</code> [Glib 2.32]

17.2.2 Glib Arrays

17.2.2.1 Interfaces for Glib Arrays

An LSB conforming implementation shall provide the generic functions for Glib Arrays specified in Table 17-5, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-5 libglib-2.0 - Glib Arrays Function Interfaces

<code>g_array_append_vals</code> [Glib 2.32]	<code>g_array_free</code> [Glib 2.32]
<code>g_array_insert_vals</code> [Glib 2.32]	<code>g_array_new</code> [Glib 2.32]
<code>g_array_prepend_vals</code> [Glib 2.32]	<code>g_array_remove_index</code> [Glib 2.32]
<code>g_array_remove_index_fast</code> [Glib 2.32]	<code>g_array_remove_range</code> [Glib 2.32]
<code>g_array_set_size</code> [Glib 2.32]	<code>g_array_sized_new</code> [Glib 2.32]
<code>g_array_sort</code> [Glib 2.32]	<code>g_array_sort_with_data</code> [Glib 2.32]

17.2.3 Glib Asynchronous Queues

17.2.3.1 Interfaces for Glib Asynchronous Queues

An LSB conforming implementation shall provide the generic functions for Glib Asynchronous Queues specified in Table 17-6, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-6 libglib-2.0 - Glib Asynchronous Queues Function Interfaces

<code>g_async_queue_length</code> [Glib 2.32]	<code>g_async_queue_length_unlocked</code> [Glib 2.32]

g_async_queue_lock [Glib 2.32]	g_async_queue_new [Glib 2.32]
g_async_queue_pop [Glib 2.32]	g_async_queue_pop_unlocked [Glib 2.32]
g_async_queue_push [Glib 2.32]	g_async_queue_push_sorted [Glib 2.32]
g_async_queue_push_sorted_unlocked [Glib 2.32]	g_async_queue_push_unlocked [Glib 2.32]
g_async_queue_ref [Glib 2.32]	g_async_queue_sort [Glib 2.32]
g_async_queue_sort_unlocked [Glib 2.32]	g_async_queue_timed_pop [Glib 2.32]
g_async_queue_timed_pop_unlocked [Glib 2.32]	g_async_queue_try_pop [Glib 2.32]
g_async_queue_try_pop_unlocked [Glib 2.32]	g_async_queue_unlock [Glib 2.32]
g_async_queue_unref [Glib 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Glib Asynchronous Queues specified in Table 17-7, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-7 libglib-2.0 - Glib Asynchronous Queues Deprecated Function Interfaces

g_async_queue_timed_pop [Glib 2.32]	g_async_queue_timed_pop_unlocked [Glib 2.32]

17.2.4 Glib Atomic Operations

17.2.4.1 Interfaces for Glib Atomic Operations

An LSB conforming implementation shall provide the generic functions for Glib Atomic Operations specified in Table 17-8, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-8 libglib-2.0 - Glib Atomic Operations Function Interfaces

g_atomic_int_add [Glib 2.32]	g_atomic_int_compare_and_exchange [Glib 2.32]
g_atomic_int_exchange_and_add [Glib 2.32]	g_atomic_pointer_compare_and_exchange [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Atomic Operations specified in Table 17-9, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-9 libglib-2.0 - Glib Atomic Operations Deprecated Function Interfaces

g_atomic_int_exchange_and_add [Glib 2.32]	

17.2.5 Glib Automatic String Completion

17.2.5.1 Interfaces for Glib Automatic String Completion

An LSB conforming implementation shall provide the generic functions for Glib Automatic String Completion specified in Table 17-10, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-10 libglib-2.0 - Glib Automatic String Completion Function Interfaces

g_completion_add_items [Glib 2.32]	g_completion_clear_items [Glib 2.32]
g_completion_complete [Glib 2.32]	g_completion_complete_utf8 [Glib 2.32]
g_completion_free [Glib 2.32]	g_completion_new [Glib 2.32]
g_completion_remove_items [Glib 2.32]	g_completion_set_compare [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Automatic String Completion specified in Table 17-11, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-11 libglib-2.0 - Glib Automatic String Completion Deprecated Function Interfaces

g_completion_add_items [Glib 2.32]	g_completion_clear_items [Glib 2.32]
g_completion_complete [Glib 2.32]	g_completion_complete_utf8 [Glib 2.32]
g_completion_free [Glib 2.32]	g_completion_new [Glib 2.32]
g_completion_remove_items [Glib 2.32]	g_completion_set_compare [Glib 2.32]

17.2.6 Glib Balanced Binary Trees

17.2.6.1 Interfaces for Glib Balanced Binary Trees

An LSB conforming implementation shall provide the generic functions for Glib Balanced Binary Trees specified in Table 17-12, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-12 libglib-2.0 - Glib Balanced Binary Trees Function Interfaces

g_tree_destroy [Glib 2.32]	g_tree_foreach [Glib 2.32]
g_tree_height [Glib 2.32]	g_tree_insert [Glib 2.32]

<code>g_tree_lookup</code> [Glib 2.32]	<code>g_tree_lookup_extended</code> [Glib 2.32]
<code>g_tree_new</code> [Glib 2.32]	<code>g_tree_new_full</code> [Glib 2.32]
<code>g_tree_new_with_data</code> [Glib 2.32]	<code>g_tree_nnodes</code> [Glib 2.32]
<code>g_tree_remove</code> [Glib 2.32]	<code>g_tree_replace</code> [Glib 2.32]
<code>g_tree_search</code> [Glib 2.32]	<code>g_tree_steal</code> [Glib 2.32]

17.2.7 Glib Byte Arrays

17.2.7.1 Interfaces for Glib Byte Arrays

An LSB conforming implementation shall provide the generic functions for Glib Byte Arrays specified in Table 17-13, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-13 libglib-2.0 - Glib Byte Arrays Function Interfaces

<code>g_byte_array_append</code> [Glib 2.32]	<code>g_byte_array_free</code> [Glib 2.32]
<code>g_byte_array_new</code> [Glib 2.32]	<code>g_byte_array_prepend</code> [Glib 2.32]
<code>g_byte_array_remove_index</code> [Glib 2.32]	<code>g_byte_array_remove_index_fast</code> [Glib 2.32]
<code>g_byte_array_remove_range</code> [Glib 2.32]	<code>g_byte_array_set_size</code> [Glib 2.32]
<code>g_byte_array_sized_new</code> [Glib 2.32]	<code>g_byte_array_sort</code> [Glib 2.32]
<code>g_byte_array_sort_with_data</code> [Glib 2.32]	

17.2.8 Glib Caches

17.2.8.1 Interfaces for Glib Caches

An LSB conforming implementation shall provide the generic functions for Glib Caches specified in Table 17-14, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-14 libglib-2.0 - Glib Caches Function Interfaces

<code>g_cache_destroy</code> [Glib 2.32]	<code>g_cache_insert</code> [Glib 2.32]
<code>g_cache_key_foreach</code> [Glib 2.32]	<code>g_cache_new</code> [Glib 2.32]
<code>g_cache_remove</code> [Glib 2.32]	<code>g_cache_value_foreach</code> [LSB]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Caches specified in Table 17-15, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-15 libglib-2.0 - Glib Caches Deprecated Function Interfaces

<code>g_cache_destroy</code> [Glib 2.32]	<code>g_cache_insert</code> [Glib 2.32]

<code>g_cache_key_foreach</code> [Glib 2.32]	<code>g_cache_new</code> [Glib 2.32]
<code>g_cache_remove</code> [Glib 2.32]	<code>g_cache_value_foreach</code> [LSB]

17.2.9 Glib Character Set Conversion

17.2.9.1 Interfaces for Glib Character Set Conversion

An LSB conforming implementation shall provide the generic functions for Glib Character Set Conversion specified in Table 17-16, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-16 libglib-2.0 - Glib Character Set Conversion Function Interfaces

<code>g_convert</code> [Glib 2.32]	<code>g_convert_error_quark</code> [Glib 2.32]
<code>g_convert_with_fallback</code> [Glib 2.32]	<code>g_convert_with_iconv</code> [Glib 2.32]
<code>g_filename_display_basename</code> [Glib 2.32]	<code>g_filename_display_name</code> [Glib 2.32]
<code>g_filename_from_uri</code> [Glib 2.32]	<code>g_filename_from_utf8</code> [Glib 2.32]
<code>g_filename_to_uri</code> [Glib 2.32]	<code>g_filename_to_utf8</code> [Glib 2.32]
<code>g_get_charset</code> [Glib 2.32]	<code>g_get_filename_charsets</code> [Glib 2.32]
<code>g_iconv</code> [Glib 2.32]	<code>g_iconv_close</code> [Glib 2.32]
<code>g_iconv_open</code> [Glib 2.32]	<code>g_locale_from_utf8</code> [Glib 2.32]
<code>g_locale_to_utf8</code> [Glib 2.32]	<code>g_uri_list_extract_uris</code> [Glib 2.32]

17.2.10 Glib Commandline Option Parser

17.2.10.1 Interfaces for Glib Commandline Option Parser

An LSB conforming implementation shall provide the generic functions for Glib Commandline Option Parser specified in Table 17-17, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-17 libglib-2.0 - Glib Commandline Option Parser Function Interfaces

<code>g_option_context_add_group</code> [Glib 2.32]	<code>g_option_context_add_main_entries</code> [Glib 2.32]
<code>g_option_context_free</code> [Glib 2.32]	<code>g_option_context_get_description</code> [Glib 2.32]
<code>g_option_context_get_help_enabled</code> [Glib 2.32]	<code>g_option_context_get_ignore_unknown_options</code> [Glib 2.32]
<code>g_option_context_get_main_group</code> [Glib 2.32]	<code>g_option_context_get_summary</code> [Glib 2.32]
<code>g_option_context_new</code> [Glib 2.32]	<code>g_option_context_parse</code> [Glib 2.32]
<code>g_option_context_set_description</code> [Glib 2.32]	<code>g_option_context_set_help_enabled</code> [Glib 2.32]
<code>g_option_context_set_ignore_unknown_options</code> [Glib 2.32]	<code>g_option_context_set_main_group</code> [Glib 2.32]

g_option_context_set_summary [Glib 2.32]	g_option_context_set_translate_func [Glib 2.32]
g_option_context_set_translation_do_main [Glib 2.32]	g_option_error_quark [Glib 2.32]
g_option_group_add_entries [Glib 2.32]	g_option_group_free [Glib 2.32]
g_option_group_new [Glib 2.32]	g_option_group_set_error_hook [Glib 2.32]
g_option_group_set_parse_hooks [Glib 2.32]	g_option_group_set_translate_func [Glib 2.32]
g_option_group_set_translation_do_main [Glib 2.32]	

17.2.11 Glib Datasets

17.2.11.1 Interfaces for Glib Datasets

An LSB conforming implementation shall provide the generic functions for Glib Datasets specified in Table 17-18, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-18 libglib-2.0 - Glib Datasets Function Interfaces

g_dataset_destroy [Glib 2.32]	g_dataset_foreach [Glib 2.32]
g_dataset_id_get_data [Glib 2.32]	g_dataset_id_remove_no_notify [Glib 2.32]
g_dataset_id_set_data_full [Glib 2.32]	

17.2.12 Glib Date and Time Functions

17.2.12.1 Interfaces for Glib Date and Time Functions

An LSB conforming implementation shall provide the generic functions for Glib Date and Time Functions specified in Table 17-19, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-19 libglib-2.0 - Glib Date and Time Functions Function Interfaces

g_date_add_days [Glib 2.32]	g_date_add_months [Glib 2.32]
g_date_add_years [Glib 2.32]	g_date_clamp [Glib 2.32]
g_date_clear [Glib 2.32]	g_date_compare [Glib 2.32]
g_date_days_between [Glib 2.32]	g_date_free [Glib 2.32]
g_date_get_day [Glib 2.32]	g_date_get_day_of_year [Glib 2.32]
g_date_get_days_in_month [Glib 2.32]	g_date_get_iso8601_week_of_year [Glib 2.32]
g_date_get_julian [Glib 2.32]	g_date_get_monday_week_of_year [Glib 2.32]

<code>g_date_get_monday_weeks_in_year</code> [Glib 2.32]	<code>g_date_get_month</code> [Glib 2.32]
<code>g_date_get_sunday_week_of_year</code> [Glib 2.32]	<code>g_date_get_sunday_weeks_in_year</code> [Glib 2.32]
<code>g_date_get_weekday</code> [Glib 2.32]	<code>g_date_get_year</code> [Glib 2.32]
<code>g_date_is_first_of_month</code> [Glib 2.32]	<code>g_date_is_last_of_month</code> [Glib 2.32]
<code>g_date_is_leap_year</code> [Glib 2.32]	<code>g_date_new</code> [Glib 2.32]
<code>g_date_new_dmy</code> [Glib 2.32]	<code>g_date_new_julian</code> [Glib 2.32]
<code>g_date_order</code> [Glib 2.32]	<code>g_date_set_day</code> [Glib 2.32]
<code>g_date_set_dmy</code> [Glib 2.32]	<code>g_date_set_julian</code> [Glib 2.32]
<code>g_date_set_month</code> [Glib 2.32]	<code>g_date_set_parse</code> [Glib 2.32]
<code>g_date_set_time</code> [Glib 2.32]	<code>g_date_set_time_t</code> [Glib 2.32]
<code>g_date_set_time_val</code> [Glib 2.32]	<code>g_date_set_year</code> [Glib 2.32]
<code>g_date_strftime</code> [Glib 2.32]	<code>g_date_subtract_days</code> [Glib 2.32]
<code>g_date_subtract_months</code> [Glib 2.32]	<code>g_date_subtract_years</code> [Glib 2.32]
<code>g_date_to_struct_tm</code> [Glib 2.32]	<code>g_date_valid</code> [Glib 2.32]
<code>g_date_valid_day</code> [Glib 2.32]	<code>g_date_valid_dmy</code> [Glib 2.32]
<code>g_date_valid_julian</code> [Glib 2.32]	<code>g_date_valid_month</code> [Glib 2.32]
<code>g_date_valid_weekday</code> [Glib 2.32]	<code>g_date_valid_year</code> [Glib 2.32]
<code>g_get_current_time</code> [Glib 2.32]	<code>g_time_val_add</code> [Glib 2.32]
<code>g_time_val_from_iso8601</code> [Glib 2.32]	<code>g_time_val_to_iso8601</code> [Glib 2.32]
<code>g_usleep</code> [Glib 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Glib Date and Time Functions specified in Table 17-20, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-20 libglib-2.0 - Glib Date and Time Functions Deprecated Function Interfaces

<code>g_date_set_time</code> [Glib 2.32]	

17.2.13 Glib Double-Ended Queues

17.2.13.1 Interfaces for Glib Double-Ended Queues

An LSB conforming implementation shall provide the generic functions for Glib Double-Ended Queues specified in Table 17-21, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-21 libglib-2.0 - Glib Double-Ended Queues Function Interfaces

g_queue_copy [Glib 2.32]	g_queue_delete_link [Glib 2.32]
g_queue_find [Glib 2.32]	g_queue_find_custom [Glib 2.32]
g_queue_foreach [Glib 2.32]	g_queue_free [Glib 2.32]
g_queue_get_length [Glib 2.32]	g_queue_index [Glib 2.32]
g_queue_insert_after [Glib 2.32]	g_queue_insert_before [Glib 2.32]
g_queue_insert_sorted [Glib 2.32]	g_queue_is_empty [Glib 2.32]
g_queue_link_index [Glib 2.32]	g_queue_new [Glib 2.32]
g_queue_peek_head [Glib 2.32]	g_queue_peek_head_link [Glib 2.32]
g_queue_peek_nth [Glib 2.32]	g_queue_peek_nth_link [Glib 2.32]
g_queue_peek_tail [Glib 2.32]	g_queue_peek_tail_link [Glib 2.32]
g_queue_pop_head [Glib 2.32]	g_queue_pop_head_link [Glib 2.32]
g_queue_pop_nth [Glib 2.32]	g_queue_pop_nth_link [Glib 2.32]
g_queue_pop_tail [Glib 2.32]	g_queue_pop_tail_link [Glib 2.32]
g_queue_push_head [Glib 2.32]	g_queue_push_head_link [Glib 2.32]
g_queue_push_nth [Glib 2.32]	g_queue_push_nth_link [Glib 2.32]
g_queue_push_tail [Glib 2.32]	g_queue_push_tail_link [Glib 2.32]
g_queue_remove [Glib 2.32]	g_queue_remove_all [Glib 2.32]
g_queue_reverse [Glib 2.32]	g_queue_sort [Glib 2.32]
g_queue_unlink [Glib 2.32]	

17.2.14 Glib Doubly-Linked Lists

17.2.14.1 Interfaces for Glib Doubly-Linked Lists

An LSB conforming implementation shall provide the generic functions for Glib Doubly-Linked Lists specified in Table 17-22, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-22 libglib-2.0 - Glib Doubly-Linked Lists Function Interfaces

g_list_alloc [Glib 2.32]	g_list_append [Glib 2.32]
g_list_concat [Glib 2.32]	g_list_copy [Glib 2.32]
g_list_delete_link [Glib 2.32]	g_list_find [Glib 2.32]
g_list_find_custom [Glib 2.32]	g_list_first [Glib 2.32]
g_list_foreach [Glib 2.32]	g_list_free [Glib 2.32]
g_list_free_1 [Glib 2.32]	g_list_index [Glib 2.32]
g_list_insert [Glib 2.32]	g_list_insert_before [Glib 2.32]
g_list_insert_sorted [Glib 2.32]	g_list_insert_sorted_with_data [Glib 2.32]

<code>g_list_last</code> [Glib 2.32]	<code>g_list_length</code> [Glib 2.32]
<code>g_list_nth</code> [Glib 2.32]	<code>g_list_nth_data</code> [Glib 2.32]
<code>g_list_nth_prev</code> [Glib 2.32]	<code>g_list_pop_allocator</code> [Glib 2.32]
<code>g_list_position</code> [Glib 2.32]	<code>g_list_prepend</code> [Glib 2.32]
<code>g_list_push_allocator</code> [Glib 2.32]	<code>g_list_remove</code> [Glib 2.32]
<code>g_list_remove_all</code> [Glib 2.32]	<code>g_list_remove_link</code> [Glib 2.32]
<code>g_list_reverse</code> [Glib 2.32]	<code>g_list_sort</code> [Glib 2.32]
<code>g_list_sort_with_data</code> [Glib 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Glib Doubly-Linked Lists specified in Table 17-23, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-23 libglib-2.0 - Glib Doubly-Linked Lists Deprecated Function Interfaces

<code>g_list_pop_allocator</code> [Glib 2.32]	<code>g_list_push_allocator</code> [Glib 2.32]

17.2.15 Glib Error Reporting

17.2.15.1 Interfaces for Glib Error Reporting

An LSB conforming implementation shall provide the generic functions for Glib Error Reporting specified in Table 17-24, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-24 libglib-2.0 - Glib Error Reporting Function Interfaces

<code>g_clear_error</code> [Glib 2.32]	<code>g_error_copy</code> [Glib 2.32]
<code>g_error_free</code> [Glib 2.32]	<code>g_error_matches</code> [Glib 2.32]
<code>g_error_new</code> [Glib 2.32]	<code>g_error_new_literal</code> [Glib 2.32]
<code>g_propagate_error</code> [Glib 2.32]	<code>g_set_error</code> [Glib 2.32]

17.2.16 Glib File Utilities

17.2.16.1 Interfaces for Glib File Utilities

An LSB conforming implementation shall provide the generic functions for Glib File Utilities specified in Table 17-25, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-25 libglib-2.0 - Glib File Utilities Function Interfaces

<code>g_dir_close</code> [Glib 2.32]	<code>g_dir_open</code> [Glib 2.32]
<code>g_dir_read_name</code> [Glib 2.32]	<code>g_dir_rewind</code> [Glib 2.32]
<code>g_file_error_from_errno</code> [Glib 2.32]	<code>g_file_error_quark</code> [Glib 2.32]

<code>g_file_get_contents</code> [Glib 2.32]	<code>g_file_open_tmp</code> [Glib 2.32]
<code>g_file_read_link</code> [Glib 2.32]	<code>g_file_set_contents</code> [Glib 2.32]
<code>g_file_test</code> [Glib 2.32]	<code>g_mapped_file_free</code> [Glib 2.32]
<code>g_mapped_file_get_contents</code> [Glib 2.32]	<code>g_mapped_file_get_length</code> [Glib 2.32]
<code>g_mapped_file_new</code> [Glib 2.32]	<code>g_mkstemp</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib File Utilities specified in Table 17-26, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-26 libglib-2.0 - Glib File Utilities Deprecated Function Interfaces

<code>g_mapped_file_free</code> [Glib 2.32]	

17.2.17 Glib Glob-Style Pattern Matching

17.2.17.1 Interfaces for Glib Glob-Style Pattern Matching

An LSB conforming implementation shall provide the generic functions for Glib Glob-Style Pattern Matching specified in Table 17-27, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-27 libglib-2.0 - Glib Glob-Style Pattern Matching Function Interfaces

<code>g_pattern_match</code> [Glib 2.32]	<code>g_pattern_match_simple</code> [Glib 2.32]
<code>g_pattern_match_string</code> [Glib 2.32]	<code>g_pattern_spec_equal</code> [Glib 2.32]
<code>g_pattern_spec_free</code> [Glib 2.32]	<code>g_pattern_spec_new</code> [Glib 2.32]

17.2.18 Glib Hash Tables

17.2.18.1 Interfaces for Glib Hash Tables

An LSB conforming implementation shall provide the generic functions for Glib Hash Tables specified in Table 17-28, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-28 libglib-2.0 - Glib Hash Tables Function Interfaces

<code>g_direct_equal</code> [Glib 2.32]	<code>g_direct_hash</code> [Glib 2.32]
<code>g_hash_table_destroy</code> [Glib 2.32]	<code>g_hash_table_find</code> [Glib 2.32]
<code>g_hash_table_foreach</code> [Glib 2.32]	<code>g_hash_table_foreach_remove</code> [Glib 2.32]
<code>g_hash_table_foreach_steal</code> [Glib 2.32]	<code>g_hash_table_insert</code> [Glib 2.32]
<code>g_hash_table_lookup</code> [Glib 2.32]	<code>g_hash_table_lookup_extended</code> [Glib 2.32]

<code>g_hash_table_new</code> [Glib 2.32]	<code>g_hash_table_new_full</code> [Glib 2.32]
<code>g_hash_table_ref</code> [Glib 2.32]	<code>g_hash_table_remove</code> [Glib 2.32]
<code>g_hash_table_remove_all</code> [Glib 2.32]	<code>g_hash_table_replace</code> [Glib 2.32]
<code>g_hash_table_size</code> [Glib 2.32]	<code>g_hash_table_steal</code> [Glib 2.32]
<code>g_hash_table_steal_all</code> [Glib 2.32]	<code>g_hash_table_unref</code> [Glib 2.32]
<code>g_int_equal</code> [Glib 2.32]	<code>g_int_hash</code> [Glib 2.32]
<code>g_str_equal</code> [Glib 2.32]	<code>g_str_hash</code> [Glib 2.32]

17.2.19 Glib Hook Functions

17.2.19.1 Interfaces for Glib Hook Functions

An LSB conforming implementation shall provide the generic functions for Glib Hook Functions specified in Table 17-29, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-29 libglib-2.0 - Glib Hook Functions Function Interfaces

<code>g_hook_alloc</code> [Glib 2.32]	<code>g_hook_compare_ids</code> [Glib 2.32]
<code>g_hook_destroy</code> [Glib 2.32]	<code>g_hook_destroy_link</code> [Glib 2.32]
<code>g_hook_find</code> [Glib 2.32]	<code>g_hook_find_data</code> [Glib 2.32]
<code>g_hook_find_func</code> [Glib 2.32]	<code>g_hook_find_func_data</code> [Glib 2.32]
<code>g_hook_first_valid</code> [Glib 2.32]	<code>g_hook_free</code> [Glib 2.32]
<code>g_hook_get</code> [Glib 2.32]	<code>g_hook_insert_before</code> [Glib 2.32]
<code>g_hook_insert_sorted</code> [Glib 2.32]	<code>g_hook_list_clear</code> [Glib 2.32]
<code>g_hook_list_init</code> [Glib 2.32]	<code>g_hook_list_invoke</code> [Glib 2.32]
<code>g_hook_list_invoke_check</code> [Glib 2.32]	<code>g_hook_list_marshall</code> [Glib 2.32]
<code>g_hook_list_marshall_check</code> [Glib 2.32]	<code>g_hook_next_valid</code> [Glib 2.32]
<code>g_hook_prepend</code> [Glib 2.32]	<code>g_hook_ref</code> [Glib 2.32]
<code>g_hook_unref</code> [Glib 2.32]	

17.2.20 Glib IO Channels

17.2.20.1 Interfaces for Glib IO Channels

An LSB conforming implementation shall provide the generic functions for Glib IO Channels specified in Table 17-30, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-30 libglib-2.0 - Glib IO Channels Function Interfaces

<code>g_io_add_watch</code> [Glib 2.32]	<code>g_io_add_watch_full</code> [Glib 2.32]
<code>g_io_channel_error_from_errno</code> [Glib 2.32]	<code>g_io_channel_error_quark</code> [Glib 2.32]

<code>g_io_channel_flush</code> [Glib 2.32]	<code>g_io_channel_get_buffer_condition</code> [Glib 2.32]
<code>g_io_channel_get_buffer_size</code> [Glib 2.32]	<code>g_io_channel_get_buffered</code> [Glib 2.32]
<code>g_io_channel_get_close_on_unref</code> [Glib 2.32]	<code>g_io_channel_get_encoding</code> [Glib 2.32]
<code>g_io_channel_get_flags</code> [Glib 2.32]	<code>g_io_channel_get_line_term</code> [Glib 2.32]
<code>g_io_channel_init</code> [Glib 2.32]	<code>g_io_channel_new_file</code> [Glib 2.32]
<code>g_io_channel_read_chars</code> [Glib 2.32]	<code>g_io_channel_read_line</code> [Glib 2.32]
<code>g_io_channel_read_line_string</code> [Glib 2.32]	<code>g_io_channel_read_to_end</code> [Glib 2.32]
<code>g_io_channel_read_unichar</code> [Glib 2.32]	<code>g_io_channel_ref</code> [Glib 2.32]
<code>g_io_channel_seek_position</code> [Glib 2.32]	<code>g_io_channel_set_buffer_size</code> [Glib 2.32]
<code>g_io_channel_set_buffered</code> [Glib 2.32]	<code>g_io_channel_set_close_on_unref</code> [Glib 2.32]
<code>g_io_channel_set_encoding</code> [Glib 2.32]	<code>g_io_channel_set_flags</code> [Glib 2.32]
<code>g_io_channel_set_line_term</code> [Glib 2.32]	<code>g_io_channel_shutdown</code> [Glib 2.32]
<code>g_io_channel_unix_get_fd</code> [Glib 2.32]	<code>g_io_channel_unix_new</code> [Glib 2.32]
<code>g_io_channel_unref</code> [Glib 2.32]	<code>g_io_channel_write_chars</code> [Glib 2.32]
<code>g_io_channel_write_unichar</code> [Glib 2.32]	<code>g_io_create_watch</code> [Glib 2.32]

17.2.21 Glib Internationalization

17.2.21.1 Interfaces for Glib Internationalization

An LSB conforming implementation shall provide the generic functions for Glib Internationalization specified in Table 17-31, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-31 libglib-2.0 - Glib Internationalization Function Interfaces

<code>g_get_language_names</code> [Glib 2.32]	<code>g_strip_context</code> [Glib 2.32]

17.2.22 Glib Key-Value File Parser

17.2.22.1 Interfaces for Glib Key-Value File Parser

An LSB conforming implementation shall provide the generic functions for Glib Key-Value File Parser specified in Table 17-32, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-32 libglib-2.0 - Glib Key-Value File Parser Function Interfaces

g_key_file_error_quark [Glib 2.32]	g_key_file_free [Glib 2.32]
g_key_file_get_boolean [Glib 2.32]	g_key_file_get_boolean_list [Glib 2.32]
g_key_file_get_comment [Glib 2.32]	g_key_file_get_double [Glib 2.32]
g_key_file_get_double_list [Glib 2.32]	g_key_file_get_groups [Glib 2.32]
g_key_file_get_integer [Glib 2.32]	g_key_file_get_integer_list [Glib 2.32]
g_key_file_get_keys [Glib 2.32]	g_key_file_get_locale_string [Glib 2.32]
g_key_file_get_locale_string_list [Glib 2.32]	g_key_file_get_start_group [Glib 2.32]
g_key_file_get_string [Glib 2.32]	g_key_file_get_string_list [Glib 2.32]
g_key_file_get_value [Glib 2.32]	g_key_file_has_group [Glib 2.32]
g_key_file_has_key [Glib 2.32]	g_key_file_load_from_data [Glib 2.32]
g_key_file_load_from_data_dirs [Glib 2.32]	g_key_file_load_from_file [Glib 2.32]
g_key_file_new [Glib 2.32]	g_key_file_remove_comment [Glib 2.32]
g_key_file_remove_group [Glib 2.32]	g_key_file_remove_key [Glib 2.32]
g_key_file_set_boolean [Glib 2.32]	g_key_file_set_boolean_list [Glib 2.32]
g_key_file_set_comment [Glib 2.32]	g_key_file_set_double [Glib 2.32]
g_key_file_set_double_list [Glib 2.32]	g_key_file_set_integer [Glib 2.32]
g_key_file_set_integer_list [Glib 2.32]	g_key_file_set_list_separator [Glib 2.32]
g_key_file_set_locale_string [Glib 2.32]	g_key_file_set_locale_string_list [Glib 2.32]
g_key_file_set_string [Glib 2.32]	g_key_file_set_string_list [Glib 2.32]
g_key_file_set_value [Glib 2.32]	g_key_file_to_data [Glib 2.32]

17.2.23 Glib Keyed Data Lists

17.2.23.1 Interfaces for Glib Keyed Data Lists

An LSB conforming implementation shall provide the generic functions for Glib Keyed Data Lists specified in Table 17-33, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-33 libglib-2.0 - Glib Keyed Data Lists Function Interfaces

g_datalist_clear [Glib 2.32]	g_datalist_foreach [Glib 2.32]
g_datalist_get_flags [Glib 2.32]	g_datalist_id_get_data [Glib 2.32]

<code>g_datalist_id_remove_no_notify</code> [Glib 2.32]	<code>g_datalist_id_set_data_full</code> [Glib 2.32]
<code>g_datalist_init</code> [Glib 2.32]	<code>g_datalist_set_flags</code> [Glib 2.32]
<code>g_datalist_unset_flags</code> [Glib 2.32]	

17.2.24 Glib Lexical Scanner

17.2.24.1 Interfaces for Glib Lexical Scanner

An LSB conforming implementation shall provide the generic functions for Glib Lexical Scanner specified in Table 17-34, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-34 libglib-2.0 - Glib Lexical Scanner Function Interfaces

<code>g_scanner_cur_line</code> [Glib 2.32]	<code>g_scanner_cur_position</code> [Glib 2.32]
<code>g_scanner_cur_token</code> [Glib 2.32]	<code>g_scanner_cur_value</code> [Glib 2.32]
<code>g_scanner_destroy</code> [Glib 2.32]	<code>g_scanner_eof</code> [Glib 2.32]
<code>g_scanner_error</code> [Glib 2.32]	<code>g_scanner_get_next_token</code> [Glib 2.32]
<code>g_scanner_input_file</code> [Glib 2.32]	<code>g_scanner_input_text</code> [Glib 2.32]
<code>g_scanner_lookup_symbol</code> [Glib 2.32]	<code>g_scanner_new</code> [Glib 2.32]
<code>g_scanner_peek_next_token</code> [Glib 2.32]	<code>g_scanner_scope_add_symbol</code> [Glib 2.32]
<code>g_scanner_scope_foreach_symbol</code> [Glib 2.32]	<code>g_scanner_scope_lookup_symbol</code> [Glib 2.32]
<code>g_scanner_scope_remove_symbol</code> [Glib 2.32]	<code>g_scanner_set_scope</code> [Glib 2.32]
<code>g_scanner_sync_file_offset</code> [Glib 2.32]	<code>g_scanner_unexp_token</code> [Glib 2.32]
<code>g_scanner_warn</code> [Glib 2.32]	

17.2.25 Glib Memory Allocation

17.2.25.1 Interfaces for Glib Memory Allocation

An LSB conforming implementation shall provide the generic functions for Glib Memory Allocation specified in Table 17-35, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-35 libglib-2.0 - Glib Memory Allocation Function Interfaces

<code>g_free</code> [Glib 2.32]	<code>g_malloc</code> [Glib 2.32]
<code>g_malloc0</code> [Glib 2.32]	<code>g_mem_is_system_malloc</code> [Glib 2.32]
<code>g_mem_profile</code> [Glib 2.32]	<code>g_mem_set_vtable</code> [Glib 2.32]
<code>g_memdup</code> [Glib 2.32]	<code>g_realloc</code> [Glib 2.32]
<code>g_try_malloc</code> [Glib 2.32]	<code>g_try_realloc</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic data interfaces for Glib Memory Allocation specified in Table 17-36, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-36 libglib-2.0 - Glib Memory Allocation Data Interfaces

<code>g_mem_gc_friendly</code> [Glib 2.32]	

17.2.26 Glib Memory Allocators

17.2.26.1 Interfaces for Glib Memory Allocators

An LSB conforming implementation shall provide the generic functions for Glib Memory Allocators specified in Table 17-37, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-37 libglib-2.0 - Glib Memory Allocators Function Interfaces

<code>g_allocator_free</code> [Glib 2.32]	<code>g_allocator_new</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Memory Allocators specified in Table 17-38, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-38 libglib-2.0 - Glib Memory Allocators Deprecated Function Interfaces

<code>g_allocator_free</code> [Glib 2.32]	<code>g_allocator_new</code> [Glib 2.32]

17.2.27 Glib Memory Chunks

17.2.27.1 Interfaces for Glib Memory Chunks

An LSB conforming implementation shall provide the generic functions for Glib Memory Chunks specified in Table 17-39, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-39 libglib-2.0 - Glib Memory Chunks Function Interfaces

<code>g_blow_chunks</code> [Glib 2.32]	<code>g_mem_chunk_alloc</code> [Glib 2.32]
<code>g_mem_chunk_alloc0</code> [Glib 2.32]	<code>g_mem_chunk_clean</code> [Glib 2.32]
<code>g_mem_chunk_destroy</code> [Glib 2.32]	<code>g_mem_chunk_free</code> [Glib 2.32]
<code>g_mem_chunk_info</code> [Glib 2.32]	<code>g_mem_chunk_new</code> [Glib 2.32]
<code>g_mem_chunk_print</code> [Glib 2.32]	<code>g_mem_chunk_reset</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Memory Chunks specified in Table 17-40, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-40 libglib-2.0 - Glib Memory Chunks Deprecated Function Interfaces

<code>g_blow_chunks</code> [Glib 2.32]	<code>g_mem_chunk_alloc</code> [Glib 2.32]
<code>g_mem_chunk_alloc0</code> [Glib 2.32]	<code>g_mem_chunk_clean</code> [Glib 2.32]
<code>g_mem_chunk_destroy</code> [Glib 2.32]	<code>g_mem_chunk_free</code> [Glib 2.32]
<code>g_mem_chunk_info</code> [Glib 2.32]	<code>g_mem_chunk_new</code> [Glib 2.32]
<code>g_mem_chunk_print</code> [Glib 2.32]	<code>g_mem_chunk_reset</code> [Glib 2.32]

17.2.28 Glib Message Logging

17.2.28.1 Interfaces for Glib Message Logging

An LSB conforming implementation shall provide the generic functions for Glib Message Logging specified in Table 17-41, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-41 libglib-2.0 - Glib Message Logging Function Interfaces

<code>g_log</code> [Glib 2.32]	<code>g_log_default_handler</code> [Glib 2.32]
<code>g_log_remove_handler</code> [Glib 2.32]	<code>g_log_set_always_fatal</code> [Glib 2.32]
<code>g_log_set_default_handler</code> [Glib 2.32]	<code>g_log_set_fatal_mask</code> [Glib 2.32]
<code>g_log_set_handler</code> [Glib 2.32]	<code>g_logv</code> [Glib 2.32]

17.2.29 Glib Message Output and Debugging Functions

17.2.29.1 Interfaces for Glib Message Output and Debugging Functions

An LSB conforming implementation shall provide the generic functions for Glib Message Output and Debugging Functions specified in Table 17-42, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-42 libglib-2.0 - Glib Message Output and Debugging Functions Function Interfaces

<code>g_on_error_query</code> [Glib 2.32]	<code>g_on_error_stack_trace</code> [Glib 2.32]
<code>g_print</code> [Glib 2.32]	<code>g_printerr</code> [Glib 2.32]
<code>g_set_print_handler</code> [Glib 2.32]	<code>g_set_printerr_handler</code> [Glib 2.32]

17.2.30 Glib Miscellaneous Utility Functions

17.2.30.1 Interfaces for Glib Miscellaneous Utility Functions

An LSB conforming implementation shall provide the generic functions for Glib Miscellaneous Utility Functions specified in Table 17-43, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-43 libglib-2.0 - Glib Miscellaneous Utility Functions Function Interfaces

<code>g_atexit</code> [Glib 2.32]	<code>g_bit_nth_lsf</code> [Glib 2.32]

<code>g_bit_nth_msf</code> [Glib 2.32]	<code>g_bit_storage</code> [Glib 2.32]
<code>g_build_filename</code> [Glib 2.32]	<code>g_build_filenamev</code> [Glib 2.32]
<code>g_build_path</code> [Glib 2.32]	<code>g_build_pathv</code> [Glib 2.32]
<code>g_find_program_in_path</code> [Glib 2.32]	<code>g_get_application_name</code> [Glib 2.32]
<code>g_get_current_dir</code> [Glib 2.32]	<code>g_get_home_dir</code> [Glib 2.32]
<code>g_get_host_name</code> [Glib 2.32]	<code>g_get_prpname</code> [Glib 2.32]
<code>g_get_real_name</code> [Glib 2.32]	<code>g_get_system_config_dirs</code> [Glib 2.32]
<code>g_get_system_data_dirs</code> [Glib 2.32]	<code>g_get_tmp_dir</code> [Glib 2.32]
<code>g_get_user_cache_dir</code> [Glib 2.32]	<code>g_get_user_config_dir</code> [Glib 2.32]
<code>g_get_user_data_dir</code> [Glib 2.32]	<code>g_get_user_name</code> [Glib 2.32]
<code>g_getenv</code> [Glib 2.32]	<code>g_listenv</code> [Glib 2.32]
<code>g_nullify_pointer</code> [Glib 2.32]	<code>g_parse_debug_string</code> [Glib 2.32]
<code>g_path_get_basename</code> [Glib 2.32]	<code>g_path_get_dirname</code> [Glib 2.32]
<code>g_path_is_absolute</code> [Glib 2.32]	<code>g_path_skip_root</code> [Glib 2.32]
<code>g_qsort_with_data</code> [Glib 2.32]	<code>g_set_application_name</code> [Glib 2.32]
<code>g_set_prpname</code> [Glib 2.32]	<code>g_setenv</code> [Glib 2.32]
<code>g_spaced_primes_closest</code> [Glib 2.32]	<code>g_unsetenv</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Miscellaneous Utility Functions specified in Table 17-44, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-44 libglib-2.0 - Glib Miscellaneous Utility Functions Deprecated Function Interfaces

<code>g_atexit</code> [Glib 2.32]	

17.2.31 Glib N-ary Trees

17.2.31.1 Interfaces for Glib N-ary Trees

An LSB conforming implementation shall provide the generic functions for Glib N-ary Trees specified in Table 17-45, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-45 libglib-2.0 - Glib N-ary Trees Function Interfaces

<code>g_node_child_index</code> [Glib 2.32]	<code>g_node_child_position</code> [Glib 2.32]
<code>g_node_children_foreach</code> [Glib 2.32]	<code>g_node_copy</code> [Glib 2.32]
<code>g_node_copy_deep</code> [Glib 2.32]	<code>g_node_depth</code> [Glib 2.32]

<code>g_node_destroy</code> [Glib 2.32]	<code>g_node_find</code> [Glib 2.32]
<code>g_node_find_child</code> [Glib 2.32]	<code>g_node_first_sibling</code> [Glib 2.32]
<code>g_node_get_root</code> [Glib 2.32]	<code>g_node_insert</code> [Glib 2.32]
<code>g_node_insert_after</code> [Glib 2.32]	<code>g_node_insert_before</code> [Glib 2.32]
<code>g_node_is_ancestor</code> [Glib 2.32]	<code>g_node_last_child</code> [Glib 2.32]
<code>g_node_last_sibling</code> [Glib 2.32]	<code>g_node_max_height</code> [Glib 2.32]
<code>g_node_n_children</code> [Glib 2.32]	<code>g_node_n_nodes</code> [Glib 2.32]
<code>g_node_new</code> [Glib 2.32]	<code>g_node_nth_child</code> [Glib 2.32]
<code>g_node_pop_allocator</code> [Glib 2.32]	<code>g_node_prepend</code> [Glib 2.32]
<code>g_node_push_allocator</code> [Glib 2.32]	<code>g_node_reverse_children</code> [Glib 2.32]
<code>g_node_traverse</code> [Glib 2.32]	<code>g_node_unlink</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib N-ary Trees specified in Table 17-46, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-46 libglib-2.0 - Glib N-ary Trees Deprecated Function Interfaces

<code>g_node_pop_allocator</code> [Glib 2.32]	<code>g_node_push_allocator</code> [Glib 2.32]

17.2.32 Glib Pointer Arrays

17.2.32.1 Interfaces for Glib Pointer Arrays

An LSB conforming implementation shall provide the generic functions for Glib Pointer Arrays specified in Table 17-47, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-47 libglib-2.0 - Glib Pointer Arrays Function Interfaces

<code>g_ptr_array_add</code> [Glib 2.32]	<code>g_ptr_array_foreach</code> [Glib 2.32]
<code>g_ptr_array_free</code> [Glib 2.32]	<code>g_ptr_array_new</code> [Glib 2.32]
<code>g_ptr_array_remove</code> [Glib 2.32]	<code>g_ptr_array_remove_fast</code> [Glib 2.32]
<code>g_ptr_array_remove_index</code> [Glib 2.32]	<code>g_ptr_array_remove_index_fast</code> [Glib 2.32]
<code>g_ptr_array_remove_range</code> [Glib 2.32]	<code>g_ptr_array_set_size</code> [Glib 2.32]
<code>g_ptr_array_sized_new</code> [Glib 2.32]	<code>g_ptr_array_sort</code> [Glib 2.32]
<code>g_ptr_array_sort_with_data</code> [Glib 2.32]	

17.2.33 Glib Quarks

17.2.33.1 Interfaces for Glib Quarks

An LSB conforming implementation shall provide the generic functions for Glib Quarks specified in Table 17-48, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-48 libglib-2.0 - Glib Quarks Function Interfaces

g_intern_static_string [Glib 2.32]	g_intern_string [Glib 2.32]
g_quark_from_static_string [Glib 2.32]	g_quark_from_string [Glib 2.32]
g_quark_to_string [Glib 2.32]	g_quark_try_string [Glib 2.32]

17.2.34 Glib Random Numbers

17.2.34.1 Interfaces for Glib Random Numbers

An LSB conforming implementation shall provide the generic functions for Glib Random Numbers specified in Table 17-49, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-49 libglib-2.0 - Glib Random Numbers Function Interfaces

g_rand_copy [Glib 2.32]	g_rand_double [Glib 2.32]
g_rand_double_range [Glib 2.32]	g_rand_free [Glib 2.32]
g_rand_int [Glib 2.32]	g_rand_int_range [Glib 2.32]
g_rand_new [Glib 2.32]	g_rand_new_with_seed [Glib 2.32]
g_rand_new_with_seed_array [Glib 2.32]	g_rand_set_seed [Glib 2.32]
g_rand_set_seed_array [Glib 2.32]	g_random_double [Glib 2.32]
g_random_double_range [Glib 2.32]	g_random_int [Glib 2.32]
g_random_int_range [Glib 2.32]	g_random_set_seed [Glib 2.32]

17.2.35 Glib Relations and Tuples

17.2.35.1 Interfaces for Glib Relations and Tuples

An LSB conforming implementation shall provide the generic functions for Glib Relations and Tuples specified in Table 17-50, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-50 libglib-2.0 - Glib Relations and Tuples Function Interfaces

g_relation_count [Glib 2.32]	g_relation_delete [Glib 2.32]
g_relation_destroy [Glib 2.32]	g_relation_exists [Glib 2.32]
g_relation_index [Glib 2.32]	g_relation_insert [Glib 2.32]
g_relation_new [Glib 2.32]	g_relation_print [Glib 2.32]

g_relation_select [Glib 2.32]	g_tuples_destroy [Glib 2.32]
g_tuples_index [Glib 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Glib Relations and Tuples specified in Table 17-51, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-51 libglib-2.0 - Glib Relations and Tuples Deprecated Function Interfaces

g_relation_count [Glib 2.32]	g_relation_delete [Glib 2.32]
g_relation_destroy [Glib 2.32]	g_relation_exists [Glib 2.32]
g_relation_index [Glib 2.32]	g_relation_insert [Glib 2.32]
g_relation_new [Glib 2.32]	g_relation_print [Glib 2.32]
g_relation_select [Glib 2.32]	g_tuples_destroy [Glib 2.32]
g_tuples_index [Glib 2.32]	

17.2.36 Glib Shell-related Utilities

17.2.36.1 Interfaces for Glib Shell-related Utilities

An LSB conforming implementation shall provide the generic functions for Glib Shell-related Utilities specified in Table 17-52, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-52 libglib-2.0 - Glib Shell-related Utilities Function Interfaces

g_shell_error_quark [Glib 2.32]	g_shell_parse_argv [Glib 2.32]
g_shell_quote [Glib 2.32]	g_shell_unquote [Glib 2.32]

17.2.37 Glib Simple XML Subset Parser

17.2.37.1 Interfaces for Glib Simple XML Subset Parser

An LSB conforming implementation shall provide the generic functions for Glib Simple XML Subset Parser specified in Table 17-53, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-53 libglib-2.0 - Glib Simple XML Subset Parser Function Interfaces

g_markup_error_quark [Glib 2.32]	g_markup_escape_text [Glib 2.32]
g_markup_parse_context_end_parse [Glib 2.32]	g_markup_parse_context_free [Glib 2.32]
g_markup_parse_context_get_element [Glib 2.32]	g_markup_parse_context_get_position [Glib 2.32]
g_markup_parse_context_new [Glib 2.32]	g_markup_parse_context_parse [Glib 2.32]

<code>g_markup_printf_escaped</code> [Glib 2.32]	<code>g_markup_vprintf_escaped</code> [Glib 2.32]

17.2.38 Glib Singly-Linked Lists

17.2.38.1 Interfaces for Glib Singly-Linked Lists

An LSB conforming implementation shall provide the generic functions for Glib Singly-Linked Lists specified in Table 17-54, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-54 libglib-2.0 - Glib Singly-Linked Lists Function Interfaces

<code>g_slist_alloc</code> [Glib 2.32]	<code>g_slist_append</code> [Glib 2.32]
<code>g_slist_concat</code> [Glib 2.32]	<code>g_slist_copy</code> [Glib 2.32]
<code>g_slist_delete_link</code> [Glib 2.32]	<code>g_slist_find</code> [Glib 2.32]
<code>g_slist_find_custom</code> [Glib 2.32]	<code>g_slist_foreach</code> [Glib 2.32]
<code>g_slist_free</code> [Glib 2.32]	<code>g_slist_free_1</code> [Glib 2.32]
<code>g_slist_index</code> [Glib 2.32]	<code>g_slist_insert</code> [Glib 2.32]
<code>g_slist_insert_before</code> [Glib 2.32]	<code>g_slist_insert_sorted</code> [Glib 2.32]
<code>g_slist_insert_sorted_with_data</code> [Glib 2.32]	<code>g_slist_last</code> [Glib 2.32]
<code>g_slist_length</code> [Glib 2.32]	<code>g_slist_nth</code> [Glib 2.32]
<code>g_slist_nth_data</code> [Glib 2.32]	<code>g_slist_pop_allocator</code> [Glib 2.32]
<code>g_slist_position</code> [Glib 2.32]	<code>g_slist_prepend</code> [Glib 2.32]
<code>g_slist_push_allocator</code> [Glib 2.32]	<code>g_slist_remove</code> [Glib 2.32]
<code>g_slist_remove_all</code> [Glib 2.32]	<code>g_slist_remove_link</code> [Glib 2.32]
<code>g_slist_reverse</code> [Glib 2.32]	<code>g_slist_sort</code> [Glib 2.32]
<code>g_slist_sort_with_data</code> [Glib 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Glib Singly-Linked Lists specified in Table 17-55, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-55 libglib-2.0 - Glib Singly-Linked Lists Deprecated Function Interfaces

<code>g_slist_pop_allocator</code> [Glib 2.32]	<code>g_slist_push_allocator</code> [Glib 2.32]

17.2.39 Glib Spawning Processes

17.2.39.1 Interfaces for Glib Spawning Processes

An LSB conforming implementation shall provide the generic functions for Glib Spawning Processes specified in Table 17-56, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-56 libglib-2.0 - Glib Spawning Processes Function Interfaces

g_spawn_async [Glib 2.32]	g_spawn_async_with_pipes [Glib 2.32]
g_spawn_close_pid [Glib 2.32]	g_spawn_command_line_async [Glib 2.32]
g_spawn_command_line_sync [Glib 2.32]	g_spawn_error_quark [Glib 2.32]
g_spawn_sync [Glib 2.32]	

17.2.40 Glib String Chunks

17.2.40.1 Interfaces for Glib String Chunks

An LSB conforming implementation shall provide the generic functions for Glib String Chunks specified in Table 17-57, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-57 libglib-2.0 - Glib String Chunks Function Interfaces

g_string_chunk_free [Glib 2.32]	g_string_chunk_insert [Glib 2.32]
g_string_chunk_insert_const [Glib 2.32]	g_string_chunk_insert_len [Glib 2.32]
g_string_chunk_new [Glib 2.32]	

17.2.41 Glib String Utility Functions

17.2.41.1 Interfaces for Glib String Utility Functions

An LSB conforming implementation shall provide the generic functions for Glib String Utility Functions specified in Table 17-58, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-58 libglib-2.0 - Glib String Utility Functions Function Interfaces

g_ascii_digit_value [Glib 2.32]	g_ascii_dtostr [Glib 2.32]
g_ascii_formatd [Glib 2.32]	g_ascii_strcasecmp [Glib 2.32]
g_ascii_strdown [Glib 2.32]	g_ascii_strncasecmp [Glib 2.32]
g_ascii_strtod [Glib 2.32]	g_ascii_strtoll [Glib 2.32]
g_ascii_strtoull [Glib 2.32]	g_ascii_strup [Glib 2.32]
g_ascii_tolower [Glib 2.32]	g_ascii_toupper [Glib 2.32]
g_ascii_xdigit_value [Glib 2.32]	g_fprintf [Glib 2.32]

g_printf [Glib 2.32]	g_printf_string_upper_bound [Glib 2.32]
g_snprintf [Glib 2.32]	g_sprintf [Glib 2.32]
g_stpcpy [Glib 2.32]	g_str_has_prefix [Glib 2.32]
g_str_has_suffix [Glib 2.32]	g_strcanon [Glib 2.32]
g_strchomp [Glib 2.32]	g_strchug [Glib 2.32]
g_strcompress [Glib 2.32]	g_strconcat [Glib 2.32]
g_strdelimit [Glib 2.32]	g_strdup [Glib 2.32]
g_strdup_printf [Glib 2.32]	g_strdup_vprintf [Glib 2.32]
g_strdupv [Glib 2.32]	g_strerror [Glib 2.32]
g_strescape [Glib 2.32]	g_strfreev [Glib 2.32]
g_string_ascii_down [Glib 2.32]	g_string_ascii_up [Glib 2.32]
g_strjoin [Glib 2.32]	g_strjoinv [Glib 2.32]
g_strlcat [Glib 2.32]	g_strlcpy [Glib 2.32]
g_strndup [Glib 2.32]	g_strnfill [Glib 2.32]
g_strreverse [Glib 2.32]	g_strrstr [Glib 2.32]
g_strrstr_len [Glib 2.32]	g_strsignal [Glib 2.32]
g_strsplit [Glib 2.32]	g_strsplit_set [Glib 2.32]
g_strstr_len [Glib 2.32]	g_strtod [Glib 2.32]
g_strv_length [Glib 2.32]	g_vasprintf [Glib 2.32]
g_vfprintf [Glib 2.32]	g_vprintf [Glib 2.32]
g_vsnprintf [Glib 2.32]	g_vsprintf [Glib 2.32]

17.2.42 Glib Strings

17.2.42.1 Interfaces for Glib Strings

An LSB conforming implementation shall provide the generic functions for Glib Strings specified in Table 17-59, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-59 libglib-2.0 - Glib Strings Function Interfaces

g_string_append [Glib 2.32]	g_string_append_c [Glib 2.32]
g_string_append_len [Glib 2.32]	g_string_append_printf [Glib 2.32]
g_string_append_unichar [Glib 2.32]	g_string_assign [Glib 2.32]
g_string_equal [Glib 2.32]	g_string_erase [Glib 2.32]
g_string_free [Glib 2.32]	g_string_hash [Glib 2.32]
g_string_insert [Glib 2.32]	g_string_insert_c [Glib 2.32]
g_string_insert_len [Glib 2.32]	g_string_insert_unichar [Glib 2.32]

<code>g_string_new</code> [Glib 2.32]	<code>g_string_new_len</code> [Glib 2.32]
<code>g_string_prepend</code> [Glib 2.32]	<code>g_string_prepend_c</code> [Glib 2.32]
<code>g_string_prepend_len</code> [Glib 2.32]	<code>g_string_prepend_unichar</code> [Glib 2.32]
<code>g_string_printf</code> [Glib 2.32]	<code>g_string_set_size</code> [Glib 2.32]
<code>g_string_sized_new</code> [Glib 2.32]	<code>g_string_truncate</code> [Glib 2.32]

17.2.43 Glib - The Main Event Loop

17.2.43.1 Interfaces for Glib - The Main Event Loop

An LSB conforming implementation shall provide the generic functions for Glib - The Main Event Loop specified in Table 17-60, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-60 libglib-2.0 - Glib - The Main Event Loop Function Interfaces

<code>g_child_watch_add</code> [Glib 2.32]	<code>g_child_watch_add_full</code> [Glib 2.32]
<code>g_child_watch_source_new</code> [Glib 2.32]	<code>g_idle_add</code> [Glib 2.32]
<code>g_idle_add_full</code> [Glib 2.32]	<code>g_idle_remove_by_data</code> [Glib 2.32]
<code>g_idle_source_new</code> [Glib 2.32]	<code>g_main_context_acquire</code> [Glib 2.32]
<code>g_main_context_add_poll</code> [Glib 2.32]	<code>g_main_context_check</code> [Glib 2.32]
<code>g_main_context_default</code> [Glib 2.32]	<code>g_main_context_dispatch</code> [Glib 2.32]
<code>g_main_context_find_source_by_funcs_user_data</code> [Glib 2.32]	<code>g_main_context_find_source_by_id</code> [Glib 2.32]
<code>g_main_context_find_source_by_user_data</code> [Glib 2.32]	<code>g_main_context_get_poll_func</code> [Glib 2.32]
<code>g_main_context_is_owner</code> [Glib 2.32]	<code>g_main_context_iteration</code> [Glib 2.32]
<code>g_main_context_new</code> [Glib 2.32]	<code>g_main_context_pending</code> [Glib 2.32]
<code>g_main_context_prepare</code> [Glib 2.32]	<code>g_main_context_query</code> [Glib 2.32]
<code>g_main_context_ref</code> [Glib 2.32]	<code>g_main_context_release</code> [Glib 2.32]
<code>g_main_context_remove_poll</code> [Glib 2.32]	<code>g_main_context_set_poll_func</code> [Glib 2.32]
<code>g_main_context_unref</code> [Glib 2.32]	<code>g_main_context_wait</code> [Glib 2.32]
<code>g_main_context_wakeup</code> [Glib 2.32]	<code>g_main_current_source</code> [Glib 2.32]
<code>g_main_depth</code> [Glib 2.32]	<code>g_main_loop_get_context</code> [Glib 2.32]
<code>g_main_loop_is_running</code> [Glib 2.32]	<code>g_main_loop_new</code> [Glib 2.32]
<code>g_main_loop_quit</code> [Glib 2.32]	<code>g_main_loop_ref</code> [Glib 2.32]
<code>g_main_loop_run</code> [Glib 2.32]	<code>g_main_loop_unref</code> [Glib 2.32]
<code>g_source_add_poll</code> [Glib 2.32]	<code>g_source_attach</code> [Glib 2.32]
<code>g_source_destroy</code> [Glib 2.32]	<code>g_source_get_can_recurse</code> [Glib 2.32]

<code>g_source_get_context</code> [Glib 2.32]	<code>g_source_get_current_time</code> [Glib 2.32]
<code>g_source_get_id</code> [Glib 2.32]	<code>g_source_get_priority</code> [Glib 2.32]
<code>g_source_is_destroyed</code> [Glib 2.32]	<code>g_source_new</code> [Glib 2.32]
<code>g_source_ref</code> [Glib 2.32]	<code>g_source_remove</code> [Glib 2.32]
<code>g_source_remove_by_funcs_userdata</code> [Glib 2.32]	<code>g_source_remove_by_user_data</code> [Glib 2.32]
<code>g_source_remove_poll</code> [Glib 2.32]	<code>g_source_set_callback</code> [Glib 2.32]
<code>g_source_set_callback_indirect</code> [Glib 2.32]	<code>g_source_set_can_recurse</code> [Glib 2.32]
<code>g_source_set_funcs</code> [Glib 2.32]	<code>g_source_set_priority</code> [Glib 2.32]
<code>g_source_unref</code> [Glib 2.32]	<code>g_timeout_add</code> [Glib 2.32]
<code>g_timeout_add_full</code> [Glib 2.32]	<code>g_timeout_source_new</code> [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib - The Main Event Loop specified in Table 17-61, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-61 libglib-2.0 - Glib - The Main Event Loop Deprecated Function Interfaces

<code>g_source_get_current_time</code> [Glib 2.32]	

17.2.44 Glib Thread Pools

17.2.44.1 Interfaces for Glib Thread Pools

An LSB conforming implementation shall provide the generic functions for Glib Thread Pools specified in Table 17-62, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-62 libglib-2.0 - Glib Thread Pools Function Interfaces

<code>g_thread_pool_free</code> [Glib 2.32]	<code>g_thread_pool_get_max_idle_time</code> [Glib 2.32]
<code>g_thread_pool_get_max_threads</code> [Glib 2.32]	<code>g_thread_pool_get_max_unused_threads</code> [Glib 2.32]
<code>g_thread_pool_get_num_threads</code> [Glib 2.32]	<code>g_thread_pool_get_num_unused_threads</code> [Glib 2.32]
<code>g_thread_pool_new</code> [Glib 2.32]	<code>g_thread_pool_push</code> [Glib 2.32]
<code>g_thread_pool_set_max_idle_time</code> [Glib 2.32]	<code>g_thread_pool_set_max_threads</code> [Glib 2.32]

g_thread_pool_set_max_unused_threads [Glib 2.32]	g_thread_pool_set_sort_function [Glib 2.32]
g_thread_pool_stop_unused_threads [Glib 2.32]	g_thread_pool_unprocessed [Glib 2.32]

17.2.45 Glib Threads

17.2.45.1 Interfaces for Glib Threads

An LSB conforming implementation shall provide the generic functions for Glib Threads specified in Table 17-63, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-63 libglib-2.0 - Glib Threads Function Interfaces

g_static_mutex_free [Glib 2.32]	g_static_mutex_init [Glib 2.32]
g_static_private_free [Glib 2.32]	g_static_private_get [Glib 2.32]
g_static_private_init [Glib 2.32]	g_static_private_set [Glib 2.32]
g_static_rec_mutex_free [Glib 2.32]	g_static_rec_mutex_init [Glib 2.32]
g_static_rec_mutex_lock [Glib 2.32]	g_static_rec_mutex_lock_full [Glib 2.32]
g_static_rec_mutex_trylock [Glib 2.32]	g_static_rec_mutex_unlock [Glib 2.32]
g_static_rec_mutex_unlock_full [Glib 2.32]	g_static_rw_lock_free [Glib 2.32]
g_static_rw_lock_init [Glib 2.32]	g_static_rw_lock_reader_lock [Glib 2.32]
g_static_rw_lock_reader_trylock [Glib 2.32]	g_static_rw_lock_reader_unlock [Glib 2.32]
g_static_rw_lock_writer_lock [Glib 2.32]	g_static_rw_lock_writer_trylock [Glib 2.32]
g_static_rw_lock_writer_unlock [Glib 2.32]	g_thread_create_full [Glib 2.32]
g_thread_error_quark [Glib 2.32]	g_thread_exit [Glib 2.32]
g_thread_foreach [Glib 2.32]	g_thread_join [Glib 2.32]
g_thread_self [Glib 2.32]	g_thread_set_priority [Glib 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Glib Threads specified in Table 17-64, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-64 libglib-2.0 - Glib Threads Deprecated Function Interfaces

g_static_mutex_free [Glib 2.32]	g_static_mutex_init [Glib 2.32]

<code>g_static_rec_mutex_free</code> [Glib 2.32]	<code>g_static_rec_mutex_init</code> [Glib 2.32]
<code>g_static_rec_mutex_lock</code> [Glib 2.32]	<code>g_static_rec_mutex_lock_full</code> [Glib 2.32]
<code>g_static_rec_mutex_trylock</code> [Glib 2.32]	<code>g_static_rec_mutex_unlock</code> [Glib 2.32]
<code>g_static_rec_mutex_unlock_full</code> [Glib 2.32]	<code>g_static_rw_lock_free</code> [Glib 2.32]
<code>g_static_rw_lock_init</code> [Glib 2.32]	<code>g_static_rw_lock_reader_lock</code> [Glib 2.32]
<code>g_thread_create_full</code> [Glib 2.32]	<code>g_thread_foreach</code> [Glib 2.32]
<code>g_thread_set_priority</code> [Glib 2.32]	

17.2.46 Glib Timers

17.2.46.1 Interfaces for Glib Timers

An LSB conforming implementation shall provide the generic functions for Glib Timers specified in Table 17-65, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-65 libglib-2.0 - Glib Timers Function Interfaces

<code>g_timer_continue</code> [Glib 2.32]	<code>g_timer_destroy</code> [Glib 2.32]
<code>g_timer_elapsed</code> [Glib 2.32]	<code>g_timer_new</code> [Glib 2.32]
<code>g_timer_reset</code> [Glib 2.32]	<code>g_timer_start</code> [Glib 2.32]
<code>g_timer_stop</code> [Glib 2.32]	

17.2.47 Glib Trash Stacks

17.2.47.1 Interfaces for Glib Trash Stacks

An LSB conforming implementation shall provide the generic functions for Glib Trash Stacks specified in Table 17-66, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-66 libglib-2.0 - Glib Trash Stacks Function Interfaces

<code>g_trash_stack_height</code> [Glib 2.32]	<code>g_trash_stack_peek</code> [Glib 2.32]
<code>g_trash_stack_pop</code> [Glib 2.32]	<code>g_trash_stack_push</code> [Glib 2.32]

17.2.48 Glib Unicode Manipulation

17.2.48.1 Interfaces for Glib Unicode Manipulation

An LSB conforming implementation shall provide the generic functions for Glib Unicode Manipulation specified in Table 17-67, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-67 libglib-2.0 - Glib Unicode Manipulation Function Interfaces

g_mkdir_with_parents [Glib 2.32]	g_ucs4_to_utf16 [Glib 2.32]
g_ucs4_to_utf8 [Glib 2.32]	g_unichar_break_type [Glib 2.32]
g_unichar_digit_value [Glib 2.32]	g_unichar_get_mirror_char [Glib 2.32]
g_unichar_isalnum [Glib 2.32]	g_unichar_isalpha [Glib 2.32]
g_unichar_iscntrl [Glib 2.32]	g_unichar_isdefined [Glib 2.32]
g_unichar_isdigit [Glib 2.32]	g_unichar_isgraph [Glib 2.32]
g_unichar_islower [Glib 2.32]	g_unichar_isprint [Glib 2.32]
g_unichar_isspace [Glib 2.32]	g_unichar_isspace [Glib 2.32]
g_unichar_istitle [Glib 2.32]	g_unichar_isupper [Glib 2.32]
g_unichar_iswide [Glib 2.32]	g_unichar_iswide_cjk [Glib 2.32]
g_unichar_isxdigit [Glib 2.32]	g_unichar_to_utf8 [Glib 2.32]
g_unichar_tolower [Glib 2.32]	g_unichar_totitle [Glib 2.32]
g_unichar_toupper [Glib 2.32]	g_unichar_type [Glib 2.32]
g_unichar_validate [Glib 2.32]	g_unichar_xdigit_value [Glib 2.32]
g_unicode_canonical_decomposition [Glib 2.32]	g_unicode_canonical_ordering [Glib 2.32]
g_utf16_to_ucs4 [Glib 2.32]	g_utf16_to_utf8 [Glib 2.32]
g_utf8_casefold [Glib 2.32]	g_utf8_collate [Glib 2.32]
g_utf8_collate_key [Glib 2.32]	g_utf8_collate_key_for_filename [Glib 2.32]
g_utf8_find_next_char [Glib 2.32]	g_utf8_find_prev_char [Glib 2.32]
g_utf8_get_char [Glib 2.32]	g_utf8_get_char_validated [Glib 2.32]
g_utf8_normalize [Glib 2.32]	g_utf8_offset_to_pointer [Glib 2.32]
g_utf8_pointer_to_offset [Glib 2.32]	g_utf8_prev_char [Glib 2.32]
g_utf8_strchr [Glib 2.32]	g_utf8_strdown [Glib 2.32]
g_utf8_strlen [Glib 2.32]	g_utf8_strncpy [Glib 2.32]
g_utf8_strrchr [Glib 2.32]	g_utf8_strreverse [Glib 2.32]
g_utf8_strup [Glib 2.32]	g_utf8_to_ucs4 [Glib 2.32]
g_utf8_to_ucs4_fast [Glib 2.32]	g_utf8_to_utf16 [Glib 2.32]
g_utf8_validate [Glib 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Glib Unicode Manipulation specified in Table 17-68, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-68 libglib-2.0 - Glib Unicode Manipulation Deprecated Function Interfaces

g_unicode_canonical_decomposition [Glib 2.32]	

17.2.49 Glib Version Information

17.2.49.1 Interfaces for Glib Version Information

An LSB conforming implementation shall provide the generic functions for Glib Version Information specified in Table 17-69, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-69 libglib-2.0 - Glib Version Information Function Interfaces

glib_check_version [Glib 2.32]	

An LSB conforming implementation shall provide the generic data interfaces for Glib Version Information specified in Table 17-70, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-70 libglib-2.0 - Glib Version Information Data Interfaces

glib_binary_age [Glib 2.32]	glib_interface_age [Glib 2.32]
glib_major_version [Glib 2.32]	glib_mem_profiler_table [Glib 2.32]
glib_micro_version [Glib 2.32]	glib_minor_version [Glib 2.32]

17.2.50 Glib Bookmark File Parser

17.2.50.1 Interfaces for Glib Bookmark File Parser

An LSB conforming implementation shall provide the generic functions for Glib Bookmark File Parser specified in Table 17-71, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-71 libglib-2.0 - Glib Bookmark File Parser Function Interfaces

g_bookmark_file_add_application [Glib 2.32]	g_bookmark_file_add_group [Glib 2.32]
g_bookmark_file_error_quark [Glib 2.32]	g_bookmark_file_free [Glib 2.32]
g_bookmark_file_get_added [Glib 2.32]	g_bookmark_file_get_app_info [Glib 2.32]
g_bookmark_file_get_applications [Glib 2.32]	g_bookmark_file_get_description [Glib 2.32]
g_bookmark_file_get_groups [Glib 2.32]	g_bookmark_file_get_icon [Glib 2.32]
g_bookmark_file_get_is_private [Glib 2.32]	g_bookmark_file_get_mime_type [Glib 2.32]
g_bookmark_file_get_modified [Glib 2.32]	g_bookmark_file_get_size [Glib 2.32]

g_bookmark_file_get_title [Glib 2.32]	g_bookmark_file_get_uris [Glib 2.32]
g_bookmark_file_get_visited [Glib 2.32]	g_bookmark_file_has_application [Glib 2.32]
g_bookmark_file_has_group [Glib 2.32]	g_bookmark_file_has_item [Glib 2.32]
g_bookmark_file_load_from_data [Glib 2.32]	g_bookmark_file_load_from_data_dirs [Glib 2.32]
g_bookmark_file_load_from_file [Glib 2.32]	g_bookmark_file_move_item [Glib 2.32]
g_bookmark_file_new [Glib 2.32]	g_bookmark_file_remove_application [Glib 2.32]
g_bookmark_file_remove_group [Glib 2.32]	g_bookmark_file_remove_item [Glib 2.32]
g_bookmark_file_set_added [Glib 2.32]	g_bookmark_file_set_app_info [Glib 2.32]
g_bookmark_file_set_description [Glib 2.32]	g_bookmark_file_set_groups [Glib 2.32]
g_bookmark_file_set_icon [Glib 2.32]	g_bookmark_file_set_is_private [Glib 2.32]
g_bookmark_file_set_mime_type [Glib 2.32]	g_bookmark_file_set_modified [Glib 2.32]
g_bookmark_file_set_title [Glib 2.32]	g_bookmark_file_set_visited [Glib 2.32]
g_bookmark_file_to_data [Glib 2.32]	g_bookmark_file_to_file [Glib 2.32]

17.2.51 Glib Memory Slices

17.2.51.1 Interfaces for Glib Memory Slices

An LSB conforming implementation shall provide the generic functions for Glib Memory Slices specified in Table 17-72, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-72 libglib-2.0 - Glib Memory Slices Function Interfaces

g_slice_alloc [Glib 2.32]	g_slice_alloc0 [Glib 2.32]
g_slice_copy [Glib 2.32]	g_slice_free1 [Glib 2.32]
g_slice_free_chain_with_offset [Glib 2.32]	

17.2.52 Glib Base64 Encoding

17.2.52.1 Interfaces for Glib Base64 Encoding

An LSB conforming implementation shall provide the generic functions for Glib Base64 Encoding specified in Table 17-73, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-73 libglib-2.0 - Glib Base64 Encoding Function Interfaces

<code>g_base64_decode</code> [Glib 2.32]	<code>g_base64_decode_inplace</code> [Glib 2.32]
<code>g_base64_decode_step</code> [Glib 2.32]	<code>g_base64_encode</code> [Glib 2.32]
<code>g_base64_encode_close</code> [Glib 2.32]	<code>g_base64_encode_step</code> [Glib 2.32]

17.3 Data Definitions for libglib-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libglib-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.3.1 glib-2.0/glib.h

```
#define GLIB_HAVE_ALLOCA_H
#define GLIB_HAVE_SYS_POLL_H
#ifdef __cplusplus
# define G_BEGIN_DECLS extern "C" {
#else
# define G_BEGIN_DECLS
#endif
#ifdef __cplusplus
# define G_END_DECLS }
#else
# define G_END_DECLS
#endif
#define G_GINT32_MODIFIER ""
#define G_GNUC_FUNCTION "" /* DEPRECATED */
#define G_GNUC_PRETTY_FUNCTION "" /* DEPRECATED */
#define G_OPTION_REMAINING ""
#define G_OS_UNIX
#define G_THREADS_ENABLED
#define G_THREADS_IMPL_POSIX /* DEPRECATED */
#define G_WIN32_DLLMAIN_FOR_DLL_NAME(static,dll_name)
#define G_CSET_LATINC \
    "\300\301\302\303\304\305\306" \
    "\307\310\311\312\313\314\315\316\317\320"
"\321\322\323\324\325\326" \
    "\330\331\332\333\334\335\336"
#define G_CSET_LATINS \
    "\337\340\341\342\343\344\345\346" \
    "\347\350\351\352\353\354\355\356\357\360"
"\361\362\363\364\365\366" \
    "\370\371\372\373\374\375\376\377"
#define g_mem_chunk_create(type,pre_alloc,alloc_type) \
    ( g_mem_chunk_new (#type " mem chunks (" #pre_alloc ")",
sizeof \
```

```

        (type), sizeof (type) * (pre_alloc), (alloc_type)) )
#define G_NODE_IS_ROOT(node) \
    (((GNode*) (node))->parent == NULL && ((GNode*) (node))->prev == NULL \
    && ((GNode*) (node))->next == NULL)
#define g_once(once, func, arg) \
    (((once)->status == G_ONCE_STATUS_READY) ? (once)->retval : \
    \
    g_once_impl ((once), (func), (arg)))
#define CLAMP(x, low, high) \
    (((x) > (high)) ? (high) : (((x) < (low)) ? (low) : (x)))
#define G_STRUCT_OFFSET(struct_type, member) \
    ((glong) ((guint8*) &((struct_type*) 0)->member))
#define G_STRUCT_MEMBER_P(struct_p, struct_offset) \
    ((gpointer) ((guint8*) (struct_p) + (glong) (struct_offset)))
#define GUIN16_SWAP_LE_BE_CONSTANT(val) \
    (((guint16) ((guint16) (val) >> 8) | ((guint16) \
    ((guint16) (val) << 8)))
#define GUIN32_SWAP_LE_BE_CONSTANT(val) \
    (((guint32) ((guint32) (val) & (guint32) 0x000000ffU) << \
    24) | \
    (((guint32) (val) & (guint32) 0x0000ff00U) << 8) | (((guint32) \
    (val) & \
    (guint32) 0x00ff0000U) >> 8) | (((guint32) (val) & (guint32) \
    \
    0xff000000U) >> 24)))
#define GUIN32_SWAP_LE_PDP(val) \
    (((guint32) ((guint32) (val) & (guint32) 0x0000ffffU) << \
    16) | \
    (((guint32) (val) & (guint32) 0xffff0000U) >> 16)))
#define GUIN32_SWAP_BE_PDP(val) \
    (((guint32) ((guint32) (val) & (guint32) 0x00ff00ffU) << \
    8) | \
    (((guint32) (val) & (guint32) 0xff00ff00U) >> 8)))
#define GUIN64_SWAP_LE_BE_CONSTANT(val) \
    (((guint64) ((guint64) (val) & (guint64) G_GINT64_CONSTANT \
    \
    (0x0000000000000000ffU)) << 56) | (((guint64) (val) & (guint64) \
    \
    G_GINT64_CONSTANT (0x00000000000000ff00U)) << 40) | (((guint64) \
    (val) & \
    (guint64) G_GINT64_CONSTANT (0x0000000000ff0000U)) << 24) | \
    \
    (((guint64) (val) & (guint64) G_GINT64_CONSTANT \
    (0x00000000ff00000000U)) >> 8) | (((guint64) (val) & (guint64) \
    \
    G_GINT64_CONSTANT (0x0000ff000000000000U)) >> 24) | (((guint64) \
    (val) & \
    (guint64) G_GINT64_CONSTANT (0x00ff00000000000000U)) >> 40) | \
    \
    (((guint64) (val) & (guint64) G_GINT64_CONSTANT \
    (0xff0000000000000000U)) \
    >> 56)))
#define g_ascii_isalnum(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_ALNUM) != 0)
#define g_ascii_isalpha(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_ALPHA) != 0)
#define g_ascii_iscntrl(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_CNTRL) != 0)
#define g_ascii_isdigit(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_DIGIT) != 0)
#define g_ascii_isgraph(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_GRAPH) != 0)

```

```

#define g_ascii_islower(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_LOWER) != 0)
#define g_ascii_isprint(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_PRINT) != 0)
#define g_ascii ispunct(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_PUNCT) != 0)
#define g_ascii_isspace(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_SPACE) != 0)
#define g_ascii_isupper(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_UPPER) != 0)
#define g_ascii_isxdigit(c) \
    ((g_ascii_table[(guchar) (c)] & G_ASCII_XDIGIT) != 0)
#define G_HOOK_ACTIVE(hook) \
    ((G_HOOK_FLAGS (hook) & G_HOOK_FLAG_ACTIVE) != 0)
#define G_HOOK_IN_CALL(hook) \
    ((G_HOOK_FLAGS (hook) & G_HOOK_FLAG_IN_CALL) != 0)
#define g_node_first_child(node) \
    ((node) ? ((GNode*) (node))>children : NULL)
#define g_node_next_sibling(node) \
    ((node) ? ((GNode*) (node))>next : NULL)
#define g_node_prev_sibling(node) \
    ((node) ? ((GNode*) (node))>prev : NULL)
#define g_new(struct_type,n_structs) \
    ((struct_type *) g_malloc (((gsize) sizeof (struct_type)) *
    ((gsize) \
    (n_structs))))
#define g_new0(struct_type,n_structs) \
    ((struct_type *) g_malloc0 (((gsize) sizeof (struct_type))
    * ((gsize) \
    (n_structs))))
#define g_renew(struct_type,mem,n_structs) \
    ((struct_type *) g_realloc ((mem), ((gsize) sizeof
    (struct_type)) * \
    ((gsize) (n_structs))))
#define g_newa(struct_type,n_structs) \
    ((struct_type*) g_alloca (sizeof (struct_type) * (gsize) \
    (n_structs)))
#define G_STRUCT_MEMBER(member_type,struct_p,struct_offset) \
    (*(member_type*) G_STRUCT_MEMBER_P ((struct_p),
    (struct_offset)))
#define G_THREAD_UP(op,arglist) \
    (*g_thread_functions_for_glib_use . op) arglist
#define GLIB_CHECK_VERSION(major,minor,micro) \
    (GLIB_MAJOR_VERSION > (major) || (GLIB_MAJOR_VERSION ==
    (major) &&
    GLIB_MINOR_VERSION > (minor)) || (GLIB_MAJOR_VERSION ==
    (major) && \
    GLIB_MINOR_VERSION == (minor) && GLIB_MICRO_VERSION >=
    (micro)))
#define g_atomic_int_dec_and_test(atomic) \
    (g_atomic_int_exchange_and_add ((atomic), -1) == 1)
#define g_static_mutex_get_mutex_impl_shortcut(mutex) \
    (g_atomic_pointer_get ((gpointer*)mutex) ? *(mutex) : \
    g_static_mutex_get_mutex_impl (mutex))
#define g_datalist_get_data(dl,k) \
    (g_datalist_id_get_data ((dl), g_quark_try_string (k)))
#define g_dataset_get_data(l,k) \
    (g_dataset_id_get_data ((l), g_quark_try_string (k)))
#define G_HOOK_IS_VALID(hook) \
    (G_HOOK (hook)>hook_id != 0 && (G_HOOK_FLAGS (hook) & \
    G_HOOK_FLAG_ACTIVE))
#define G_HOOK_IS_UNLINKED(hook) \
    (G_HOOK (hook)>next == NULL && G_HOOK (hook)>prev == NULL
    && G_HOOK \
    (hook)>hook_id == 0 && G_HOOK (hook)>ref_count == 0)
#define g_thread_create(func,data,joinable,error) \

```

```

        (g_thread_create_full (func, data, 0, joinable, FALSE, \
        G_THREAD_PRIORITY_NORMAL, error))
#define G_THREAD_ECF(op,fail,mutex,type) \
        (g_thread_supported () ? ((type*)(GMutex*, gulong, gchar*)) \
        \
        (*g_thread_functions_for_glib_use . op)) (mutex,
G_MUTEX_DEBUG_MAGIC, \
        G_STRLOC) : (fail))
#define G_THREAD_CF(op,fail,arg) \
        (g_thread_supported () ? G_THREAD_UF (op, arg) : (fail))
#define G_LOCK_DEFINE(name) \
        GStaticMutex G_LOCK_NAME (name) = G_STATIC_MUTEX_INIT
#define g_datalist_remove_no_notify(dl,k) \
        g_datalist_id_remove_no_notify ((dl), g_quark_try_string
(k))
#define g_datalist_id_remove_data(dl,q) \
        g_datalist_id_set_data ((dl), (q), NULL)
#define g_datalist_remove_data(dl,k) \
        g_datalist_id_set_data ((dl), g_quark_try_string (k), NULL)
#define g_datalist_id_set_data(dl,q,d) \
        g_datalist_id_set_data_full ((dl), (q), (d), NULL)
#define g_datalist_set_data_full(dl,k,d,f) \
        g_datalist_id_set_data_full ((dl), g_quark_from_string (k),
(d), (f))
#define g_datalist_set_data(dl,k,d) \
        g_datalist_set_data_full ((dl), (k), (d), NULL)
#define g_dataset_remove_no_notify(l,k) \
        g_dataset_id_remove_no_notify ((l), g_quark_try_string (k))
#define g_dataset_id_remove_data(l,k) \
        g_dataset_id_set_data ((l), (k), NULL)
#define g_dataset_remove_data(l,k) \
        g_dataset_id_set_data ((l), g_quark_try_string (k), NULL)
#define g_dataset_id_set_data(l,k,d) \
        g_dataset_id_set_data_full ((l), (k), (d), NULL)
#define g_dataset_set_data_full(l,k,d,f) \
        g_dataset_id_set_data_full ((l), g_quark_from_string (k),
(d), (f))
#define g_dataset_set_data(l,k,d) \
        g_dataset_set_data_full ((l), (k), (d), NULL)
#define g_hook_append(hook_list,hook) \
        g_hook_insert_before ((hook_list), NULL, (hook))
#define g_critical(...) \
        g_log (G_LOG_DOMAIN, G_LOG_LEVEL_CRITICAL, __VA_ARGS__)
#define g_message(...) \
        g_log (G_LOG_DOMAIN, G_LOG_LEVEL_MESSAGE, __VA_ARGS__)
#define g_warning(...) \
        g_log (G_LOG_DOMAIN, G_LOG_LEVEL_WARNING, __VA_ARGS__)
#define g_static_mutex_lock(mutex) \
        g_mutex_lock (g_static_mutex_get_mutex (mutex))
#define g_static_mutex_trylock(mutex) \
        g_mutex_trylock (g_static_mutex_get_mutex (mutex))
#define g_static_mutex_unlock(mutex) \
        g_mutex_unlock (g_static_mutex_get_mutex (mutex))
#define g_node_insert_data(parent,position,data) \
        g_node_insert ((parent), (position), g_node_new (data))
#define g_node_insert_data_before(parent,sibling,data) \
        g_node_insert_before ((parent), (sibling), g_node_new (data))
#define g_node_append(parent,node) \
        g_node_insert_before ((parent), NULL, (node))
#define g_node_append_data(parent,data) \
        g_node_insert_before ((parent), NULL, g_node_new (data))
#define g_node_prepend_data(parent,data) \
        g_node_prepend ((parent), g_node_new (data))
#define g_chunk_free(mem,mem_chunk) \
        G_STMT_START { g_mem_chunk_free ((mem_chunk), (mem)); }
G_STMT_END

```

```

#define g_memmove(d,s,n) \
    G_STMT_START { memmove ((d), (s), (n)); } G_STMT_END
#define g_assert_not_reached() \
    G_STMT_START{ g_assert_warning (G_LOG_DOMAIN, __FILE__, \
    __LINE__, \
    __PRETTY_FUNCTION__, NULL); }G_STMT_END
#define g_return_val_if_reached(val) \
    G_STMT_START{ g_log (G_LOG_DOMAIN, G_LOG_LEVEL_CRITICAL, \
    "file %s: \
    line %d (%s): should not be reached", __FILE__, __LINE__, \
    __PRETTY_FUNCTION__); return (val); }G_STMT_END
#define g_return_if_reached() \
    G_STMT_START{ g_log (G_LOG_DOMAIN, G_LOG_LEVEL_CRITICAL, \
    "file %s: \
    line %d (%s): should not be reached", __FILE__, __LINE__, \
    __PRETTY_FUNCTION__); return; }G_STMT_END
#define g_assert(expr) \
    G_STMT_START{ if G_LIKELY(expr) { } else g_assert_warning \
    (G_LOG_DOMAIN, __FILE__, __LINE__, __PRETTY_FUNCTION__, \
    #expr); \
    }G_STMT_END
#define g_return_val_if_fail(expr,val) \
    G_STMT_START{ if G_LIKELY(expr) { } else \
    { g_return_if_fail_warning \
    (G_LOG_DOMAIN, __PRETTY_FUNCTION__, #expr); return (val); }; \
    }G_STMT_END
#define g_return_if_fail(expr) \
    G_STMT_START{ if G_LIKELY(expr) { } else \
    { g_return_if_fail_warning \
    (G_LOG_DOMAIN, __PRETTY_FUNCTION__, #expr); \
    return; }; }G_STMT_END
#define g_cond_broadcast(cond) \
    G_THREAD_CF (cond_broadcast, (void)0, (cond))
#define g_cond_timed_wait(cond,mutex,abs_time) \
    G_THREAD_CF (cond_timed_wait, TRUE, (cond, mutex, abs_time))
#define g_cond_wait(cond,mutex) \
    G_THREAD_CF (cond_wait, (void)0, (cond, mutex))
#define g_private_get(private_key) \
    G_THREAD_CF (private_get, ((gpointer)private_key), \
    (private_key))
#define g_private_set(private_key,value) \
    G_THREAD_CF (private_set, (void) (private_key = (GPrivate*) \
    (value)), \
    (private_key, value))
#define G_GNUC_PRINTF(format_idx,arg_idx) \
    __attribute__((__format__ (__printf__, format_idx, \
    arg_idx)))
#define G_GNUC_SCANF(format_idx,arg_idx) \
    __attribute__((__format__ (__scanf__, format_idx, arg_idx)))
#define G_STATIC_RW_LOCK_INIT \
    { G_STATIC_MUTEX_INIT, NULL, 0, FALSE, 0, 0 }
#define G_STRINGIFY_ARG(contents) #contents
#define G_DIR_SEPARATOR '/'
#define G_SEARCHPATH_SEPARATOR ':'
#define g_chunk_new(type,chunk) ( (type *) g_mem_chunk_alloc \
    (chunk) )
#define g_chunk_new0(type,chunk) ( (type *) \
    g_mem_chunk_alloc0 (chunk) )
#define MIN(a,b) ((a) < (b)) ? (a) : (b)
#define ABS(a) (((a) < 0) ? -(a) : (a))
#define MAX(a,b) ((a) > (b)) ? (a) : (b)
#define G_NODE_IS_LEAF(node) (((GNode*) (node))>children == \
    NULL)
#define g_array_index(a,t,i) (((t*) (void *) (a)>data) [(i)])
#define g_ptr_array_index(array,index_) ((array)>pdata)[index_]

```

```

#define G_IS_DIR_SEPARATOR(c) ((c) == G_DIR_SEPARATOR)
#define G_STRFUNC ((const char*) (__PRETTY_FUNCTION__))
#define G_LOG_DOMAIN ((gchar*) 0)
#define G_HOOK(hook) ((GHook*) (hook))
#define GINT_TO_BE(val) ((gint) GINT32_TO_BE (val))
#define GINT_TO_LE(val) ((gint) GINT32_TO_LE (val))
#define GINT16_TO_LE(val) ((gint16) (val))
#define G_MAXINT16 ((gint16) 0x7fff)
#define G_MININT16 ((gint16) 0x8000)
#define GINT16_TO_BE(val) ((gint16) GUINT16_SWAP_LE_BE (val))
#define GINT32_TO_LE(val) ((gint32) (val))
#define G_MAXINT32 ((gint32) 0x7fffffff)
#define G_MININT32 ((gint32) 0x80000000)
#define GINT32_TO_BE(val) ((gint32) GUINT32_SWAP_LE_BE (val))
#define GINT64_TO_LE(val) ((gint64) (val))
#define GINT64_TO_BE(val) ((gint64) GUINT64_SWAP_LE_BE (val))
#define G_MAXINT8 ((gint8) 0x7f)
#define G_MININT8 ((gint8) 0x80)
#define GSIZE_TO_POINTER(s) ((gpointer) (gsize) (s))
#define GPOINTER_TO_SIZE(p) ((gsize) (p))
#define GUINT_TO_BE(val) ((guint) GUINT32_TO_BE (val))
#define GUINT_TO_LE(val) ((guint) GUINT32_TO_LE (val))
#define GUINT16_SWAP_LE_PDP(val) ((guint16) (val))
#define GUINT16_TO_LE(val) ((guint16) (val))
#define G_MAXUINT16 ((guint16) 0xffff)
#define GUINT32_TO_LE(val) ((guint32) (val))
#define G_MAXUINT32 ((guint32) 0xffffffff)
#define GUINT64_TO_LE(val) ((guint64) (val))
#define G_MAXUINT8 ((guint8) 0xff)
#define g_random_boolean() ((g_random_int () & (1 << 15)) != 0)
#define g_rand_boolean(rand_) ((g_rand_int (rand_) & (1 << 15)) != 0)
#define g_list_next(list) ((list) ? (((GList *) (list))->next) : NULL)
#define g_list_previous(list) ((list) ? (((GList *) (list))->prev) : NULL)
#define g_slist_next(slist) ((slist) ? (((GSList *) (slist))->next) : NULL)
#define g_atomic_int_set(atomic, newval) ((void) (*(atomic) = (newval)))
#define g_atomic_pointer_set(atomic, newval) ((void) (*(atomic) = (newval)))
#define g_atomic_int_get(atomic) (*(atomic))
#define g_atomic_pointer_get(atomic) (*(atomic))
#define G_LOG_2_BASE_10 (0.30102999566398119521)
#define G_ALLOCATOR_LIST (1)
#define G_IEEE754_DOUBLE_BIAS (1023)
#define G_IEEE754_FLOAT_BIAS (127)
#define G_ALLOCATOR_SLIST (2)
#define G_ASCII_DTOSTR_BUF_SIZE (29 + 10)
#define G_ALLOCATOR_NODE (3)
#define G_HOOK_FLAG_USER_SHIFT (4)
#define G_LOG_LEVEL_USER_SHIFT (8)
#define g_ATEXIT(proc) (atexit (proc))
#define g_utf8_next_char(p) (char *) ((p) + g_utf8_skip[(const gchar *) (p)])
#define G_LIKELY(expr) (expr)
#define G_UNLIKELY(expr) (expr)
#define GINT16_FROM_BE(val) (GINT16_TO_BE (val))
#define GINT16_FROM_LE(val) (GINT16_TO_LE (val))
#define GINT32_FROM_BE(val) (GINT32_TO_BE (val))
#define GINT32_FROM_LE(val) (GINT32_TO_LE (val))
#define GINT64_FROM_BE(val) (GINT64_TO_BE (val))
#define GINT64_FROM_LE(val) (GINT64_TO_LE (val))
#define GINT_FROM_BE(val) (GINT_TO_BE (val))

```



```

#define GINT_FROM_LE(val)      (GINT_TO_LE (val))
#define GLONG_FROM_BE(val)     (GLONG_TO_BE (val))
#define GLONG_FROM_LE(val)     (GLONG_TO_LE (val))
#define g_ntohs(val)           (GUIN16_FROM_BE (val))
#define GUIN16_SWAP_BE_PDP(val) (GUIN16_SWAP_LE_BE (val))
#define GUIN16_TO_BE(val)      (GUIN16_SWAP_LE_BE (val))
#define GUIN16_SWAP_LE_BE(val) (GUIN16_SWAP_LE_BE_CONSTANT (val))
#define GUIN16_FROM_BE(val)     (GUIN16_TO_BE (val))
#define g_htons(val)           (GUIN16_TO_BE (val))
#define GUIN16_FROM_LE(val)     (GUIN16_TO_LE (val))
#define g_ntohl(val)           (GUIN32_FROM_BE (val))
#define GUIN32_TO_BE(val)      (GUIN32_SWAP_LE_BE (val))
#define GUIN32_SWAP_LE_BE(val) (GUIN32_SWAP_LE_BE_CONSTANT (val))
#define GUIN32_FROM_BE(val)     (GUIN32_TO_BE (val))
#define g_htonl(val)           (GUIN32_TO_BE (val))
#define GUIN32_FROM_LE(val)     (GUIN32_TO_LE (val))
#define GUIN64_TO_BE(val)      (GUIN64_SWAP_LE_BE (val))
#define GUIN64_SWAP_LE_BE(val) (GUIN64_SWAP_LE_BE_CONSTANT (val))
#define GUIN64_FROM_BE(val)     (GUIN64_TO_BE (val))
#define GUIN64_FROM_LE(val)     (GUIN64_TO_LE (val))
#define GUINT_FROM_BE(val)      (GUINT_TO_BE (val))
#define GUINT_FROM_LE(val)      (GUINT_TO_LE (val))
#define GULONG_FROM_BE(val)     (GULONG_TO_BE (val))
#define GULONG_FROM_LE(val)     (GULONG_TO_LE (val))
#define g_atomic_int_inc(atomic) (g_atomic_int_add ((atomic),
1))
#define G_HOOK_FLAGS(hook)      (G_HOOK (hook)->flags)
#define G_LOG_FATAL_MASK        (G_LOG_FLAG_RECURSION |
G_LOG_LEVEL_ERROR)
#define G_OPTION_ERROR (g_option_error_quark ())
#define g_thread_supported()    (g_threads_got_initialized)
#define G_N_ELEMENTS(arr)      (sizeof (arr) / sizeof ((arr)[0]))
#define G_STMT_START (void) __extension__ (
#define G_STMT_END )
#define G_PRIORITY_HIGH -100
#define G_DIR_SEPARATOR_S "/"
#define G_HAVE_GROWING_STACK 0
#define G_PRIORITY_DEFAULT 0
#define G_LN2
0.69314718055994530941723212145817656807550013436026
#define G_PI_4
0.78539816339744830961566084581987572104929234984378
#define G_CSET_DIGITS "0123456789"
#define G_DATE_BAD_DAY 0U
#define G_DATE_BAD_JULIAN 0U
#define G_DATE_BAD_YEAR 0U
#define G_MUTEX_DEBUG_MAGIC 0xf8e18ad7
#define GLIB_MICRO_VERSION 1
#define G_ALLOC_ONLY 1
#define G_CAN_INLINE 1
#define G_HAVE_GINT64 1
#define G_HAVE_GNUC_VARARGS 1
#define G_HAVE_GNUC_VISIBILITY 1
#define G_HAVE_INLINE 1
#define G_HAVE_ISO_VARARGS 1
#define G_HAVE___INLINE 1
#define G_HAVE___INLINE__ 1
#define G_SQRT2
1.4142135623730950488016887242096980785696718753769
#define G_PI_2
1.5707963267948966192313216916397514420985846996876
#define G_PRIORITY_HIGH_IDLE 100
#define G_USEC_PER_SEC 1000000
#define G_LITTLE_ENDIAN 1234
#define GLIB_MAJOR_VERSION 2
#define G_ALLOC_AND_FREE 2

```



```

#define G_LN10
2.3025850929940456840179914546843642076011014886288
#define G_E
2.7182818284590452353602874713526624977572470937000
#define G_PRIORITY_DEFAULT_IDLE 200
#define G_PI
3.1415926535897932384626433832795028841971693993751
#define G_PRIORITY_LOW 300
#define GLIB_MINOR_VERSION 32
#define G_PDP_ENDIAN 3412
#define G_BIG_ENDIAN 4321
#define G_SEARCHPATH_SEPARATOR_S ":"
#define GLIB_SYSDEF_POLLIN =1
#define GLIB_SYSDEF_POLLHUP =16
#define GLIB_SYSDEF_POLLPRI =2
#define GLIB_SYSDEF_POLLNVAL =32
#define GLIB_SYSDEF_POLLOUT =4
#define GLIB_SYSDEF_POLLERR =8
#define G_CSET_A_2_Z "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
#define G_CSET_a_2_z "abcdefghijklmnopqrstuvwxyz"
#define g_alloca(size) alloca (size)
#define G_CONST_RETURN const /* DEPRECATED */
#define G_MAXDOUBLE DBL_MAX
#define G_MINDOUBLE DBL_MIN
#define GLIB_VAR extern
#define G_LOCK_EXTERN(name) extern GStaticMutex G_LOCK_NAME
(name)
#define G_INLINE_FUNC extern inline
#define G_MAXFLOAT FLT_MAX
#define G_MINFLOAT FLT_MIN
#define G_MEM_ALIGN GLIB_SIZEOF_LONG
#define g_array_append_val(a,v) g_array_append_vals (a, &(v), 1)
#define g_array_insert_val(a,i,v) g_array_insert_vals (a, i,
&(v), 1)
#define g_array_prepend_val(a,v) g_array_prepend_vals (a,
&(v), 1)
#define ATEXTIT(proc) g_ATEXTIT(proc)
#define G_CONVERT_ERROR g_convert_error_quark()
#define g_date_day g_date_get_day
#define g_date_days_in_month g_date_get_days_in_month
#define g_date_day_of_year g_date_get_day_of_year
#define g_date_julian g_date_get_julian
#define g_date_monday_weeks_in_year
g_date_get_monday_weeks_in_year
#define g_date_monday_week_of_year
g_date_get_monday_week_of_year
#define g_date_month g_date_get_month
#define g_date_sunday_weeks_in_year
g_date_get_sunday_weeks_in_year
#define g_date_sunday_week_of_year
g_date_get_sunday_week_of_year
#define g_date_weekday g_date_get_weekday
#define g_date_year g_date_get_year
#define G_FILE_ERROR g_file_error_quark ()
#define G_MAXINT64 G_GINT64_CONSTANT(0x7fffffffffffffff)
#define G_MININT64 G_GINT64_CONSTANT(0x8000000000000000)
#define G_MAXUINT64 G_GINT64_CONSTANT(0xffffffffffffffffU)
#define G_IO_CHANNEL_ERROR g_io_channel_error_quark()
#define G_KEY_FILE_ERROR g_key_file_error_quark()
#define g_debug(...) g_log (G_LOG_DOMAIN, G_LOG_LEVEL_DEBUG,
__VA_ARGS__)
#define g_error(...) g_log (G_LOG_DOMAIN, G_LOG_LEVEL_ERROR,
__VA_ARGS__)
#define G_MARKUP_ERROR g_markup_error_quark ()
#define G_SHELL_ERROR g_shell_error_quark ()
#define G_SPAWN_ERROR g_spawn_error_quark ()

```

```

#define g_static_mutex_get_mutex(mutex)
g_static_mutex_get_mutex_impl
#define G_LOCK(name) g_static_mutex_lock (&G_LOCK_NAME (name))
#define G_TRYLOCK(name) g_static_mutex_trylock (&G_LOCK_NAME
(name))
#define G_UNLOCK(name) g_static_mutex_unlock (&G_LOCK_NAME (name))
#define g_strstrip(string) g_strchomp (g_strchug (string))
#define G_STRINGIFY(macro_or_string) G_STRINGIFY_ARG
(macro_or_string)
#define g_cond_free(cond) G_THREAD_CF (cond_free, (void)0,
(cond))
#define g_cond_signal(cond) G_THREAD_CF (cond_signal, (void)0,
(cond))
#define g_mutex_free(mutex) G_THREAD_CF (mutex_free, (void)0,
(mutex))
#define g_mutex_lock(mutex) G_THREAD_CF (mutex_lock, (void)0,
(mutex))
#define g_mutex_trylock(mutex) G_THREAD_CF (mutex_trylock, TRUE,
(mutex))
#define g_mutex_unlock(mutex) G_THREAD_CF (mutex_unlock, (void)0,
(mutex))
#define g_thread_yield() G_THREAD_CF (thread_yield, (void)0,
())
#define G_THREAD_ERROR g_thread_error_quark ()
#define g_cond_new() G_THREAD_UF (cond_new, ())
#define g_mutex_new() G_THREAD_UF (mutex_new, ())
#define g_private_new(destructor) G_THREAD_UF (private_new,
(destructor))
#define G_LOCK_NAME(name) g_ ## name ## _lock
#define G_GINT16_MODIFIER "h"
#define G_GINT16_FORMAT "hi"
#define G_GUINT16_FORMAT "hu"
#define G_GINT32_FORMAT "i"
#define G_MAXINT INT_MAX
#define G_MININT INT_MIN
#define G_MAXLONG LONG_MAX
#define G_MINLONG LONG_MIN
#define G_MAXSHORT SHRT_MAX
#define G_MINSHORT SHRT_MIN
#define G_MODULE_SUFFIX "so"
#define G_LOCK_DEFINE_STATIC(name) static G_LOCK_DEFINE (name)
#define G_GUINT32_FORMAT "u"
#define G_MAXUINT UINT_MAX
#define G_MAXULONG ULONG_MAX
#define G_MAXUSHORT USHRT_MAX
#define G_VA_COPY va_copy
#define G_STR_DELIMITERS "-|> <."
#define G_GNUC_INTERNAL __attribute__((visibility("hidden")))
#define G_GNUC_CONST __attribute__((const))
#define G_GNUC_DEPRECATED __attribute__((deprecated))
#define G_GNUC_FORMAT(arg_idx) __attribute__((format_arg__
(arg_idx)))
#define G_GNUC_MALLOC __attribute__((malloc))
#define G_GNUC_NORETURN __attribute__((noreturn))
#define G_GNUC_NO_INSTRUMENT
__attribute__((no_instrument_function))
#define G_GNUC_PURE __attribute__((pure))
#define G_GNUC_UNUSED __attribute__((unused))
#define alloca(size) __builtin_alloca (size)
#define G_GNUC_EXTENSION __extension__
#define G_STRLOC __FILE__ ":" G_STRINGIFY (__LINE__)
#define G_STATIC_PRIVATE_INIT { 0 }
#define G_ONCE_INIT { G_ONCE_STATUS_NOTCALLED, NULL }
#define G_STATIC_REC_MUTEX_INIT { G_STATIC_MUTEX_INIT }
#define G_STATIC_MUTEX_INIT { NULL }

```

```

typedef float gfloat;
typedef gint64 goffset;

#define __G_BOOKMARK_FILE_H__
#define __G_SLICE_H__
#define g_slice_new(type) ((type*) g_slice_alloc (sizeof
(type))
#define g_slice_new0(type) ((type*) g_slice_alloc0 (sizeof
(type))
#define g_slice_dup(type,mem) (1 ? (type*) g_slice_copy (sizeof
(type), (mem)) : ((void) ((type*) 0 == (mem)), (type*) 0))
#define G_BOOKMARK_FILE_ERROR (g_bookmark_file_error_quark ())
#define g_slice_free(type,mem) do { if (1) g_slice_free1 (sizeof
(type), (mem)); else (void) ((type*) 0 == (mem)); } while (0)
#define g_slice_free_chain(type,mem_chain,next) do { if (1)
g_slice_free_chain_with_offset (sizeof (type), (mem_chain),
G_STRUCT_OFFSET (type, next)); else (void) ((type*) 0 ==
(mem_chain)); } while (0)

typedef short unsigned int guint16;
typedef int gint;
typedef gint gboolean;
typedef unsigned int guint;
typedef void *gpointer;
typedef gpointer(*GThreadFunc) (gpointer);
typedef enum {
    G_THREAD_PRIORITY_LOW = 0,
    G_THREAD_PRIORITY_NORMAL = 1,
    G_THREAD_PRIORITY_HIGH = 2,
    G_THREAD_PRIORITY_URGENT = 3
} GThreadPriority;
typedef struct _GThread {
    GThreadFunc func;
    gpointer data;
    gboolean joinable;
    GThreadPriority priority;
} GThread;
typedef unsigned int guint32;
typedef guint32 GQuark;
typedef char gchar;
typedef struct _GError {
    GQuark domain;
    gint code;
    gchar *message;
} GError;
typedef void (*GPrintFunc) (const gchar *);
typedef struct _GMemChunk GMemChunk;
typedef long unsigned int gulong;
typedef struct _GList {
    gpointer data;
    GList *next;
    GList *prev;
} GList;
typedef struct _GMainLoop GMainLoop;
typedef struct _GHook {
    gpointer data;
    GHook *next;
    GHook *prev;
    guint ref_count;
    gulong hook_id;
    guint flags;
    gpointer func;
    GDestroyNotify destroy;
} GHook;
typedef void (*GDestroyNotify) (gpointer);
typedef struct _GHookList {

```

```

        gulong seq_id;
        guint hook_size:16;
        guint is_setup:1;
        GHook *hooks;
        gpointer dummy3;
        GHookFinalizeFunc finalize_hook;
        gpointer dummy[2];
    } GHookList;
typedef void (*GHookFinalizeFunc) (GHookList *, GHook *);
typedef struct _GQueue {
    GList *head;
    GList *tail;
    guint length;
} GQueue;
typedef struct _GSList {
    gpointer data;
    GSList *next;
} GSList;
typedef struct _GString {
    gchar *str;
    gsize len;
    gsize allocated_len;
} GString;
typedef struct _GPtrArray {
    gpointer *pdata;
    guint len;
} GPtrArray;
typedef struct _GRandom GRand;
typedef struct _GDir GDir;
typedef struct _GRelation GRelation;
typedef struct _GOptionContext GOptionContext;
typedef struct _GKeyFile GKeyFile;
typedef struct _GPatternSpec GPatternSpec;
typedef guint32 gunichar;
typedef gchar *(*GCompletionFunc) (gpointer);
typedef gint(*GCompletionStrncmpFunc) (const gchar *, const gchar
*,
                                gsize);

typedef struct _GCompletion {
    GList *items;
    GCompletionFunc func;
    gchar *prefix;
    GList *cache;
    GCompletionStrncmpFunc strncmp_func;
} GCompletion;
typedef struct _GTimer GTimer;
typedef struct _GHashTable GHashTable;
typedef const void *gconstpointer;
typedef guint(*GHashFunc) (gconstpointer);
typedef gboolean(*GEqualFunc) (gconstpointer, gconstpointer);
typedef union _GMutex {
    gpointer p;
    guint i[2];
} GMutex;
typedef union _GSystemThread {
    char data[GLIB_LSB_DATA_SIZE];
    double dummy_double;
    void *dummy_pointer;
    long int dummy_long;
} GSystemThread;
typedef struct _GMarkupParseContext GMarkupParseContext;
typedef struct _GData GData;
typedef enum {
    G_IO_STATUS_ERROR = 0,
    G_IO_STATUS_NORMAL = 1,
    G_IO_STATUS_EOF = 2,

```

```

    G_IO_STATUS_AGAIN = 3
} GIOStatus;
typedef struct _GIOChannel {
    guint ref_count;
    GIOFuncs *funcs;
    gchar *encoding;
    GIconv read_cd;
    GIconv write_cd;
    gchar *line_term;
    guint line_term_len;
    gsize buf_size;
    GString *read_buf;
    GString *encoded_read_buf;
    GString *write_buf;
    gchar partial_write_buf[6];
    guint use_buffer:1;
    guint do_encode:1;
    guint close_on_unref:1;
    guint is_readable:1;
    guint is_writable:1;
    guint is_seekable:1;
    gpointer reserved1;
    gpointer reserved2;
} GIOChannel;
typedef enum {
    G_SEEK_CUR = 0,
    G_SEEK_SET = 1,
    G_SEEK_END = 2
} GSeekType;
typedef struct _GSource {
    gpointer callback_data;
    GSourceCallbackFuncs *callback_funcs;
    GSourceFuncs *source_funcs;
    guint ref_count;
    GMainContext *context;
    gint priority;
    guint flags;
    guint source_id;
    GSList *poll_fds;
    GSource *prev;
    GSource *next;
    char *name;
    GSourcePrivate *priv;
} GSource;
typedef gboolean(*GSourceFunc) (gpointer);
typedef struct _GSourceCallbackFuncs {
    void (*ref) (gpointer);
    void (*unref) (gpointer);
    void (*get) (gpointer, GSource *, GSourceFunc *, gpointer *);
} GSourceCallbackFuncs;
typedef void (*GSourceDummyMarshal) (void);
typedef struct _GSourceFuncs {
    gboolean(*prepare) (GSource *, gint *);
    gboolean(*check) (GSource *);
    gboolean(*dispatch) (GSource *, GSourceFunc, gpointer);
    void (*finalize) (GSource *);
    GSourceFunc closure_callback;
    GSourceDummyMarshal closure_marshal;
} GSourceFuncs;
typedef struct _GMainContext GMainContext;
typedef enum {
    G_IO_IN = 1,
    G_IO_OUT = 4,
    G_IO_PRI = 2,
    G_IO_ERR = 8,
    G_IO_HUP = 16,

```

```

    G_IO_NVAL = 32
} GIOCondition;
typedef enum {
    G_IO_FLAG_APPEND = 1,
    G_IO_FLAG_NONBLOCK = 2,
    G_IO_FLAG_IS_READABLE = 4,
    G_IO_FLAG_IS_WRITEABLE = 8,
    G_IO_FLAG_IS_SEEKABLE = 16,
    G_IO_FLAG_MASK = 31,
    G_IO_FLAG_GET_MASK = 31,
    G_IO_FLAG_SET_MASK = 3
} GIOFlags;
typedef struct _GIOFuncs {
    GIOStatus(*io_read) (GIOChannel *, gchar *, gsize, gsize *,
                        GError * *);
    GIOStatus(*io_write) (GIOChannel *, const gchar *, gsize, gsize
*,
                        GError * *);
    GIOStatus(*io_seek) (GIOChannel *, gint64, GSeekType, GError *
*);
    GIOStatus(*io_close) (GIOChannel *, GError * *);
    GSource *(*io_create_watch) (GIOChannel *, GIOCondition);
    void (*io_free) (GIOChannel *);
    GIOStatus(*io_set_flags) (GIOChannel *, GIOFlags, GError * *);
    GIOFlags(*io_get_flags) (GIOChannel *);
} GIOFuncs;
typedef struct _GIconv *GIconv;
typedef struct _GOptionGroup GOptionGroup;
typedef enum {
    G_SPAWN_LEAVE_DESCRIPTOR_OPEN = 1,
    G_SPAWN_DO_NOT_REAP_CHILD = 2,
    G_SPAWN_SEARCH_PATH = 4,
    G_SPAWN_STDOUT_TO_DEV_NULL = 8,
    G_SPAWN_STDERR_TO_DEV_NULL = 16,
    G_SPAWN_CHILD_INHERITS_STDIN = 32,
    G_SPAWN_FILE_AND_ARGV_ZERO = 64
} GSpawnFlags;
typedef void (*GSpawnChildSetupFunc) (gpointer);
typedef int GPid;
typedef void (*GFunc) (gpointer, gpointer);
typedef struct _GThreadPool {
    GFunc func;
    gpointer user_data;
    gboolean exclusive;
} GThreadPool;
typedef struct _GDate {
    quint julian_days:32;
    quint julian:1;
    quint dmy:1;
    quint day:6;
    quint month:4;
    quint year:16;
} GDate;
typedef enum {
    G_DATE_BAD_MONTH = 0,
    G_DATE_JANUARY = 1,
    G_DATE_FEBRUARY = 2,
    G_DATE_MARCH = 3,
    G_DATE_APRIL = 4,
    G_DATE_MAY = 5,
    G_DATE_JUNE = 6,
    G_DATE_JULY = 7,
    G_DATE_AUGUST = 8,
    G_DATE_SEPTEMBER = 9,
    G_DATE_OCTOBER = 10,
    G_DATE_NOVEMBER = 11,

```

```

    G_DATE_DECEMBER = 12
} GDateMonth;
typedef struct _GAsyncQueue GAsyncQueue;
typedef short unsigned int gushort;
typedef struct _GPollFD {
    gint fd;
    gushort events;
    gushort revents;
} GPollFD;
typedef double gdouble;
typedef struct _GTree GTree;
typedef gint (*GCompareDataFunc) (gconstpointer, gconstpointer,
gpointer);
typedef unsigned char guint8;
typedef guint16 GDateYear;
typedef struct _GCache GCache;
typedef void (*GHFunc) (gpointer, gpointer, gpointer);
typedef struct _GScannerConfig {
    gchar *cset_skip_characters;
    gchar *cset_identifier_first;
    gchar *cset_identifier_nth;
    gchar *cpair_comment_single;
    guint case_sensitive:1;
    guint skip_comment_multi:1;
    guint skip_comment_single:1;
    guint scan_comment_multi:1;
    guint scan_identifier:1;
    guint scan_identifier_lchar:1;
    guint scan_identifier_NULL:1;
    guint scan_symbols:1;
    guint scan_binary:1;
    guint scan_octal:1;
    guint scan_float:1;
    guint scan_hex:1;
    guint scan_hex_dollar:1;
    guint scan_string_sq:1;
    guint scan_string_dq:1;
    guint numbers_2_int:1;
    guint int_2_float:1;
    guint identifier_2_string:1;
    guint char_2_token:1;
    guint symbol_2_token:1;
    guint scope_0_fallback:1;
    guint store_int64:1;
    guint padding_dummy;
} GScannerConfig;
typedef enum {
    G_TOKEN_EOF = 0,
    G_TOKEN_LEFT_PAREN = 40,
    G_TOKEN_RIGHT_PAREN = 41,
    G_TOKEN_LEFT_CURLY = 123,
    G_TOKEN_RIGHT_CURLY = 125,
    G_TOKEN_LEFT_BRACE = 91,
    G_TOKEN_RIGHT_BRACE = 93,
    G_TOKEN_EQUAL_SIGN = 61,
    G_TOKEN_COMMA = 44,
    G_TOKEN_NONE = 256,
    G_TOKEN_ERROR = 257,
    G_TOKEN_CHAR = 258,
    G_TOKEN_BINARY = 259,
    G_TOKEN_OCTAL = 260,
    G_TOKEN_INT = 261,
    G_TOKEN_HEX = 262,
    G_TOKEN_FLOAT = 263,
    G_TOKEN_STRING = 264,
    G_TOKEN_SYMBOL = 265,

```

```

        G_TOKEN_IDENTIFIER = 266,
        G_TOKEN_IDENTIFIER_NULL = 267,
        G_TOKEN_COMMENT_SINGLE = 268,
        G_TOKEN_COMMENT_MULTI = 269,
        G_TOKEN_LAST = 270
    } GTokenType;
typedef unsigned char guchar;
typedef union _GTokenValue {
    gpointer v_symbol;
    gchar *v_identifier;
    gulong v_binary;
    gulong v_octal;
    gulong v_int;
    guint64 v_int64;
    gdouble v_float;
    gulong v_hex;
    gchar *v_string;
    gchar *v_comment;
    guchar v_char;
    guint v_error;
} GTokenValue;
typedef struct _GScanner {
    gpointer user_data;
    guint max_parse_errors;
    guint parse_errors;
    const gchar *input_name;
    GData *qdata;
    GScannerConfig *config;
    GTokenType token;
    GTokenValue value;
    guint line;
    guint position;
    GTokenType next_token;
    GTokenValue next_value;
    guint next_line;
    guint next_position;
    GHashTable *symbol_table;
    gint input_fd;
    const gchar *text;
    const gchar *text_end;
    gchar *buffer;
    guint scope_id;
    GScannerMsgFunc msg_handler;
} GScanner;
typedef void (*GScannerMsgFunc) (GScanner *, gchar *, gboolean);
typedef struct _GByteArray {
    guint8 *data;
    guint len;
} GByteArray;
typedef enum {
    G_KEY_FILE_NONE = 0,
    G_KEY_FILE_KEEP_COMMENTS = 1,
    G_KEY_FILE_KEEP_TRANSLATIONS = 2
} GKeyFileFlags;
typedef struct _GTrashStack {
    GTrashStack *next;
} GTrashStack;
typedef guint16 gunichar2;
typedef long int glong;
typedef struct _GArray {
    gchar *data;
    guint len;
} GArray;
typedef struct _GNode {
    gpointer data;
    GNode *next;

```



```

    GNode *prev;
    GNode *parent;
    GNode *children;
} GNode;
typedef gboolean(*GHRFunc) (gpointer, gpointer, gpointer);
typedef gint(*GCompareFunc) (gconstpointer, gconstpointer);
typedef int gint32;
typedef gint32 GTime;
typedef gint(*GPollFunc) (GPollFD *, guint, gint);
typedef guint8 GDateDay;
typedef enum {
    G_UNICODE_BREAK_MANDATORY = 0,
    G_UNICODE_BREAK_CARRIAGE_RETURN = 1,
    G_UNICODE_BREAK_LINE_FEED = 2,
    G_UNICODE_BREAK_COMBINING_MARK = 3,
    G_UNICODE_BREAK_SURROGATE = 4,
    G_UNICODE_BREAK_ZERO_WIDTH_SPACE = 5,
    G_UNICODE_BREAK_INSEPARABLE = 6,
    G_UNICODE_BREAK_NON_BREAKING_GLUE = 7,
    G_UNICODE_BREAK_CONTINGENT = 8,
    G_UNICODE_BREAK_SPACE = 9,
    G_UNICODE_BREAK_AFTER = 10,
    G_UNICODE_BREAK_BEFORE = 11,
    G_UNICODE_BREAK_BEFORE_AND_AFTER = 12,
    G_UNICODE_BREAK_HYPHEN = 13,
    G_UNICODE_BREAK_NON_STARTER = 14,
    G_UNICODE_BREAK_OPEN_PUNCTUATION = 15,
    G_UNICODE_BREAK_CLOSE_PUNCTUATION = 16,
    G_UNICODE_BREAK_QUOTATION = 17,
    G_UNICODE_BREAK_EXCLAMATION = 18,
    G_UNICODE_BREAK_IDEOGRAPHIC = 19,
    G_UNICODE_BREAK_NUMERIC = 20,
    G_UNICODE_BREAK_INFIX_SEPARATOR = 21,
    G_UNICODE_BREAK_SYMBOL = 22,
    G_UNICODE_BREAK_ALPHABETIC = 23,
    G_UNICODE_BREAK_PREFIX = 24,
    G_UNICODE_BREAK_POSTFIX = 25,
    G_UNICODE_BREAK_COMPLEX_CONTEXT = 26,
    G_UNICODE_BREAK_AMBIGUOUS = 27,
    G_UNICODE_BREAK_UNKNOWN = 28,
    G_UNICODE_BREAK_NEXT_LINE = 29,
    G_UNICODE_BREAK_WORD_JOINER = 30
} GUnicodeBreakType;
typedef struct _GStringChunk GStringChunk;
typedef struct _GCond GCond;
typedef void (*GChildWatchFunc) (GPid, gint, gpointer);
typedef struct _GTimeVal {
    glong tv_sec;
    glong tv_usec;
} GTimeVal;
typedef enum {
    G_LOG_FLAG_RECURSION = 1,
    G_LOG_FLAG_FATAL = 2,
    G_LOG_LEVEL_ERROR = 4,
    G_LOG_LEVEL_CRITICAL = 8,
    G_LOG_LEVEL_WARNING = 16,
    G_LOG_LEVEL_MESSAGE = 32,
    G_LOG_LEVEL_INFO = 64,
    G_LOG_LEVEL_DEBUG = 128,
    G_LOG_LEVEL_MASK = -4
} GLogLevelFlags;
typedef enum {
    G_DATE_BAD_WEEKDAY = 0,
    G_DATE_MONDAY = 1,
    G_DATE_TUESDAY = 2,
    G_DATE_WEDNESDAY = 3,

```

```

        G_DATE_THURSDAY = 4,
        G_DATE_FRIDAY = 5,
        G_DATE_SATURDAY = 6,
        G_DATE_SUNDAY = 7
    } GDateWeekday;
typedef enum {
    G_IN_ORDER = 0,
    G_PRE_ORDER = 1,
    G_POST_ORDER = 2,
    G_LEVEL_ORDER = 3
} GTraverseType;
typedef enum {
    G_TRAVERSE_LEAVES = 1,
    G_TRAVERSE_NON_LEAVES = 2,
    G_TRAVERSE_ALL = 3,
    G_TRAVERSE_MASK = 3,
    G_TRAVERSE_LEAFS = 1,
    G_TRAVERSE_NON_LEAFS = 2
} GTraverseFlags;
typedef struct _GMarkupParser {
    void (*start_element) (GMarkupParseContext *, const gchar *,
                          const gchar **, const gchar **, gpointer,
                          GError * *);
    void (*end_element) (GMarkupParseContext *, const gchar *,
                        gpointer,
                        GError * *);
    void (*text) (GMarkupParseContext *, const gchar *, gsize,
                 gpointer,
                 GError * *);
    void (*passthrough) (GMarkupParseContext *, const gchar *, gsize,
                        gpointer, GError * *);
    void (*error) (GMarkupParseContext *, GError *, gpointer);
} GMarkupParser;
typedef enum {
    G_MARKUP_DO_NOT_USE_THIS_UNSUPPORTED_FLAG = 1
} GMarkupParseFlags;
typedef gboolean(*GHookCheckMarshaller) (GHook *, gpointer);
typedef gboolean(*GNodeTraverseFunc) (GNode *, gpointer);
typedef enum {
    G_NORMALIZE_DEFAULT = 0,
    G_NORMALIZE_NFD = 0,
    G_NORMALIZE_DEFAULT_COMPOSE = 1,
    G_NORMALIZE_NFC = 1,
    G_NORMALIZE_ALL = 2,
    G_NORMALIZE_NFKD = 2,
    G_NORMALIZE_ALL_COMPOSE = 3,
    G_NORMALIZE_NFKC = 3
} GNormalizeMode;
typedef struct {
    GMutex *mutex;
} GStaticMutex;
typedef struct _GStaticPrivate {
    guint index;
} GStaticPrivate;
typedef enum {
    G_FILE_ERROR_EXIST = 0,
    G_FILE_ERROR_ISDIR = 1,
    G_FILE_ERROR_ACCES = 2,
    G_FILE_ERROR_NAMETOOLONG = 3,
    G_FILE_ERROR_NOENT = 4,
    G_FILE_ERROR_NOTDIR = 5,
    G_FILE_ERROR_NXIO = 6,
    G_FILE_ERROR_NODEV = 7,
    G_FILE_ERROR_ROFS = 8,
    G_FILE_ERROR_TXTBSY = 9,
    G_FILE_ERROR_FAULT = 10,

```

```

G_FILE_ERROR_LOOP = 11,
G_FILE_ERROR_NOSPC = 12,
G_FILE_ERROR_NOMEM = 13,
G_FILE_ERROR_MFILE = 14,
G_FILE_ERROR_NFILE = 15,
G_FILE_ERROR_BADF = 16,
G_FILE_ERROR_INVALID = 17,
G_FILE_ERROR_PIPE = 18,
G_FILE_ERROR_AGAIN = 19,
G_FILE_ERROR_INTR = 20,
G_FILE_ERROR_IO = 21,
G_FILE_ERROR_PERM = 22,
G_FILE_ERROR_NOSYS = 23,
G_FILE_ERROR_FAILED = 24
} GFileError;
typedef void (*GDataForeachFunc) (GQuark, gpointer, gpointer);
typedef struct _GMemVTable {
    gpointer(*malloc) (gsize);
    gpointer(*realloc) (gpointer, gsize);
    void(*free) (gpointer);
    gpointer(*calloc) (gsize, gsize);
    gpointer(*try_malloc) (gsize);
    gpointer(*try_realloc) (gpointer, gsize);
} GMemVTable;
typedef enum {
    G_OPTION_ARG_NONE = 0,
    G_OPTION_ARG_STRING = 1,
    G_OPTION_ARG_INT = 2,
    G_OPTION_ARG_CALLBACK = 3,
    G_OPTION_ARG_FILENAME = 4,
    G_OPTION_ARG_STRING_ARRAY = 5,
    G_OPTION_ARG_FILENAME_ARRAY = 6
} GOptionArg;
typedef struct _GOptionEntry {
    const gchar *long_name;
    gchar short_name;
    gint flags;
    GOptionArg arg;
    gpointer arg_data;
    const gchar *description;
    const gchar *arg_description;
} GOptionEntry;
typedef enum {
    G_UNICODE_CONTROL = 0,
    G_UNICODE_FORMAT = 1,
    G_UNICODE_UNASSIGNED = 2,
    G_UNICODE_PRIVATE_USE = 3,
    G_UNICODE_SURROGATE = 4,
    G_UNICODE_LOWERCASE_LETTER = 5,
    G_UNICODE_MODIFIER_LETTER = 6,
    G_UNICODE_OTHER_LETTER = 7,
    G_UNICODE_TITLECASE_LETTER = 8,
    G_UNICODE_UPPERCASE_LETTER = 9,
    G_UNICODE_COMBINING_MARK = 10,
    G_UNICODE_ENCLOSING_MARK = 11,
    G_UNICODE_NON_SPACING_MARK = 12,
    G_UNICODE_DECIMAL_NUMBER = 13,
    G_UNICODE_LETTER_NUMBER = 14,
    G_UNICODE_OTHER_NUMBER = 15,
    G_UNICODE_CONNECT_PUNCTUATION = 16,
    G_UNICODE_DASH_PUNCTUATION = 17,
    G_UNICODE_CLOSE_PUNCTUATION = 18,
    G_UNICODE_FINAL_PUNCTUATION = 19,
    G_UNICODE_INITIAL_PUNCTUATION = 20,
    G_UNICODE_OTHER_PUNCTUATION = 21,
    G_UNICODE_OPEN_PUNCTUATION = 22,

```

```

    G_UNICODE_CURRENCY_SYMBOL = 23,
    G_UNICODE_MODIFIER_SYMBOL = 24,
    G_UNICODE_MATH_SYMBOL = 25,
    G_UNICODE_OTHER_SYMBOL = 26,
    G_UNICODE_LINE_SEPARATOR = 27,
    G_UNICODE_PARAGRAPH_SEPARATOR = 28,
    G_UNICODE_SPACE_SEPARATOR = 29
} GUnicodeType;
typedef void (*GLogFunc) (const gchar *, GLogLevelFlags, const
gchar *,
                        gpointer);
typedef struct _GAllocator GAllocator;
typedef const gchar *(*GTranslateFunc) (const gchar *, gpointer);
typedef gboolean(*GOptionParseFunc) (GOptionContext *,
GOptionGroup *,
                        gpointer, GError * *);
typedef void (*GVoidFunc) (void);
typedef gboolean(*GHookFindFunc) (GHook *, gpointer);
typedef struct _GTuples {
    guint len;
} GTuples;
typedef gpointer(*GCopyFunc) (gconstpointer, gpointer);
typedef void (*GOptionErrorFunc) (GOptionContext *, GOptionGroup *,
                        gpointer, GError * *);
typedef gpointer(*GCacheNewFunc) (gpointer);
typedef void (*GCacheDestroyFunc) (gpointer);
typedef gpointer(*GCacheDupFunc) (gpointer);
typedef enum {
    G_FILE_TEST_IS_REGULAR = 1,
    G_FILE_TEST_IS_SYMLINK = 2,
    G_FILE_TEST_IS_DIR = 4,
    G_FILE_TEST_IS_EXECUTABLE = 8,
    G_FILE_TEST_EXISTS = 16
} GFileTest;
typedef enum {
    G_ONCE_STATUS_NOTCALLED = 0,
    G_ONCE_STATUS_PROGRESS = 1,
    G_ONCE_STATUS_READY = 2
} GOnceStatus;
typedef struct _GOnce {
    volatile GOnceStatus status;
    volatile gpointer retval;
} GOnce;
typedef gboolean(*GTraverseFunc) (gpointer, gpointer, gpointer);
typedef gint(*GHookCompareFunc) (GHook *, GHook *);
typedef void (*GNodeForeachFunc) (GNode *, gpointer);
typedef struct _GDebugKey {
    gchar *key;
    guint value;
} GDebugKey;
typedef struct _GPrivate GPrivate;
typedef struct _GThreadFunctions {
    GMutex *(*mutex_new) (void);
    void (*mutex_lock) (GMutex *);
    gboolean(*mutex_trylock) (GMutex *);
    void (*mutex_unlock) (GMutex *);
    void (*mutex_free) (GMutex *);
    GCond *(*cond_new) (void);
    void (*cond_signal) (GCond *);
    void (*cond_broadcast) (GCond *);
    void (*cond_wait) (GCond *, GMutex *);
    gboolean(*cond_timed_wait) (GCond *, GMutex *, GTimeVal *);
    void (*cond_free) (GCond *);
    GPrivate *(*private_new) (GDestroyNotify);
    gpointer(*private_get) (GPrivate *);
    void (*private_set) (GPrivate *, gpointer);

```

```

void (*thread_create) (GThreadFunc, gpointer, gulong, gboolean,
                      gboolean, GThreadPriority, gpointer,
                      GError * *);
void (*thread_yield) (void);
void (*thread_join) (gpointer);
void (*thread_exit) (void);
void (*thread_set_priority) (gpointer, GThreadPriority);
void (*thread_self) (gpointer);
gboolean(*thread_equal) (gpointer, gpointer);
} GThreadFunctions;
typedef void (*GHookMarshaller) (GHook *, gpointer);
typedef enum {
    G_IO_CHANNEL_ERROR_FBIG = 0,
    G_IO_CHANNEL_ERROR_INVALID = 1,
    G_IO_CHANNEL_ERROR_IO = 2,
    G_IO_CHANNEL_ERROR_ISDIR = 3,
    G_IO_CHANNEL_ERROR_NOSPC = 4,
    G_IO_CHANNEL_ERROR_NXIO = 5,
    G_IO_CHANNEL_ERROR_OVERFLOW = 6,
    G_IO_CHANNEL_ERROR_PIPE = 7,
    G_IO_CHANNEL_ERROR_FAILED = 8
} GIOChannelError;
typedef gboolean(*GIOFunc) (GIOChannel *, GIOCondition, gpointer);
typedef void (*GFreeFunc) (gpointer);
typedef gboolean(*GHookCheckFunc) (gpointer);
typedef void (*GHookFunc) (gpointer);
typedef short int gint16;
typedef enum {
    G_DATE_DAY = 0,
    G_DATE_MONTH = 1,
    G_DATE_YEAR = 2
} GDateDMY;
typedef signed char gint8;
typedef enum {
    G_MARKUP_ERROR_BAD_UTF8 = 0,
    G_MARKUP_ERROR_EMPTY = 1,
    G_MARKUP_ERROR_PARSE = 2,
    G_MARKUP_ERROR_UNKNOWN_ELEMENT = 3,
    G_MARKUP_ERROR_UNKNOWN_ATTRIBUTE = 4,
    G_MARKUP_ERROR_INVALID_CONTENT = 5
} GMarkupError;
typedef enum {
    G_IO_ERROR_NONE = 0,
    G_IO_ERROR_AGAIN = 1,
    G_IO_ERROR_INVALID = 2,
    G_IO_ERROR_UNKNOWN = 3
} GIOError;
typedef gboolean(*GOptionArgFunc) (const gchar *, const gchar *,
gpointer,
                                GError * *);
typedef enum {
    G_OPTION_FLAG_HIDDEN = 1 << 0,
    G_OPTION_FLAG_IN_MAIN = 1 << 1,
    G_OPTION_FLAG_REVERSE = 1 << 2
} GOptionFlags;
typedef enum {
    G_CONVERT_ERROR_NO_CONVERSION,
    G_CONVERT_ERROR_ILLEGAL_SEQUENCE,
    G_CONVERT_ERROR_FAILED,
    G_CONVERT_ERROR_PARTIAL_INPUT,
    G_CONVERT_ERROR_BAD_URI,
    G_CONVERT_ERROR_NOT_ABSOLUTE_PATH
} GConvertError;
typedef enum {
    G_ERR_UNKNOWN,
    G_ERR_UNEXP_EOF,

```

```

        G_ERR_UNEXP_EOF_IN_STRING,
        G_ERR_UNEXP_EOF_IN_COMMENT,
        G_ERR_NON_DIGIT_IN_CONST,
        G_ERR_DIGIT_RADIX,
        G_ERR_FLOAT_RADIX,
        G_ERR_FLOAT_MALFORMED
    } GErrorType;
    typedef enum {
        G_KEY_FILE_ERROR_UNKNOWN_ENCODING,
        G_KEY_FILE_ERROR_PARSE,
        G_KEY_FILE_ERROR_NOT_FOUND,
        G_KEY_FILE_ERROR_KEY_NOT_FOUND,
        G_KEY_FILE_ERROR_GROUP_NOT_FOUND,
        G_KEY_FILE_ERROR_INVALID_VALUE
    } GKeyFileError;
    typedef enum {
        G_SPAWN_ERROR_FORK,
        G_SPAWN_ERROR_READ,
        G_SPAWN_ERROR_CHDIR,
        G_SPAWN_ERROR_ACCES,
        G_SPAWN_ERROR_PERM,
        G_SPAWN_ERROR_2BIG,
        G_SPAWN_ERROR_NOEXEC,
        G_SPAWN_ERROR_NAMETOOLONG,
        G_SPAWN_ERROR_NOENT,
        G_SPAWN_ERROR_NOMEM,
        G_SPAWN_ERROR_NOTDIR,
        G_SPAWN_ERROR_LOOP,
        G_SPAWN_ERROR_TXTBUSY,
        G_SPAWN_ERROR_IO,
        G_SPAWN_ERROR_NFILE,
        G_SPAWN_ERROR_MFILE,
        G_SPAWN_ERROR_INVAL,
        G_SPAWN_ERROR_ISDIR,
        G_SPAWN_ERROR_LIBBAD,
        G_SPAWN_ERROR_FAILED
    } GSpawnError;
    typedef enum {
        G_HOOK_FLAG_ACTIVE = 1 << 0,
        G_HOOK_FLAG_IN_CALL = 1 << 1,
        G_HOOK_FLAG_MASK = 0x0f
    } GHookFlagMask;
    typedef enum {
        G_THREAD_ERROR_AGAIN
    } GThreadError;
    typedef enum {
        G_OPTION_ERROR_UNKNOWN_OPTION,
        G_OPTION_ERROR_BAD_VALUE,
        G_OPTION_ERROR_FAILED
    } GOptionError;
    typedef enum {
        G_ASCII_ALNUM = 1 << 0,
        G_ASCII_ALPHA = 1 << 1,
        G_ASCII_CNTRL = 1 << 2,
        G_ASCII_DIGIT = 1 << 3,
        G_ASCII_GRAPH = 1 << 4,
        G_ASCII_LOWER = 1 << 5,
        G_ASCII_PRINT = 1 << 6,
        G_ASCII_PUNCT = 1 << 7,
        G_ASCII_SPACE = 1 << 8,
        G_ASCII_UPPER = 1 << 9,
        G_ASCII_XDIGIT = 1 << 10
    } GAsciiType;
    typedef enum {
        G_SHELL_ERROR_BAD_QUOTING,
        G_SHELL_ERROR_EMPTY_STRING,

```

```

    G_SHELL_ERROR_FAILED
} GShellError;
typedef short int gshort;
typedef struct _GMappedFile GMappedFile;
typedef enum {
    G_BOOKMARK_FILE_ERROR_INVALID_URI,
    G_BOOKMARK_FILE_ERROR_INVALID_VALUE,
    G_BOOKMARK_FILE_ERROR_APP_NOT_REGISTERED,
    G_BOOKMARK_FILE_ERROR_URI_NOT_FOUND,
    G_BOOKMARK_FILE_ERROR_READ,
    G_BOOKMARK_FILE_ERROR_UNKNOWN_ENCODING,
    G_BOOKMARK_FILE_ERROR_WRITE,
    G_BOOKMARK_FILE_ERROR_FILE_NOT_FOUND
} GBookmarkFileError;
typedef struct _GBookmarkFile GBookmarkFile;
typedef struct _GSourcePrivate GSourcePrivate;
typedef struct GTestCase GTestCase;
typedef struct GTestSuite GTestSuite;
typedef void (*GTestFunc) (void);
typedef void (*GTestDataFunc) (gconstpointer);
typedef void (*GTestFixtureFunc) (gconstpointer);
typedef enum {
    G_TEST_TRAP_SILENCE_STDOUT = 1 << 7,
    G_TEST_TRAP_SILENCE_STDERR = 1 << 8,
    G_TEST_TRAP_INHERIT_STDIN = 1 << 9
} GTestTrapFlags;
typedef struct {
    gboolean test_initialized;
    gboolean test_quick;
    gboolean test_perf;
    gboolean test_verbose;
    gboolean test_quiet;
    gboolean test_undefined;
} GTestConfig;
typedef enum {
    G_TEST_LOG_NONE,
    G_TEST_LOG_ERROR,
    G_TEST_LOG_START_BINARY,
    G_TEST_LOG_LIST_CASE,
    G_TEST_LOG_SKIP_CASE,
    G_TEST_LOG_START_CASE,
    G_TEST_LOG_STOP_CASE,
    G_TEST_LOG_MIN_RESULT,
    G_TEST_LOG_MAX_RESULT,
    G_TEST_LOG_MESSAGE
} GTestLogType;
typedef struct {
    GTestLogType log_type;
    guint n_strings;
    gchar **strings;
    guint n_nums;
    long double *nums;
} GTestLogMsg;
typedef struct {
    GString *data;
    GSList *msgs;
} GTestLogBuffer;
typedef gboolean (*GTestLogFatalFunc) (const gchar *, GLogLevelFlags,
                                       const gchar *, gpointer);

typedef enum {
    G_USER_DIRECTORY_DESKTOP,
    G_USER_DIRECTORY_DOCUMENTS,
    G_USER_DIRECTORY_DOWNLOAD,
    G_USER_DIRECTORY_MUSIC,
    G_USER_DIRECTORY_PICTURES,
    G_USER_DIRECTORY_PUBLIC_SHARE,

```

```

        G_USER_DIRECTORY_TEMPLATES,
        G_USER_DIRECTORY_VIDEOS,
        G_USER_N_DIRECTORIES
    } GUserDirectory;
typedef enum {
    G_FORMAT_SIZE_DEFAULT,
    G_FORMAT_SIZE_LONG_FORMAT,
    G_FORMAT_SIZE_IEC_UNITS
} GFormatSizeFlags;
typedef struct _GSequence GSequence;
typedef struct _GSequenceNode GSequenceIter;
typedef gint(*GSequenceIterCompareFunc) (void *);
typedef enum {
    G_MARKUP_COLLECT_INVALID,
    G_MARKUP_COLLECT_STRING,
    G_MARKUP_COLLECT_STRDUP,
    G_MARKUP_COLLECT_BOOLEAN,
    G_MARKUP_COLLECT_TRISTATE,
    G_MARKUP_COLLECT_OPTIONAL
} GMarkupCollectType;
typedef struct _GHashTableIter {
    gpointer dummy1;
    gpointer dummy2;
    gpointer dummy3;
    int dummy4;
    gboolean dummy5;
    gpointer dummy6;
} GHashTableIter;
typedef enum {
    G_UNICODE_SCRIPT_INVALID_CODE,
    G_UNICODE_SCRIPT_COMMON,
    G_UNICODE_SCRIPT_INHERITED,
    G_UNICODE_SCRIPT_ARABIC,
    G_UNICODE_SCRIPT_ARMENIAN,
    G_UNICODE_SCRIPT_BENGALI,
    G_UNICODE_SCRIPT_BOPOMOFO,
    G_UNICODE_SCRIPT_CHEROKEE,
    G_UNICODE_SCRIPT_COPTIC,
    G_UNICODE_SCRIPT_CYRILLIC,
    G_UNICODE_SCRIPT_DESERET,
    G_UNICODE_SCRIPT_DEVANAGARI,
    G_UNICODE_SCRIPT_ETHIOPIA,
    G_UNICODE_SCRIPT_GEORGIAN,
    G_UNICODE_SCRIPT_GOTHIC,
    G_UNICODE_SCRIPT_GREEK,
    G_UNICODE_SCRIPT_GUJARATI,
    G_UNICODE_SCRIPT_GURMUKHI,
    G_UNICODE_SCRIPT_HAN,
    G_UNICODE_SCRIPT_HANGUL,
    G_UNICODE_SCRIPT_HEBREW,
    G_UNICODE_SCRIPT_HIRAGANA,
    G_UNICODE_SCRIPT_KANNADA,
    G_UNICODE_SCRIPT_KATAKANA,
    G_UNICODE_SCRIPT_KHMER,
    G_UNICODE_SCRIPT_LAO,
    G_UNICODE_SCRIPT_LATIN,
    G_UNICODE_SCRIPT_MALAYALAM,
    G_UNICODE_SCRIPT_MONGOLIAN,
    G_UNICODE_SCRIPT_Myanmar,
    G_UNICODE_SCRIPT_OGHAM,
    G_UNICODE_SCRIPT_OLD_ITALIC,
    G_UNICODE_SCRIPT_ORIYA,
    G_UNICODE_SCRIPT_RUNIC,
    G_UNICODE_SCRIPT_SINHALA,
    G_UNICODE_SCRIPT_SYRIAC,
    G_UNICODE_SCRIPT_TAMIL,

```


G_UNICODE_SCRIPT_TELUGU,
 G_UNICODE_SCRIPT_THAANA,
 G_UNICODE_SCRIPT_THAI,
 G_UNICODE_SCRIPT_TIBETAN,
 G_UNICODE_SCRIPT_CANADIAN_ABORIGINAL,
 G_UNICODE_SCRIPT_YI,
 G_UNICODE_SCRIPT_TAGALOG,
 G_UNICODE_SCRIPT_HANUNOO,
 G_UNICODE_SCRIPT_BUHID,
 G_UNICODE_SCRIPT_TAGBANWA,
 G_UNICODE_SCRIPT_BRaille,
 G_UNICODE_SCRIPT_CYPRIOT,
 G_UNICODE_SCRIPT_LIMBU,
 G_UNICODE_SCRIPT_OSMANYA,
 G_UNICODE_SCRIPT_SHAVIAN,
 G_UNICODE_SCRIPT_LINEAR_B,
 G_UNICODE_SCRIPT_TAI_LE,
 G_UNICODE_SCRIPT_UGARITIC,
 G_UNICODE_SCRIPT_NEW_TAI_LUE,
 G_UNICODE_SCRIPT_BUGINESE,
 G_UNICODE_SCRIPT_GLAGOLITIC,
 G_UNICODE_SCRIPT_TIFINAGH,
 G_UNICODE_SCRIPT_SYLOTI_NAGRI,
 G_UNICODE_SCRIPT_OLD_PERSIAN,
 G_UNICODE_SCRIPT_KHAROSHTHI,
 G_UNICODE_SCRIPT_UNKNOWN,
 G_UNICODE_SCRIPT_BALINESE,
 G_UNICODE_SCRIPT_CUNEIFORM,
 G_UNICODE_SCRIPT_PHOENICIAN,
 G_UNICODE_SCRIPT_PHAGS_PA,
 G_UNICODE_SCRIPT_NKO,
 G_UNICODE_SCRIPT_KAYAH_LI,
 G_UNICODE_SCRIPT_LEPCHA,
 G_UNICODE_SCRIPT_REJANG,
 G_UNICODE_SCRIPT_SUNDANESE,
 G_UNICODE_SCRIPT_SAURASHTRA,
 G_UNICODE_SCRIPT_CHAM,
 G_UNICODE_SCRIPT_OL_CHIKI,
 G_UNICODE_SCRIPT_VAI,
 G_UNICODE_SCRIPT_GARIAN,
 G_UNICODE_SCRIPT_LYCIAN,
 G_UNICODE_SCRIPT_LYDIAN,
 G_UNICODE_SCRIPT_AVESTAN,
 G_UNICODE_SCRIPT_BAMUM,
 G_UNICODE_SCRIPT_EGYPTIAN_HIEROGLYPHS,
 G_UNICODE_SCRIPT_IMPERIAL_ARAMAIC,
 G_UNICODE_SCRIPT_INSCRIPTIONAL_PAHLAVI,
 G_UNICODE_SCRIPT_INSCRIPTIONAL_PARTHIAN,
 G_UNICODE_SCRIPT_JAVANESE,
 G_UNICODE_SCRIPT_KAITHI,
 G_UNICODE_SCRIPT_LISU,
 G_UNICODE_SCRIPT_MEETEI_MAYEK,
 G_UNICODE_SCRIPT_OLD_SOUTH_ARABIAN,
 G_UNICODE_SCRIPT_OLD_TURKIC,
 G_UNICODE_SCRIPT_SAMARITAN,
 G_UNICODE_SCRIPT_TAI_THAM,
 G_UNICODE_SCRIPT_TAI_VIET,
 G_UNICODE_SCRIPT_BATAK,
 G_UNICODE_SCRIPT_BRAHMI,
 G_UNICODE_SCRIPT_MANDAIC,
 G_UNICODE_SCRIPT_CHAKMA,
 G_UNICODE_SCRIPT_MEROITIC_CURSIVE,
 G_UNICODE_SCRIPT_MEROITIC_HIEROGLYPHS,
 G_UNICODE_SCRIPT_MIAO,
 G_UNICODE_SCRIPT_SHARADA,
 G_UNICODE_SCRIPT_SORA_SOMPENG,

```

    G_UNICODE_SCRIPT_TAKRI
} GUnicodeScript;
typedef struct _GTimeZone GTimeZone;
typedef enum {
    G_TIME_TYPE_STANDARD,
    G_TIME_TYPE_DAYLIGHT,
    G_TIME_TYPE_UNIVERSAL
} GTimeType;
typedef struct _GBytes GBytes;
typedef struct _GVariantType GVariantType;
typedef enum {
    G_CHECKSUM_MD5,
    G_CHECKSUM_SHA1,
    G_CHECKSUM_SHA256
} GChecksumType;
typedef struct _GChecksum GChecksum;
typedef union _GDoubleIEEE754 {
    gdouble v_double;
    struct {
        guint mantissa_low:32;
        guint mantissa_high:20;
        guint biased_exponent:11;
        guint sign:1;
    } mpn;
} GDoubleIEEE754;
typedef union _GFloatIEEE754 {
    gfloat v_float;
    struct {
        guint mantissa:23;
        guint biased_exponent:8;
        guint sign:1;
    } mpn;
} GFloatIEEE754;
union _GFloatIEEE754 {
    gfloat v_float;
    struct {
        guint mantissa:23;
        guint biased_exponent:8;
        guint sign:1;
    } mpn;
};
union _GDoubleIEEE754 {
    gdouble v_double;
    struct {
        guint mantissa_low:32;
        guint mantissa_high:20;
        guint biased_exponent:11;
        guint sign:1;
    } mpn;
};
typedef struct _GHmac GHmac;
typedef struct _GVariant GVariant;
typedef enum {
    G_VARIANT_CLASS_BOOLEAN,
    G_VARIANT_CLASS_BYTE,
    G_VARIANT_CLASS_INT16,
    G_VARIANT_CLASS_UINT16,
    G_VARIANT_CLASS_INT32,
    G_VARIANT_CLASS_UINT32,
    G_VARIANT_CLASS_INT64,
    G_VARIANT_CLASS_UINT64,
    G_VARIANT_CLASS_HANDLE,
    G_VARIANT_CLASS_DOUBLE,
    G_VARIANT_CLASS_STRING,
    G_VARIANT_CLASS_OBJECT_PATH,
    G_VARIANT_CLASS_SIGNATURE,

```

```

    G_VARIANT_CLASS_VARIANT,
    G_VARIANT_CLASS_MAYBE,
    G_VARIANT_CLASS_ARRAY,
    G_VARIANT_CLASS_TUPLE,
    G_VARIANT_CLASS_DICT_ENTRY
} GVariantClass;
typedef struct _GVariantIter {
    gsize x[16];
} GVariantIter;
typedef struct _GVariantBuilder {
    gsize x[16];
} GVariantBuilder;
typedef enum {
    G_VARIANT_PARSE_ERROR_FAILED,
    G_VARIANT_PARSE_ERROR_BASIC_TYPE_EXPECTED,
    G_VARIANT_PARSE_ERROR_CANNOT_INFER_TYPE,
    G_VARIANT_PARSE_ERROR_DEFINITE_TYPE_EXPECTED,
    G_VARIANT_PARSE_ERROR_INPUT_NOT_AT_END,
    G_VARIANT_PARSE_ERROR_INVALID_CHARACTER,
    G_VARIANT_PARSE_ERROR_INVALID_FORMAT_STRING,
    G_VARIANT_PARSE_ERROR_INVALID_OBJECT_PATH,
    G_VARIANT_PARSE_ERROR_INVALID_SIGNATURE,
    G_VARIANT_PARSE_ERROR_INVALID_TYPE_STRING,
    G_VARIANT_PARSE_ERROR_NO_COMMON_TYPE,
    G_VARIANT_PARSE_ERROR_NUMBER_OUT_OF_RANGE,
    G_VARIANT_PARSE_ERROR_NUMBER_TOO_BIG,
    G_VARIANT_PARSE_ERROR_TYPE_ERROR,
    G_VARIANT_PARSE_ERROR_UNEXPECTED_TOKEN,
    G_VARIANT_PARSE_ERROR_UNKNOWN_KEYWORD,
    G_VARIANT_PARSE_ERROR_UNTERMINATED_STRING_CONSTANT,
    G_VARIANT_PARSE_ERROR_VALUE_EXPECTED
} GVariantParseError;
typedef enum {
    G_REGEX_ERROR_COMPILE,
    G_REGEX_ERROR_OPTIMIZE,
    G_REGEX_ERROR_REPLACE,
    G_REGEX_ERROR_MATCH,
    G_REGEX_ERROR_INTERNAL,
    G_REGEX_ERROR_STRAY_BACKSLASH,
    G_REGEX_ERROR_MISSING_CONTROL_CHAR,
    G_REGEX_ERROR_UNRECOGNIZED_ESCAPE,
    G_REGEX_ERROR_QUANTIFIERS_OUT_OF_ORDER,
    G_REGEX_ERROR_QUANTIFIER_TOO_BIG,
    G_REGEX_ERROR_UNTERMINATED_CHARACTER_CLASS,
    G_REGEX_ERROR_INVALID_ESCAPE_IN_CHARACTER_CLASS,
    G_REGEX_ERROR_RANGE_OUT_OF_ORDER,
    G_REGEX_ERROR_NOTHING_TO_REPEAT,
    G_REGEX_ERROR_UNRECOGNIZED_CHARACTER,
    G_REGEX_ERROR_POSIX_NAMED_CLASS_OUTSIDE_CLASS,
    G_REGEX_ERROR_UNMATCHED_PARENTHESIS,
    G_REGEX_ERROR_INEXISTENT_SUBPATTERN_REFERENCE,
    G_REGEX_ERROR_UNTERMINATED_COMMENT,
    G_REGEX_ERROR_EXPRESSION_TOO_LARGE,
    G_REGEX_ERROR_MEMORY_ERROR,
    G_REGEX_ERROR_VARIABLE_LENGTH_LOOKBEHIND,
    G_REGEX_ERROR_MALFORMED_CONDITION,
    G_REGEX_ERROR_TOO_MANY_CONDITIONAL_BRANCHES,
    G_REGEX_ERROR_ASSERTION_EXPECTED,
    G_REGEX_ERROR_UNKNOWN_POSIX_CLASS_NAME,
    G_REGEX_ERROR_POSIX_COLLATING_ELEMENTS_NOT_SUPPORTED,
    G_REGEX_ERROR_HEX_CODE_TOO_LARGE,
    G_REGEX_ERROR_INVALID_CONDITION,
    G_REGEX_ERROR_SINGLE_BYTE_MATCH_IN_LOOKBEHIND,
    G_REGEX_ERROR_INFINITE_LOOP,
    G_REGEX_ERROR_MISSING_SUBPATTERN_NAME_TERMINATOR,
    G_REGEX_ERROR_DUPLICATE_SUBPATTERN_NAME,

```

```

    G_REGEX_ERROR_MALFORMED_PROPERTY,
    G_REGEX_ERROR_UNKNOWN_PROPERTY,
    G_REGEX_ERROR_SUBPATTERN_NAME_TOO_LONG,
    G_REGEX_ERROR_TOO_MANY_SUBPATTERNS,
    G_REGEX_ERROR_INVALID_OCTAL_VALUE,
    G_REGEX_ERROR_TOO_MANY_BRANCHES_IN_DEFINE,
    G_REGEX_ERROR_DEFINE_REPETITION,
    G_REGEX_ERROR_INCONSISTENT_NEWLINE_OPTIONS,
    G_REGEX_ERROR_MISSING_BACK_REFERENCE
} GRegexError;
typedef enum {
    G_REGEX_CASELESS,
    G_REGEX_MULTILINE,
    G_REGEX_DOTALL,
    G_REGEX_EXTENDED,
    G_REGEX_ANCHORED,
    G_REGEX_DOLLAR_ENDONLY,
    G_REGEX_UNGREEDY,
    G_REGEX_RAW,
    G_REGEX_NO_AUTO_CAPTURE,
    G_REGEX_OPTIMIZE,
    G_REGEX_DUPNAMES,
    G_REGEX_NEWLINE_CR,
    G_REGEX_NEWLINE_LF,
    G_REGEX_NEWLINE_CRLF
} GRegexCompileFlags;
typedef enum {
    G_REGEX_MATCH_ANCHORED,
    G_REGEX_MATCH_NOTBOL,
    G_REGEX_MATCH_NOTEOL,
    G_REGEX_MATCH_NOTEMPTY,
    G_REGEX_MATCH_PARTIAL,
    G_REGEX_MATCH_NEWLINE_CR,
    G_REGEX_MATCH_NEWLINE_LF,
    G_REGEX_MATCH_NEWLINE_CRLF,
    G_REGEX_MATCH_NEWLINE_ANY
} GRegexMatchFlags;
typedef struct _GRegex GRegex;
typedef struct _GMatchInfo GMatchInfo;
typedef gboolean (*GRegexEvalCallback) (const GMatchInfo *, GString
*,
                                     gpointer);

typedef struct _GRecMutex {
    gpointer p;
    guint i[2];
} GRecMutex;
typedef struct _GRWLock {
    gpointer p;
    guint i[2];
} GRWLock;
union _GMutex {
    gpointer p;
    guint i[2];
};
typedef gint64 GTimeSpan;
typedef struct _GDateTime GDateTime;

typedef struct _GStaticRecMutex GStaticRecMutex;
typedef struct _GStaticRWLock GStaticRWLock;
extern void g_allocator_free(GAllocator * allocator);
extern GAllocator *g_allocator_new(const gchar * name, guint
n_preallocs);
extern GArray *g_array_append_vals(GArray * array, gconstpointer
data,
                                   guint len);
extern gchar *g_array_free(GArray * array, gboolean free_segment);

```

```

extern quint g_array_get_element_size(GArray * array);
extern GArray *g_array_insert_vals(GArray * array, quint index_,
                                   gconstpointer data, quint len);
extern GArray *g_array_new(gboolean zero_terminated, gboolean
clear_,
                           quint element_size);
extern GArray *g_array_prepend_vals(GArray * array, gconstpointer
data,
                                   quint len);
extern GArray *g_array_ref(GArray * array);
extern GArray *g_array_remove_index(GArray * array, quint index_);
extern GArray *g_array_remove_index_fast(GArray * array, quint
index_);
extern GArray *g_array_remove_range(GArray * array, quint index_,
                                   quint length);
extern void g_array_set_clear_func(GArray * array,
                                   GDestroyNotify clear_func);
extern GArray *g_array_set_size(GArray * array, quint length);
extern GArray *g_array_sized_new(gboolean zero_terminated,
gboolean clear_,
                                   quint element_size, quint
reserved_size);
extern void g_array_sort(GArray * array, GCompareFunc compare_func);
extern void g_array_sort_with_data(GArray * array,
                                   GCompareDataFunc compare_func,
                                   gpointer user_data);
extern void g_array_unref(GArray * array);
extern gint g_ascii_digit_value(gchar c);
extern gchar *g_ascii_dtostr(gchar * buffer, gint buf_len, gdouble
d);
extern gchar *g_ascii_formatd(gchar * buffer, gint buf_len,
                              const gchar * format, gdouble d);
extern gint g_ascii_strcasecmp(const gchar * s1, const gchar * s2);
extern gchar *g_ascii_strdown(const gchar * str, gssize len);
extern gint g_ascii_strncasecmp(const gchar * s1, const gchar * s2,
                                gsize n);
extern gdouble g_ascii_strtod(const gchar * nptr, gchar * *endptr);
extern gint64 g_ascii_strtoll(const gchar * nptr, gchar * *endptr,
                              quint base);
extern quint64 g_ascii_strtoull(const gchar * nptr, gchar * *endptr,
                                quint base);
extern gchar *g_ascii_strup(const gchar * str, gssize len);
extern const guint16 *const g_ascii_table;
extern gchar g_ascii_tolower(gchar c);
extern gchar g_ascii_toupper(gchar c);
extern gint g_ascii_xdigit_value(gchar c);
extern void g_assert_warning(const char *log_domain, const char
*file,
                             const int line, const char
*pretty_function,
                             const char *expression);
extern void g_assertion_message(const char *domain, const char
*file,
                               int line, const char *func,
                               const char *message);
extern void g_assertion_message_cmpnum(const char *domain,
                                       const char *file, int line,
                                       const char *func, const char *expr,
                                       long double arg1, const char *cmp,
                                       long double arg2, char numtype);
extern void g_assertion_message_cmpstr(const char *domain,
                                       const char *file, int line,
                                       const char *func, const char *expr,
                                       const char *arg1, const char *cmp,
                                       const char *arg2);

```

```

extern void g_assertion_message_error(const char *domain, const
char *file,
                                int line, const char *func,
                                const char *expr,
                                const GError * error,
                                GQuark      error_domain,      int
error_code);
extern void g_assertion_message_expr(const char *domain, const char
*file,
                                int line, const char *func,
                                const char *expr);
extern gint g_async_queue_length(GAsyncQueue * queue);
extern gint g_async_queue_length_unlocked(GAsyncQueue * queue);
extern void g_async_queue_lock(GAsyncQueue * queue);
extern GAsyncQueue *g_async_queue_new(void);
extern GAsyncQueue *g_async_queue_new_full(GDestroyNotify
item_free_func);
extern gpointer g_async_queue_pop(GAsyncQueue * queue);
extern gpointer g_async_queue_pop_unlocked(GAsyncQueue * queue);
extern void g_async_queue_push(GAsyncQueue * queue, gpointer data);
extern void g_async_queue_push_sorted(GAsyncQueue * queue, gpointer
data,
                                GCompareDataFunc func,
                                gpointer user_data);
extern void g_async_queue_push_sorted_unlocked(GAsyncQueue * queue,
gpointer data,
                                GCompareDataFunc func,
                                gpointer user_data);
extern void g_async_queue_push_unlocked(GAsyncQueue * queue,
gpointer data);
extern GAsyncQueue *g_async_queue_ref(GAsyncQueue * queue);
extern void g_async_queue_ref_unlocked(GAsyncQueue * queue);
extern void g_async_queue_sort(GAsyncQueue * queue,
GCompareDataFunc func,
                                gpointer user_data);
extern void g_async_queue_sort_unlocked(GAsyncQueue * queue,
GCompareDataFunc func,
                                gpointer user_data);
extern gpointer g_async_queue_timed_pop(GAsyncQueue * queue,
GTimeVal * end_time);
extern gpointer g_async_queue_timed_pop_unlocked(GAsyncQueue *
queue,
                                GTimeVal * end_time);
extern gpointer g_async_queue_timeout_pop(GAsyncQueue * queue,
guint64 timeout);
extern gpointer g_async_queue_timeout_pop_unlocked(GAsyncQueue *
queue,
                                guint64 timeout);
extern gpointer g_async_queue_try_pop(GAsyncQueue * queue);
extern gpointer g_async_queue_try_pop_unlocked(GAsyncQueue *
queue);
extern void g_async_queue_unlock(GAsyncQueue * queue);
extern void g_async_queue_unref(GAsyncQueue * queue);
extern void g_async_queue_unref_and_unlock(GAsyncQueue * queue);
extern void g_atexit(GVoidFunc func);
extern gint g_atomic_int_add(volatile gint * atomic, gint val);
extern guint g_atomic_int_and(volatile guint * atomic, guint val);
extern gboolean g_atomic_int_compare_and_exchange(volatile gint *
atomic,
                                gint oldval,
                                gint newval);
extern gboolean g_atomic_int_dec_and_test(volatile int *atomic);
extern gint g_atomic_int_exchange_and_add(volatile gint * atomic,
gint val);
extern gint g_atomic_int_get(volatile int *atomic);
extern void g_atomic_int_inc(volatile int *atomic);

```

```

extern guint g_atomic_int_or(volatile guint * atomic, guint val);
extern void g_atomic_int_set(volatile int *atomic, gint newval);
extern guint g_atomic_int_xor(volatile guint * atomic, guint val);
extern gssize g_atomic_pointer_add(volatile void *atomic, gssize
val);
extern gsize g_atomic_pointer_and(volatile void *atomic, gsize val);
extern gboolean g_atomic_pointer_compare_and_exchange(volatile
void
                                *atomic,
                                gpointer oldval,
                                gpointer newval);
extern void *g_atomic_pointer_get(volatile void *atomic);
extern gsize g_atomic_pointer_or(volatile void *atomic, gsize val);
extern void g_atomic_pointer_set(volatile void *atomic, void
*newval);
extern gsize g_atomic_pointer_xor(volatile void *atomic, gsize val);
extern guchar *g_base64_decode(const gchar * text, gsize * out_len);
extern guchar *g_base64_decode_inplace(gchar *, gsize *);
extern gsize g_base64_decode_step(const gchar * in, gsize len,
                                guchar * out, gint * state,
                                guint * save);
extern gchar *g_base64_encode(const unsigned char *data, gsize len);
extern gsize g_base64_encode_close(gboolean break_lines, gchar *
out,
                                gint * state, gint * save);
extern gsize g_base64_encode_step(const unsigned char *in, gsize
len,
                                gboolean break_lines, gchar * out,
                                gint * state, gint * save);
extern const char *g_basename(const char *file_name);
extern void g_bit_lock(volatile int *address, gint lock_bit);
extern gint g_bit_nth_lsf(gulong mask, gint nth_bit);
extern gint g_bit_nth_msf(gulong mask, gint nth_bit);
extern guint g_bit_storage(gulong number);
extern gboolean g_bit_trylock(volatile int *address, gint lock_bit);
extern void g_bit_unlock(volatile int *address, gint lock_bit);
extern void g_blow_chunks(void);
extern void g_bookmark_file_add_application(GBookmarkFile *
bookmark,
                                const char *uri,
                                const char *name,
                                const char *exec);
extern void g_bookmark_file_add_group(GBookmarkFile * bookmark,
                                const char *uri, const char *group);
extern GQuark g_bookmark_file_error_quark(void);
extern void g_bookmark_file_free(GBookmarkFile * bookmark);
extern time_t g_bookmark_file_get_added(GBookmarkFile * bookmark,
                                const char *uri, GError * *error);
extern gboolean g_bookmark_file_get_app_info(GBookmarkFile *
bookmark,
                                const char *uri,
                                const char *name,
                                gchar * *exec, guint * count,
                                time_t * stamp,
                                GError * *error);
extern gchar **g_bookmark_file_get_applications(GBookmarkFile *
bookmark,
                                const char *uri,
                                gsize * length,
                                GError * *error);
extern gchar *g_bookmark_file_get_description(GBookmarkFile *
bookmark,
                                const char *uri,
                                GError * *error);
extern gchar **g_bookmark_file_get_groups(GBookmarkFile * bookmark,
                                const char *uri, gsize * length,

```

```

        GError * *error);
extern gboolean g_bookmark_file_get_icon(GBookmarkFile * bookmark,
        const char *uri, gchar * *href,
        gchar * *mime_type,
        GError * *error);
extern gboolean g_bookmark_file_get_is_private(GBookmarkFile *
bookmark,
        const char *uri,
        GError * *error);
extern gchar *g_bookmark_file_get_mime_type(GBookmarkFile *
bookmark,
        const char *uri,
        GError * *error);
extern time_t g_bookmark_file_get_modified(GBookmarkFile *
bookmark,
        const char *uri,
        GError * *error);
extern gint g_bookmark_file_get_size(GBookmarkFile * bookmark);
extern gchar *g_bookmark_file_get_title(GBookmarkFile * bookmark,
        const char *uri, GError * *error);
extern gchar **g_bookmark_file_get_uris(GBookmarkFile * bookmark,
        gsize * length);
extern time_t g_bookmark_file_get_visited(GBookmarkFile * bookmark,
        const char *uri,
        GError * *error);
extern gboolean g_bookmark_file_has_application(GBookmarkFile *
bookmark,
        const char *uri,
        const char *name,
        GError * *error);
extern gboolean g_bookmark_file_has_group(GBookmarkFile * bookmark,
        const char *uri,
        const char *group,
        GError * *error);
extern gboolean g_bookmark_file_has_item(GBookmarkFile * bookmark,
        const char *uri);
extern gboolean g_bookmark_file_load_from_data(GBookmarkFile *
bookmark,
        const char *data,
        gsize length,
        GError * *error);
extern gboolean g_bookmark_file_load_from_data_dirs(GBookmarkFile
*
        bookmark,
        const char *file,
        gchar * *full_path,
        GError * *error);
extern gboolean g_bookmark_file_load_from_file(GBookmarkFile *
bookmark,
        const char *filename,
        GError * *error);
extern gboolean g_bookmark_file_move_item(GBookmarkFile * bookmark,
        const char *old_uri,
        const char *new_uri,
        GError * *error);
extern GBookmarkFile *g_bookmark_file_new(void);
extern gboolean g_bookmark_file_remove_application(GBookmarkFile *
        bookmark,
        const char *uri,
        const char *name,
        GError * *error);
extern gboolean g_bookmark_file_remove_group(GBookmarkFile *
bookmark,
        const char *uri,
        const char *group,
        GError * *error);

```



```

extern gboolean g_bookmark_file_remove_item(GBookmarkFile *
bookmark,
const char *uri,
GError * *error);
extern void g_bookmark_file_set_added(GBookmarkFile * bookmark,
const char *uri, time_t added);
extern gboolean g_bookmark_file_set_app_info(GBookmarkFile *
bookmark,
const char *uri,
const char *name,
const char *exec, gint count,
time_t stamp,
GError * *error);
extern void g_bookmark_file_set_description(GBookmarkFile *
bookmark,
const char *uri,
const char *description);
extern void g_bookmark_file_set_groups(GBookmarkFile * bookmark,
const char *uri,
const char **groups, gsize length);
extern void g_bookmark_file_set_icon(GBookmarkFile * bookmark,
const char *uri, const char *href,
const char *mime_type);
extern void g_bookmark_file_set_is_private(GBookmarkFile *
bookmark,
const char *uri,
gboolean is_private);
extern void g_bookmark_file_set_mime_type(GBookmarkFile * bookmark,
const char *uri,
const char *mime_type);
extern void g_bookmark_file_set_modified(GBookmarkFile * bookmark,
const char *uri, time_t
modified);
extern void g_bookmark_file_set_title(GBookmarkFile * bookmark,
const char *uri, const char *title);
extern void g_bookmark_file_set_visited(GBookmarkFile * bookmark,
const char *uri, time_t visited);
extern gchar *g_bookmark_file_to_data(GBookmarkFile * bookmark,
gsize * length, GError * *error);
extern gboolean g_bookmark_file_to_file(GBookmarkFile * bookmark,
const char *filename,
GError * *error);
extern gchar *g_build_filename(const gchar * first_element, ...);
extern gchar *g_build_filenamev(gchar * *args);
extern gchar *g_build_path(const gchar * separator,
const gchar * first_element, ...);
extern gchar *g_build_pathv(const gchar * separator, gchar * *args);
extern GByteArray *g_byte_array_append(GByteArray * array,
const guint8 * data, guint len);
extern guint8 *g_byte_array_free(GByteArray * array,
gboolean free_segment);
extern GBytes *g_byte_array_free_to_bytes(GByteArray * array);
extern GByteArray *g_byte_array_new(void);
extern GByteArray *g_byte_array_new_take(guint8 * data, gsize len);
extern GByteArray *g_byte_array_prepend(GByteArray * array,
const guint8 * data, guint len);
extern GByteArray *g_byte_array_ref(GByteArray * array);
extern GByteArray *g_byte_array_remove_index(GByteArray * array,
guint index);
extern GByteArray *g_byte_array_remove_index_fast(GByteArray *
array,
guint index);
extern GByteArray *g_byte_array_remove_range(GByteArray * array,
guint index, guint length);
extern GByteArray *g_byte_array_set_size(GByteArray * array, guint
length);

```

```

extern GByteArray *g_byte_array_sized_new(guint reserved_size);
extern void g_byte_array_sort(GByteArray * array,
                             GCompareFunc compare_func);
extern void g_byte_array_sort_with_data(GByteArray * array,
                                       GCompareDataFunc compare_func,
                                       gpointer user_data);
extern void g_byte_array_unref(GByteArray * array);
extern gint g_bytes_compare(gconstpointer bytes1, gconstpointer
bytes2);
extern gboolean g_bytes_equal(gconstpointer bytes1, gconstpointer
bytes2);
extern gconstpointer g_bytes_get_data(GBytes * bytes, gsize * size);
extern gsize g_bytes_get_size(GBytes * bytes);
extern guint g_bytes_hash(gconstpointer bytes);
extern GBytes *g_bytes_new(gconstpointer data, gsize size);
extern GBytes *g_bytes_new_from_bytes(GBytes * bytes, gsize offset,
                                     gsize length);
extern GBytes *g_bytes_new_static(gconstpointer data, gsize size);
extern GBytes *g_bytes_new_take(void *data, gsize size);
extern GBytes *g_bytes_new_with_free_func(gconstpointer data,
                                     gsize size,
                                     GDestroyNotify free_func,
                                     void *user_data);
extern GBytes *g_bytes_ref(GBytes * bytes);
extern void g_bytes_unref(GBytes * bytes);
extern GByteArray *g_bytes_unref_to_array(GBytes * bytes);
extern void *g_bytes_unref_to_data(GBytes * bytes, gsize * size);
extern void g_cache_destroy(GCache * cache);
extern gpointer g_cache_insert(GCache * cache, gpointer key);
extern void g_cache_key_foreach(GCache * cache, GHFunc func,
                               gpointer user_data);
extern GCache *g_cache_new(GCacheNewFunc value_new_func,
                          GCacheDestroyFunc value_destroy_func,
                          GCacheDupFunc key_dup_func,
                          GCacheDestroyFunc key_destroy_func,
                          GHashFunc hash_key_func,
                          GHashFunc hash_value_func,
                          GEqualFunc key_equal_func);
extern void g_cache_remove(GCache * cache, gconstpointer value);
extern void g_cache_value_foreach(GCache * cache, GHFunc func,
                                  gpointer user_data);
extern GChecksum *g_checksum_copy(const GChecksum * checksum);
extern void g_checksum_free(GChecksum * checksum);
extern void g_checksum_get_digest(GChecksum * checksum, guint8 *
buffer,
                                gsize * digest_len);
extern const gchar *g_checksum_get_string(GChecksum * checksum);
extern GChecksum *g_checksum_new(GChecksumType checksum_type);
extern void g_checksum_reset(GChecksum * checksum);
extern gssize g_checksum_type_get_length(GChecksumType
checksum_type);
extern void g_checksum_update(GChecksum * checksum,
                             const unsigned char *data, gsize length);
extern guint g_child_watch_add(GPid pid, GChildWatchFunc function,
                              gpointer data);
extern guint g_child_watch_add_full(gint priority, GPid pid,
                                   GChildWatchFunc function,
                                   gpointer data, GDestroyNotify
notify);
extern GSourceFuncs g_child_watch_funcs;
extern GSource *g_child_watch_source_new(GPid pid);
extern void g_clear_error(GError * *err);
extern void g_completion_add_items(GCompletion * cmp, GList *
items);
extern void g_completion_clear_items(GCompletion * cmp);
extern GList *g_completion_complete(GCompletion * cmp,

```

```

        const gchar * prefix,
        gchar * *new_prefix);
extern GList *g_completion_complete_utf8(GCompletion * cmp,
        const gchar * prefix,
        gchar * *new_prefix);
extern void g_completion_free(GCompletion * cmp);
extern GCompletion *g_completion_new(GCompletionFunc func);
extern void g_completion_remove_items(GCompletion * cmp, GList *
items);
extern void g_completion_set_compare(GCompletion * cmp,
        GCompletionStrncmpFunc
strncmp_func);
extern      gchar      *g_compute_checksum_for_data(GChecksumType
checksum_type,
        const unsigned char *data,
        gsize length);
extern      gchar      *g_compute_checksum_for_string(GChecksumType
checksum_type,
        const gchar * str,
        gssize length);
extern gchar *g_compute_hmac_for_data(GChecksumType digest_type,
        const gchar * key, gsize key_len,
        const gchar * data, gsize length);
extern gchar *g_compute_hmac_for_string(GChecksumType digest_type,
        const gchar * key, gsize key_len,
        const gchar * str, gssize length);
extern void g_cond_broadcast(GCond * cond);
extern void g_cond_clear(GCond * cond);
extern void g_cond_free(GCond * cond);
extern void g_cond_init(GCond * cond);
extern GCond *g_cond_new(void);
extern void g_cond_signal(GCond * cond);
extern gboolean g_cond_timed_wait(GCond * cond, GMutex * mutex,
        GTimeVal * timeval);
extern void g_cond_wait(GCond * cond, GMutex * mutex);
extern gboolean g_cond_wait_until(GCond * cond, GMutex * mutex,
        gint64 end_time);
extern gchar *g_convert(const gchar * str, gssize len,
        const gchar * to_codeset,
        const gchar * from_codeset, gsize * bytes_read,
        gsize * bytes_written, GError * *error);
extern GQuark g_convert_error_quark(void);
extern gchar *g_convert_with_fallback(const gchar * str, gssize len,
        const gchar * to_codeset,
        const gchar * from_codeset,
        const gchar * fallback,
        gsize * bytes_read,
        gsize * bytes_written,
        GError * *error);
extern gchar *g_convert_with_iconv(const gchar * str, gssize len,
        GIconv converter, gsize * bytes_read,
        gsize * bytes_written, GError *
*error);
extern void g_datalist_clear(GData * *datalist);
extern void g_datalist_foreach(GData * *datalist, GDataForeachFunc
func,
        gpointer user_data);
extern gpointer g_datalist_get_data(GData * *datalist, const char
*key);
extern guint g_datalist_get_flags(GData * *datalist);
extern gpointer g_datalist_id_get_data(GData * *datalist, GQuark
key_id);
extern gpointer g_datalist_id_remove_no_notify(GData * *datalist,
        GQuark key_id);
extern void g_datalist_id_set_data_full(GData * *datalist, GQuark
key_id,

```

```

        gpointer data,
        GDestroyNotify destroy_func);
extern void g_datalist_init(GData * *datalist);
extern void g_datalist_set_flags(GData * *datalist, guint flags);
extern void g_datalist_unset_flags(GData * *datalist, guint flags);
extern void g_dataset_destroy(gconstpointer dataset_location);
extern void g_dataset_foreach(gconstpointer dataset_location,
        GDataForeachFunc func, gpointer
        user_data);
extern gpointer g_dataset_id_get_data(gconstpointer
        dataset_location,
        GQuark key_id);
extern gpointer g_dataset_id_remove_no_notify(gconstpointer
        dataset_location,
        GQuark key_id);
extern void g_dataset_id_set_data_full(gconstpointer
        dataset_location,
        GQuark key_id, gpointer data,
        GDestroyNotify destroy_func);
extern void g_date_add_days(GDate * date, guint n_days);
extern void g_date_add_months(GDate * date, guint n_months);
extern void g_date_add_years(GDate * date, guint n_years);
extern void g_date_clamp(GDate * date, const GDate * min_date,
        const GDate * max_date);
extern void g_date_clear(GDate * date, guint n_dates);
extern gint g_date_compare(const GDate * lhs, const GDate * rhs);
extern gint g_date_days_between(const GDate * date1, const GDate *
        date2);
extern void g_date_free(GDate * date);
extern GDateDay g_date_get_day(const GDate * date);
extern guint g_date_get_day_of_year(const GDate * date);
extern guint8 g_date_get_days_in_month(GDateMonth month, GDateYear
        year);
extern guint g_date_get_iso8601_week_of_year(const GDate * date);
extern guint32 g_date_get_julian(const GDate * date);
extern guint g_date_get_monday_week_of_year(const GDate * date);
extern guint8 g_date_get_monday_weeks_in_year(GDateYear year);
extern GDateMonth g_date_get_month(const GDate * date);
extern guint g_date_get_sunday_week_of_year(const GDate * date);
extern guint8 g_date_get_sunday_weeks_in_year(GDateYear year);
extern GDateWeekday g_date_get_weekday(const GDate * date);
extern GDateYear g_date_get_year(const GDate * date);
extern gboolean g_date_is_first_of_month(const GDate * date);
extern gboolean g_date_is_last_of_month(const GDate * date);
extern gboolean g_date_is_leap_year(GDateYear year);
extern GDate *g_date_new(void);
extern GDate *g_date_new_dmy(GDateDay day, GDateMonth month,
        GDateYear year);
extern GDate *g_date_new_julian(guint32 julian_day);
extern void g_date_order(GDate * date1, GDate * date2);
extern void g_date_set_day(GDate * date, GDateDay day);
extern void g_date_set_dmy(GDate * date, GDateDay day, GDateMonth
        month,
        GDateYear y);
extern void g_date_set_julian(GDate * date, guint32 julian_date);
extern void g_date_set_month(GDate * date, GDateMonth month);
extern void g_date_set_parse(GDate * date, const gchar * str);
extern void g_date_set_time(GDate * date, GTime time_);
extern void g_date_set_time_t(GDate * date, time_t timet);
extern void g_date_set_time_val(GDate * date, GTimeVal * timeval);
extern void g_date_set_year(GDate * date, GDateYear year);
extern gsize g_date_strftime(gchar * s, gsize slen, const gchar *
        format,
        const GDate * date);
extern void g_date_subtract_days(GDate * date, guint n_days);
extern void g_date_subtract_months(GDate * date, guint n_months);

```

```

extern void g_date_subtract_years(GDate * date, guint n_years);
extern GDateTime *g_date_time_add(GDateTime * datetime,
                                  GTimeSpan timespan);
extern GDateTime *g_date_time_add_days(GDateTime * datetime, gint
days);
extern GDateTime *g_date_time_add_full(GDateTime * datetime, gint
years,
                                     gint months, gint days, gint hours,
                                     gint minutes, gdouble seconds);
extern GDateTime *g_date_time_add_hours(GDateTime * datetime, gint
hours);
extern GDateTime *g_date_time_add_minutes(GDateTime * datetime,
gint minutes);
extern GDateTime *g_date_time_add_months(GDateTime * datetime,
gint months);
extern GDateTime *g_date_time_add_seconds(GDateTime * datetime,
gdouble seconds);
extern GDateTime *g_date_time_add_weeks(GDateTime * datetime, gint
weeks);
extern GDateTime *g_date_time_add_years(GDateTime * datetime, gint
years);
extern gint g_date_time_compare(gconstpointer dt1, gconstpointer
dt2);
extern GTimeSpan g_date_time_difference(GDateTime * end,
GDateTime * begin);
extern gboolean g_date_time_equal(gconstpointer dt1, gconstpointer
dt2);
extern gchar *g_date_time_format(GDateTime * datetime,
const gchar * format);
extern gint g_date_time_get_day_of_month(GDateTime * datetime);
extern gint g_date_time_get_day_of_week(GDateTime * datetime);
extern gint g_date_time_get_day_of_year(GDateTime * datetime);
extern gint g_date_time_get_hour(GDateTime * datetime);
extern gint g_date_time_get_microsecond(GDateTime * datetime);
extern gint g_date_time_get_minute(GDateTime * datetime);
extern gint g_date_time_get_month(GDateTime * datetime);
extern gint g_date_time_get_second(GDateTime * datetime);
extern gdouble g_date_time_get_seconds(GDateTime * datetime);
extern const gchar
*g_date_time_get_timezone_abbreviation(GDateTime *
datetime);
extern GTimeSpan g_date_time_get_utc_offset(GDateTime * datetime);
extern gint g_date_time_get_week_numbering_year(GDateTime *
datetime);
extern gint g_date_time_get_week_of_year(GDateTime * datetime);
extern gint g_date_time_get_year(GDateTime * datetime);
extern void g_date_time_get_ymd(GDateTime * datetime, gint * year,
gint * month, gint * day);
extern guint g_date_time_hash(gconstpointer datetime);
extern gboolean g_date_time_is_daylight_savings(GDateTime *
datetime);
extern GDateTime *g_date_time_new(GTimeZone * tz, gint year, gint
month,
gint day, gint hour, gint minute,
gdouble seconds);
extern GDateTime *g_date_time_new_from_timeval_local(const
GTimeVal * tv);
extern GDateTime *g_date_time_new_from_timeval_utc(const GTimeVal
* tv);
extern GDateTime *g_date_time_new_from_unix_local(gint64 t);
extern GDateTime *g_date_time_new_from_unix_utc(gint64 t);
extern GDateTime *g_date_time_new_local(gint year, gint month, gint
day,
gint hour, gint minute,
gdouble seconds);
extern GDateTime *g_date_time_new_now(GTimeZone * tz);

```

```

extern GDateTime *g_date_time_new_now_local(void);
extern GDateTime *g_date_time_new_now_utc(void);
extern GDateTime *g_date_time_new_utc(gint year, gint month, gint
day,
                                gint hour, gint minute,
                                gdouble seconds);
extern GDateTime *g_date_time_ref(GDateTime * datetime);
extern GDateTime *g_date_time_to_local(GDateTime * datetime);
extern gboolean g_date_time_to_timeval(GDateTime * datetime,
                                GTimeVal * tv);
extern GDateTime *g_date_time_to_timezone(GDateTime * datetime,
                                GTimeZone * tz);
extern gint64 g_date_time_to_unix(GDateTime * datetime);
extern GDateTime *g_date_time_to_utc(GDateTime * datetime);
extern void g_date_time_unref(GDateTime * datetime);
extern void g_date_to_struct_tm(const GDate * date, struct tm *tm);
extern gboolean g_date_valid(const GDate * date);
extern gboolean g_date_valid_day(GDateDay day);
extern gboolean g_date_valid_dmy(GDateDay day, GDateMonth month,
                                GDateYear year);
extern gboolean g_date_valid_julian(guint32 julian_date);
extern gboolean g_date_valid_month(GDateMonth month);
extern gboolean g_date_valid_weekday(GDateWeekday weekday);
extern gboolean g_date_valid_year(GDateYear year);
extern const gchar *g_dcgettext(const gchar * domain, const gchar
* msgid,
                                gint category);
extern const gchar *g_dgettext(const gchar * domain, const gchar *
msgid);
extern void g_dir_close(GDir * dir);
extern gchar *g_dir_make_tmp(const gchar * tpl, GError * *error);
extern GDir *g_dir_open(const gchar * path, guint flags, GError *
*error);
extern const gchar *g_dir_read_name(GDir * dir);
extern void g_dir_rewind(GDir * dir);
extern gboolean g_direct_equal(gconstpointer v1, gconstpointer v2);
extern guint g_direct_hash(gconstpointer v);
extern const gchar *g_dngettext(const gchar * domain, const gchar
* msgid,
                                const gchar * msgid_plural, gulong n);
extern gboolean g_double_equal(gconstpointer v1, gconstpointer v2);
extern guint g_double_hash(gconstpointer v);
extern const gchar *g_dpgettext(const gchar * domain,
                                const gchar * msgctxtid,
                                gsize msgidoffset);
extern const gchar *g_dpgettext2(const gchar * domain,
                                const gchar * context,
                                const gchar * msgid);
extern const gchar *g_envIRON_getenv(gchar * *envp,
                                const gchar * variable);
extern gchar **g_envIRON_setenv(gchar * *envp, const gchar *
variable,
                                const gchar * value, gboolean overwrite);
extern gchar **g_envIRON_unsetenv(gchar * *envp, const gchar *
variable);
extern GError *g_error_copy(const GError * error);
extern void g_error_free(GError * error);
extern gboolean g_error_matches(const GError * error, GQuark domain,
                                gint code);
extern GError *g_error_new(GQuark domain, gint code, const gchar *
format,
                                ...);
extern GError *g_error_new_literal(GQuark domain, gint code,
                                const gchar * message);
extern GError *g_error_new_valist(GQuark domain, gint code,
                                const gchar * format, va_list args);

```

```

extern GFileError g_file_error_from_errno(gint err_no);
extern GQuark g_file_error_quark(void);
extern gboolean g_file_get_contents(const gchar * filename,
                                   gchar * *contents, gsize * length,
                                   GError * *error);
extern gint g_file_open_tmp(const gchar * tmpl, gchar * *name_used,
                           GError * *error);
extern gchar *g_file_read_link(const gchar * filename, GError *
*error);
extern gboolean g_file_set_contents(const gchar * filename,
                                   const gchar * contents, gssize length,
                                   GError * *error);
extern gboolean g_file_test(const gchar * filename, GFileTest test);
extern gchar *g_filename_display_basename(const gchar * filename);
extern gchar *g_filename_display_name(const gchar * filename);
extern gchar *g_filename_from_uri(const gchar * uri, gchar *
*hostname,
                                GError * *error);
extern gchar *g_filename_from_utf8(const gchar * utf8string, gssize
len,
                                gsize * bytes_read,
                                gsize * bytes_written, GError *
*error);
extern gchar *g_filename_to_uri(const gchar * filename,
                                const gchar * hostname, GError * *error);
extern gchar *g_filename_to_utf8(const gchar * opsysstring, gssize
len,
                                gsize * bytes_read, gsize *
bytes_written,
                                GError * *error);
extern gchar *g_find_program_in_path(const gchar * program);
extern gchar *g_format_size(guint64 size);
extern gchar *g_format_size_for_display(goffset size);
extern gchar *g_format_size_full(guint64 size, GFormatSizeFlags
flags);
extern void g_free(gpointer mem);
extern const gchar *g_get_application_name(void);
extern gboolean g_get_charset(const char **charset);
extern gchar *g_get_codeset(void);
extern gchar *g_get_current_dir(void);
extern void g_get_current_time(GTimeVal * result);
extern gchar **g_get_environ(void);
extern gboolean g_get_filename_charsets(const gchar * **charsets);
extern const gchar *g_get_home_dir(void);
extern const gchar *g_get_host_name(void);
extern const gchar *const *g_get_language_names(void);
extern gchar **g_get_locale_variants(const gchar * locale);
extern gint64 g_get_monotonic_time(void);
extern gchar *g_get_prpname(void);
extern const gchar *g_get_real_name(void);
extern gint64 g_get_real_time(void);
extern const gchar *const *g_get_system_config_dirs(void);
extern const gchar *const *g_get_system_data_dirs(void);
extern const gchar *g_get_tmp_dir(void);
extern const gchar *g_get_user_cache_dir(void);
extern const gchar *g_get_user_config_dir(void);
extern const gchar *g_get_user_data_dir(void);
extern const gchar *g_get_user_name(void);
extern const gchar *g_get_user_runtime_dir(void);
extern const gchar *g_get_user_special_dir(GUserDirectory
directory);
extern const gchar *g_getenv(const gchar * variable);
extern void g_hash_table_add(GHashTable * hash_table, void *key);
extern gboolean g_hash_table_contains(GHashTable * hash_table,
gconstpointer key);
extern void g_hash_table_destroy(GHashTable * hash_table);

```



```

extern gpointer g_hash_table_find(GHashTable * hash_table,
                                  GHRFunc predicate, gpointer user_data);
extern void g_hash_table_foreach(GHashTable * hash_table, GHRFunc
func,
                                gpointer user_data);
extern guint g_hash_table_foreach_remove(GHashTable * hash_table,
                                          GHRFunc func, gpointer
user_data);
extern guint g_hash_table_foreach_steal(GHashTable * hash_table,
                                         GHRFunc func, gpointer user_data);
extern GList *g_hash_table_get_keys(GHashTable * hash_table);
extern GList *g_hash_table_get_values(GHashTable * hash_table);
extern void g_hash_table_insert(GHashTable * hash_table, gpointer
key,
                                gpointer value);
extern GHashTable *g_hash_table_iter_get_hash_table(GHashTableIter
* iter);
extern void g_hash_table_iter_init(GHashTableIter * iter,
                                   GHashTable * hash_table);
extern gboolean g_hash_table_iter_next(GHashTableIter * iter, void
**key,
                                       void **value);
extern void g_hash_table_iter_remove(GHashTableIter * iter);
extern void g_hash_table_iter_replace(GHashTableIter * iter, void
*value);
extern void g_hash_table_iter_steal(GHashTableIter * iter);
extern gpointer g_hash_table_lookup(GHashTable * hash_table,
                                   gconstpointer key);
extern gboolean g_hash_table_lookup_extended(GHashTable *
hash_table,
                                             gconstpointer lookup_key,
                                             gpointer * orig_key,
                                             gpointer * value);
extern GHashTable *g_hash_table_new(GHashFunc hash_func,
                                   GEqualFunc key_equal_func);
extern GHashTable *g_hash_table_new_full(GHashFunc hash_func,
                                         GEqualFunc key_equal_func,
                                         GDestroyNotify key_destroy_func,
                                         GDestroyNotify
value_destroy_func);
extern GHashTable *g_hash_table_ref(GHashTable * hash_table);
extern gboolean g_hash_table_remove(GHashTable * hash_table,
                                   gconstpointer key);
extern void g_hash_table_remove_all(GHashTable * hash_table);
extern void g_hash_table_replace(GHashTable * hash_table, gpointer
key,
                                gpointer value);
extern guint g_hash_table_size(GHashTable * hash_table);
extern gboolean g_hash_table_steal(GHashTable * hash_table,
                                   gconstpointer key);
extern void g_hash_table_steal_all(GHashTable * hash_table);
extern void g_hash_table_unref(GHashTable * hash_table);
extern GHmac *g_hmac_copy(const GHmac * hmac);
extern void g_hmac_get_digest(GHmac * hmac, guint8 * buffer,
                              gsize * digest_len);
extern const gchar *g_hmac_get_string(GHmac * hmac);
extern GHmac *g_hmac_new(GChecksumType digest_type, const gchar *
key,
                        gsize key_len);
extern GHmac *g_hmac_ref(GHmac * hmac);
extern void g_hmac_unref(GHmac * hmac);
extern void g_hmac_update(GHmac * hmac, const gchar * data,
                          gssize length);
extern GHook *g_hook_alloc(GHookList * hook_list);
extern gint g_hook_compare_ids(GHook * new_hook, GHook * sibling);

```



```

extern gboolean g_hook_destroy(GHookList * hook_list, gulong
hook_id);
extern void g_hook_destroy_link(GHookList * hook_list, GHook *
hook);
extern GHook *g_hook_find(GHookList * hook_list, gboolean
need_valids,
                        GHookFindFunc func, gpointer data);
extern GHook *g_hook_find_data(GHookList * hook_list, gboolean
need_valids,
                        gpointer data);
extern GHook *g_hook_find_func(GHookList * hook_list, gboolean
need_valids,
                        gpointer func);
extern GHook *g_hook_find_func_data(GHookList * hook_list,
                        gboolean need_valids, gpointer func,
                        gpointer data);
extern GHook *g_hook_first_valid(GHookList * hook_list,
                        gboolean may_be_in_call);
extern void g_hook_free(GHookList * hook_list, GHook * hook);
extern GHook *g_hook_get(GHookList * hook_list, gulong hook_id);
extern void g_hook_insert_before(GHookList * hook_list, GHook *
sibling,
                        GHook * hook);
extern void g_hook_insert_sorted(GHookList * hook_list, GHook *
hook,
                        GHookCompareFunc func);
extern void g_hook_list_clear(GHookList * hook_list);
extern void g_hook_list_init(GHookList * hook_list, guint
hook_size);
extern void g_hook_list_invoke(GHookList * hook_list,
                        gboolean may_recurse);
extern void g_hook_list_invoke_check(GHookList * hook_list,
                        gboolean may_recurse);
extern void g_hook_list_marshall(GHookList * hook_list,
                        gboolean may_recurse,
                        GHookMarshaller marshaller,
                        gpointer marshal_data);
extern void g_hook_list_marshall_check(GHookList * hook_list,
                        gboolean may_recurse,
                        GHookCheckMarshaller marshaller,
                        gpointer marshal_data);
extern GHook *g_hook_next_valid(GHookList * hook_list, GHook * hook,
                        gboolean may_be_in_call);
extern void g_hook_prepend(GHookList * hook_list, GHook * hook);
extern GHook *g_hook_ref(GHookList * hook_list, GHook * hook);
extern void g_hook_unref(GHookList * hook_list, GHook * hook);
extern gboolean g_hostname_is_ascii_encoded(const gchar * hostname);
extern gboolean g_hostname_is_ip_address(const gchar * hostname);
extern gboolean g_hostname_is_non_ascii(const gchar * hostname);
extern gchar *g_hostname_to_ascii(const gchar * hostname);
extern gchar *g_hostname_to_unicode(const gchar * hostname);
extern gsize g_iconv(GIconv converter, gchar * *inbuf,
                        gsize * inbytes_left, gchar * *outbuf,
                        gsize * outbytes_left);
extern gint g_iconv_close(GIconv converter);
extern GIconv g_iconv_open(const gchar * to_codeset,
                        const gchar * from_codeset);
extern guint g_idle_add(GSourceFunc function, gpointer data);
extern guint g_idle_add_full(gint priority, GSourceFunc function,
                        gpointer data, GDestroyNotify notify);
extern GSourceFuncs g_idle_funcs;
extern gboolean g_idle_remove_by_data(gpointer data);
extern GSource *g_idle_source_new(void);
extern gboolean g_int64_equal(gconstpointer v1, gconstpointer v2);
extern guint g_int64_hash(gconstpointer v);
extern gboolean g_int_equal(gconstpointer v1, gconstpointer v2);

```

```

extern guint g_int_hash(gconstpointer v);
extern const gchar *g_intern_static_string(const gchar * string);
extern const gchar *g_intern_string(const gchar * string);
extern guint g_io_add_watch(GIOChannel * channel, GIOCondition
condition,
                                GIOFunc func, gpointer user_data);
extern guint g_io_add_watch_full(GIOChannel * channel, gint
priority,
                                GIOCondition condition, GIOFunc func,
                                gpointer user_data,
                                GDestroyNotify notify);
extern void g_io_channel_close(GIOChannel * channel);
extern GIOChannelError g_io_channel_error_from_errno(gint en);
extern GQuark g_io_channel_error_quark(void);
extern GIOStatus g_io_channel_flush(GIOChannel * channel, GError *
*error);
extern GIOCondition g_io_channel_get_buffer_condition(GIOChannel *
channel);
extern gsize g_io_channel_get_buffer_size(GIOChannel * channel);
extern gboolean g_io_channel_get_buffered(GIOChannel * channel);
extern gboolean g_io_channel_get_close_on_unref(GIOChannel *
channel);
extern const gchar *g_io_channel_get_encoding(GIOChannel * channel);
extern GIOFlags g_io_channel_get_flags(GIOChannel * channel);
extern const gchar *g_io_channel_get_line_term(GIOChannel * channel,
gint * length);
extern void g_io_channel_init(GIOChannel * channel);
extern GIOChannel *g_io_channel_new_file(const gchar * filename,
const gchar * mode,
GError * *error);
extern GIOError g_io_channel_read(GIOChannel * channel, gchar * buf,
gsize count, gsize * bytes_read);
extern GIOStatus g_io_channel_read_chars(GIOChannel * channel,
gchar * buf,
gsize count, gsize * bytes_read,
GError * *error);
extern GIOStatus g_io_channel_read_line(GIOChannel * channel,
gchar * *str_return,
gsize * length,
gsize * terminator_pos,
GError * *error);
extern GIOStatus g_io_channel_read_line_string(GIOChannel *
channel,
GString * buffer,
gsize * terminator_pos,
GError * *error);
extern GIOStatus g_io_channel_read_to_end(GIOChannel * channel,
gchar * *str_return,
gsize * length, GError * *error);
extern GIOStatus g_io_channel_read_unichar(GIOChannel * channel,
gunichar * thechar,
GError * *error);
extern GIOChannel *g_io_channel_ref(GIOChannel * channel);
extern GIOError g_io_channel_seek(GIOChannel * channel, gint64
offset,
GSeekType type);
extern GIOStatus g_io_channel_seek_position(GIOChannel * channel,
gint64 offset, GSeekType type,
GError * *error);
extern void g_io_channel_set_buffer_size(GIOChannel * channel,
gsize size);
extern void g_io_channel_set_buffered(GIOChannel * channel,
gboolean buffered);
extern void g_io_channel_set_close_on_unref(GIOChannel * channel,
gboolean do_close);
extern GIOStatus g_io_channel_set_encoding(GIOChannel * channel,

```

```

        const gchar * encoding,
        GError * *error);
extern GIOStatus g_io_channel_set_flags(GIOChannel * channel,
        GIOFlags flags, GError * *error);
extern void g_io_channel_set_line_term(GIOChannel * channel,
        const gchar * line_term,
        gint length);
extern GIOStatus g_io_channel_shutdown(GIOChannel * channel,
        gboolean flush, GError * *err);
extern gint g_io_channel_unix_get_fd(GIOChannel * channel);
extern GIOChannel *g_io_channel_unix_new(gint fd);
extern void g_io_channel_unref(GIOChannel * channel);
extern GIOError g_io_channel_write(GIOChannel * channel, const char
*buf,
        gsize count, gsize * bytes_written);
extern GIOStatus g_io_channel_write_chars(GIOChannel * channel,
        const gchar * buf, gssize count,
        gsize * bytes_written,
        GError * *error);
extern GIOStatus g_io_channel_write_unichar(GIOChannel * channel,
        gunichar thechar,
        GError * *error);
extern GSource *g_io_create_watch(GIOChannel * channel,
        GIOCondition condition);
extern GSourceFuncs g_io_watch_funcs;
extern GQuark g_key_file_error_quark(void);
extern void g_key_file_free(GKeyFile * key_file);
extern gboolean g_key_file_get_boolean(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key, GError *
*error);
extern gboolean *g_key_file_get_boolean_list(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key,
        gsize * length,
        GError * *error);
extern gchar *g_key_file_get_comment(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key, GError * *error);
extern gdouble g_key_file_get_double(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key, GError * *error);
extern gdouble *g_key_file_get_double_list(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key,
        gsize * length,
        GError * *error);
extern gchar **g_key_file_get_groups(GKeyFile * key_file, gsize *
length);
extern gint64 g_key_file_get_int64(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key, GError * *error);
extern gint g_key_file_get_integer(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key, GError * *error);
extern gint *g_key_file_get_integer_list(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key, gsize * length,
        GError * *error);
extern gchar **g_key_file_get_keys(GKeyFile * key_file,
        const gchar * group_name,
        gsize * length, GError * *error);
extern gchar *g_key_file_get_locale_string(GKeyFile * key_file,
        const gchar * group_name,
        const gchar * key,
        const gchar * locale,

```

```

                                GError * *error);
extern gchar **g_key_file_get_locale_string_list(GKeyFile *
key_file,
                                const gchar * group_name,
                                const gchar * key,
                                const gchar * locale,
                                gsize * length,
                                GError * *error);
extern gchar *g_key_file_get_start_group(GKeyFile * key_file);
extern gchar *g_key_file_get_string(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, GError * *error);
extern gchar **g_key_file_get_string_list(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key,
                                gsize * length, GError * *error);
extern guint64 g_key_file_get_uint64(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, GError * *error);
extern gchar *g_key_file_get_value(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, GError * *error);
extern gboolean g_key_file_has_group(GKeyFile * key_file,
                                const gchar * group_name);
extern gboolean g_key_file_has_key(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, GError * *error);
extern gboolean g_key_file_load_from_data(GKeyFile * key_file,
                                const gchar * data, gsize length,
                                GKeyFileFlags flags,
                                GError * *error);
extern gboolean g_key_file_load_from_data_dirs(GKeyFile * key_file,
                                const gchar * file,
                                gchar * *full_path,
                                GKeyFileFlags flags,
                                GError * *error);
extern gboolean g_key_file_load_from_dirs(GKeyFile * key_file,
                                const gchar * file,
                                const gchar * *search_dirs,
                                gchar * *full_path,
                                GKeyFileFlags flags,
                                GError * *error);
extern gboolean g_key_file_load_from_file(GKeyFile * key_file,
                                const gchar * file,
                                GKeyFileFlags flags,
                                GError * *error);
extern GKeyFile *g_key_file_new(void);
extern GKeyFile *g_key_file_ref(GKeyFile * key_file);
extern gboolean g_key_file_remove_comment(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key,
                                GError * *error);
extern gboolean g_key_file_remove_group(GKeyFile * key_file,
                                const gchar * group_name,
                                GError * *error);
extern gboolean g_key_file_remove_key(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, GError * *error);
extern void g_key_file_set_boolean(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, gboolean value);
extern void g_key_file_set_boolean_list(GKeyFile * key_file,
                                const gchar * group_name,
                                const gchar * key, gboolean *
list,
                                gsize length);

```

```

extern gboolean g_key_file_set_comment(GKeyFile * key_file,
                                       const gchar * group_name,
                                       const gchar * key,
                                       const gchar * comment,
                                       GError * *error);
extern void g_key_file_set_double(GKeyFile * key_file,
                                  const gchar * group_name,
                                  const gchar * key, gdouble value);
extern void g_key_file_set_double_list(GKeyFile * key_file,
                                       const gchar * group_name,
                                       const gchar * key, gdouble * list,
                                       gsize length);
extern void g_key_file_set_int64(GKeyFile * key_file,
                                 const gchar * group_name,
                                 const gchar * key, gint64 value);
extern void g_key_file_set_integer(GKeyFile * key_file,
                                   const gchar * group_name,
                                   const gchar * key, gint value);
extern void g_key_file_set_integer_list(GKeyFile * key_file,
                                       const gchar * group_name,
                                       const gchar * key, gint * list,
                                       gsize length);
extern void g_key_file_set_list_separator(GKeyFile * key_file,
                                           gchar separator);
extern void g_key_file_set_locale_string(GKeyFile * key_file,
                                         const gchar * group_name,
                                         const gchar * key,
                                         const gchar * locale,
                                         const gchar * string);
extern void g_key_file_set_locale_string_list(GKeyFile * key_file,
                                              const gchar * group_name,
                                              const gchar * key,
                                              const gchar * locale,
                                              const gchar * const *list,
                                              gsize length);
extern void g_key_file_set_string(GKeyFile * key_file,
                                  const gchar * group_name,
                                  const gchar * key, const gchar *
string);
extern void g_key_file_set_string_list(GKeyFile * key_file,
                                       const gchar * group_name,
                                       const gchar * key,
                                       const gchar * const *list,
                                       gsize length);
extern void g_key_file_set_uint64(GKeyFile * key_file,
                                  const gchar * group_name,
                                  const gchar * key, guint64 value);
extern void g_key_file_set_value(GKeyFile * key_file,
                                 const gchar * group_name,
                                 const gchar * key, const gchar * value);
extern gchar *g_key_file_to_data(GKeyFile * key_file, gsize *
length,
                                GError * *error);
extern void g_key_file_unref(GKeyFile * key_file);
extern GList *g_list_alloc(void);
extern GList *g_list_append(GList * list, gpointer data);
extern GList *g_list_concat(GList * list1, GList * list2);
extern GList *g_list_copy(GList * list);
extern GList *g_list_delete_link(GList * list, GList * link_);
extern GList *g_list_find(GList * list, gpointer data);
extern GList *g_list_find_custom(GList * list, gpointer data,
                                GCompareFunc func);
extern GList *g_list_first(GList * list);
extern void g_list_foreach(GList * list, GFunc func, gpointer
user_data);
extern void g_list_free(GList * list);

```

```

extern void g_list_free_1(GList * list);
extern void g_list_free_full(GList * list, GDestroyNotify
free_func);
extern gint g_list_index(GList * list, gconstpointer data);
extern GList *g_list_insert(GList * list, gpointer data, gint
position);
extern GList *g_list_insert_before(GList * list, GList * sibling,
gpointer data);
extern GList *g_list_insert_sorted(GList * list, gpointer data,
GCompareFunc func);
extern GList *g_list_insert_sorted_with_data(GList * list, gpointer
data,
GCompareDataFunc func,
gpointer user_data);

extern GList *g_list_last(GList * list);
extern guint g_list_length(GList * list);
extern GList *g_list_nth(GList * list, guint n);
extern gpointer g_list_nth_data(GList * list, guint n);
extern GList *g_list_nth_prev(GList * list, guint n);
extern void g_list_pop_allocator(void);
extern gint g_list_position(GList * list, GList * llink);
extern GList *g_list_prepend(GList * list, gpointer data);
extern void g_list_push_allocator(GAllocator * allocator);
extern GList *g_list_remove(GList * list, gconstpointer data);
extern GList *g_list_remove_all(GList * list, gconstpointer data);
extern GList *g_list_remove_link(GList * list, GList * llink);
extern GList *g_list_reverse(GList * list);
extern GList *g_list_sort(GList * list, GCompareFunc compare_func);
extern GList *g_list_sort_with_data(GList * list,
GCompareDataFunc compare_func,
gpointer user_data);

extern gchar **g_listenv(void);
extern gchar *g_locale_from_utf8(const gchar * utf8string, gssize
len,
gsize * bytes_read, gsize *
bytes_written,
GError * *error);
extern gchar *g_locale_to_utf8(const gchar * opsysstring, gssize
len,
gsize * bytes_read, gsize * bytes_written,
GError * *error);
extern void g_log(const gchar * log_domain, GLogLevelFlags
log_level,
const gchar * format, ...);
extern void g_log_default_handler(const gchar * log_domain,
GLogLevelFlags log_level,
const gchar * message,
gpointer unused_data);
extern void g_log_remove_handler(const gchar * log_domain,
guint handler_id);
extern GLogLevelFlags g_log_set_always_fatal(GLogLevelFlags
fatal_mask);
extern GLogFunc g_log_set_default_handler(GLogFunc log_func,
gpointer user_data);
extern GLogLevelFlags g_log_set_fatal_mask(const gchar * log_domain,
GLogLevelFlags fatal_mask);
extern guint g_log_set_handler(const gchar * log_domain,
GLogLevelFlags log_levels,
GLogFunc log_func, gpointer user_data);
extern void g_logv(const gchar * log_domain, GLogLevelFlags
log_level,
const gchar * format, va_list args);
extern gboolean g_main_context_acquire(GMainContext * context);
extern void g_main_context_add_poll(GMainContext * context, GPollFD
* fd,
gint priority);

```

```

extern gint g_main_context_check(GMainContext * context, gint
max_priority,
                                GPollFD * fds, gint n_fds);
extern GMainContext *g_main_context_default(void);
extern void g_main_context_dispatch(GMainContext * context);
extern
                                GSource
*g_main_context_find_source_by_funcs_user_data(GMainContext
                                * context,
                                GSourceFuncs
                                * funcs,
                                gpointer
                                user_data);
extern GSource *g_main_context_find_source_by_id(GMainContext *
context,
                                guint source_id);
extern
                                GSource
*g_main_context_find_source_by_user_data(GMainContext *
                                context,
                                gpointer
                                user_data);
extern GPollFunc g_main_context_get_poll_func(GMainContext *
context);
extern GMainContext *g_main_context_get_thread_default(void);
extern void g_main_context_invoke(GMainContext * context,
                                GSourceFunc function, void *data);
extern void g_main_context_invoke_full(GMainContext * context,
                                gint priority, GSourceFunc
function,
                                void *data, GDestroyNotify notify);
extern gboolean g_main_context_is_owner(GMainContext * context);
extern gboolean g_main_context_iteration(GMainContext * context,
                                gboolean may_block);
extern GMainContext *g_main_context_new(void);
extern gboolean g_main_context_pending(GMainContext * context);
extern void g_main_context_pop_thread_default(GMainContext *
context);
extern gboolean g_main_context_prepare(GMainContext * context,
                                gint * priority);
extern void g_main_context_push_thread_default(GMainContext *
context);
extern gint g_main_context_query(GMainContext * context, gint
max_priority,
                                gint * timeout_, GPollFD * fds,
                                gint n_fds);
extern GMainContext *g_main_context_ref(GMainContext * context);
extern GMainContext *g_main_context_ref_thread_default(void);
extern void g_main_context_release(GMainContext * context);
extern void g_main_context_remove_poll(GMainContext * context,
                                GPollFD * fd);
extern void g_main_context_set_poll_func(GMainContext * context,
                                GPollFunc func);
extern void g_main_context_unref(GMainContext * context);
extern gboolean g_main_context_wait(GMainContext * context, GCond
* cond,
                                GMutex * mutex);
extern void g_main_context_wakeup(GMainContext * context);
extern GSource *g_main_current_source(void);
extern gint g_main_depth(void);
extern GMainContext *g_main_loop_get_context(GMainLoop * loop);
extern gboolean g_main_loop_is_running(GMainLoop * loop);
extern GMainLoop *g_main_loop_new(GMainContext * context,
                                gboolean is_running);
extern void g_main_loop_quit(GMainLoop * loop);
extern GMainLoop *g_main_loop_ref(GMainLoop * loop);
extern void g_main_loop_run(GMainLoop * loop);
extern void g_main_loop_unref(GMainLoop * loop);

```

```

extern gpointer g_malloc(gulong n_bytes);
extern gpointer g_malloc0(gulong n_bytes);
extern void *g_malloc0_n(gsize n_blocks, gsize n_block_bytes);
extern void *g_malloc_n(gsize n_blocks, gsize n_block_bytes);
extern void g_mapped_file_free(GMappedFile * file);
extern gchar *g_mapped_file_get_contents(GMappedFile * file);
extern gsize g_mapped_file_get_length(GMappedFile * file);
extern GMappedFile *g_mapped_file_new(const gchar * filename,
                                     gboolean writable, GError * *error);
extern GMappedFile *g_mapped_file_new_from_fd(gint fd, gboolean
writable,
                                     GError * *error);
extern GMappedFile *g_mapped_file_ref(GMappedFile * file);
extern void g_mapped_file_unref(GMappedFile * file);
extern gboolean g_markup_collect_attributes(const gchar *
element_name,
                                     const gchar *
*attribute_names,
                                     const gchar *
*attribute_values,
                                     GError * *error,
                                     GMarkupCollectType first_type,
                                     const gchar * first_attr, ...);
extern GQuark g_markup_error_quark(void);
extern gchar *g_markup_escape_text(const gchar * text, gssize
length);
extern gboolean
g_markup_parse_context_end_parse(GMarkupParseContext *
context, GError * *error);
extern void g_markup_parse_context_free(GMarkupParseContext *
context);
extern const GList
*g_markup_parse_context_get_element_stack(GMarkupParseContext
*
context);
extern void
g_markup_parse_context_get_position(GMarkupParseContext *
context,
gint * line_number,
gint * char_number);
extern void
*g_markup_parse_context_get_user_data(GMarkupParseContext *
context);
extern GMarkupParseContext *g_markup_parse_context_new(const
GMarkupParser
* parser,
GMarkupParseFlags
flags,
gpointer user_data,
GDestroyNotify
user_data_dnotify);
extern gboolean g_markup_parse_context_parse(GMarkupParseContext *
context,
const gchar * text,
gssize text_len,
GError * *error);
extern void *g_markup_parse_context_pop(GMarkupParseContext *
context);
extern void g_markup_parse_context_push(GMarkupParseContext *
context,
const GMarkupParser * parser,
void *user_data);
extern char *g_markup_printf_escaped(const char *format, ...);

```



```

extern char *g_markup_vprintf_escaped(const char *format, va_list
args);
extern gchar *g_match_info_expand_references(const GMatchInfo *
match_info,
                                     const gchar *
                                     string_to_expand,
                                     GError * *error);
extern gchar *g_match_info_fetch(const GMatchInfo * match_info,
                                gint match_num);
extern gchar **g_match_info_fetch_all(const GMatchInfo *
match_info);
extern gchar *g_match_info_fetch_named(const GMatchInfo *
match_info,
                                     const gchar * name);
extern gboolean g_match_info_fetch_named_pos(const GMatchInfo *
match_info,
                                     const gchar * name,
                                     gint * start_pos,
                                     gint * end_pos);
extern gboolean g_match_info_fetch_pos(const GMatchInfo *
match_info,
                                     gint match_num, gint * start_pos,
                                     gint * end_pos);
extern void g_match_info_free(GMatchInfo * match_info);
extern gint g_match_info_get_match_count(const GMatchInfo *
match_info);
extern GRegex *g_match_info_get_regex(const GMatchInfo *
match_info);
extern const gchar *g_match_info_get_string(const GMatchInfo *
match_info);
extern gboolean g_match_info_is_partial_match(const GMatchInfo *
match_info);
extern gboolean g_match_info_matches(const GMatchInfo * match_info);
extern gboolean g_match_info_next(GMatchInfo * match_info,
GError * *error);
extern GMatchInfo *g_match_info_ref(GMatchInfo * match_info);
extern void g_match_info_unref(GMatchInfo * match_info);
extern gpointer g_mem_chunk_alloc(GMemChunk * mem_chunk);
extern gpointer g_mem_chunk_alloc0(GMemChunk * mem_chunk);
extern void g_mem_chunk_clean(GMemChunk * mem_chunk);
extern void g_mem_chunk_destroy(GMemChunk * mem_chunk);
extern void g_mem_chunk_free(GMemChunk * mem_chunk, gpointer mem);
extern void g_mem_chunk_info(void);
extern GMemChunk *g_mem_chunk_new(const gchar * name, gint
atom_size,
                                gulong area_size, gint type);
extern void g_mem_chunk_print(GMemChunk * mem_chunk);
extern void g_mem_chunk_reset(GMemChunk * mem_chunk);
extern gboolean g_mem_gc_friendly;
extern gboolean g_mem_is_system_malloc(void);
extern void g_mem_profile(void);
extern void g_mem_set_vtable(GMemVTable * vtable);
extern gpointer g_memdup(gconstpointer mem, guint byte_size);
extern gint g_mkdir_with_parents(const gchar * pathname, int mode);
extern gchar *g_mkdtemp(gchar * tmpl);
extern gchar *g_mkdtemp_full(gchar * tmpl, gint mode);
extern gint g_mkstemp(gchar * tmpl);
extern gint g_mkstemp_full(gchar * tmpl, gint flags, gint mode);
extern void g_mutex_clear(GMutex * mutex);
extern void g_mutex_free(GMutex * mutex);
extern void g_mutex_init(GMutex * mutex);
extern void g_mutex_lock(GMutex * mutex);
extern GMutex *g_mutex_new(void);
extern gboolean g_mutex_trylock(GMutex * mutex);
extern void g_mutex_unlock(GMutex * mutex);
extern gint g_node_child_index(GNode * node, gpointer data);

```

```

extern gint g_node_child_position(GNode * node, GNode * child);
extern void g_node_children_foreach(GNode * node, GTraverseFlags
flags,
                                GNodeForeachFunc func, gpointer
data);
extern GNode *g_node_copy(GNode * node);
extern GNode *g_node_copy_deep(GNode * node, GCopyFunc copy_func,
                                gpointer data);
extern guint g_node_depth(GNode * node);
extern void g_node_destroy(GNode * root);
extern GNode *g_node_find(GNode * root, GTraverseType order,
                                GTraverseFlags flags, gpointer data);
extern GNode *g_node_find_child(GNode * node, GTraverseFlags flags,
                                gpointer data);
extern GNode *g_node_first_sibling(GNode * node);
extern GNode *g_node_get_root(GNode * node);
extern GNode *g_node_insert(GNode * parent, gint position, GNode *
node);
extern GNode *g_node_insert_after(GNode * parent, GNode * sibling,
                                GNode * node);
extern GNode *g_node_insert_before(GNode * parent, GNode * sibling,
                                GNode * node);
extern gboolean g_node_is_ancestor(GNode * node, GNode *
descendant);
extern GNode *g_node_last_child(GNode * node);
extern GNode *g_node_last_sibling(GNode * node);
extern guint g_node_max_height(GNode * root);
extern guint g_node_n_children(GNode * node);
extern guint g_node_n_nodes(GNode * root, GTraverseFlags flags);
extern GNode *g_node_new(gpointer data);
extern GNode *g_node_nth_child(GNode * node, guint n);
extern void g_node_pop_allocator(void);
extern GNode *g_node_prepend(GNode * parent, GNode * node);
extern void g_node_push_allocator(GAllocator * dummy);
extern void g_node_reverse_children(GNode * node);
extern void g_node_traverse(GNode * root, GTraverseType order,
                                GTraverseFlags flags, gint max_depth,
                                GNodeTraverseFunc func, gpointer data);

extern void g_node_unlink(GNode * node);
extern void g_nullify_pointer(gpointer * nullify_location);
extern void g_on_error_query(const gchar * prg_name);
extern void g_on_error_stack_trace(const gchar * prg_name);
extern gpointer g_once_impl(GOnce * once, GThreadFunc func,
gpointer arg);
extern gboolean g_once_init_enter(volatile void *location);
extern gboolean g_once_init_enter_impl(volatile unsigned int
*location);
extern void g_once_init_leave(volatile void *location, gsize
result);
extern void g_option_context_add_group(GOptionContext * context,
                                GOptionGroup * group);
extern void g_option_context_add_main_entries(GOptionContext *
context,
                                const GOptionEntry * entries,
                                const gchar *
translation_domain);
extern void g_option_context_free(GOptionContext * context);
extern const gchar
*g_option_context_get_description(GOptionContext *
                                context);
extern gchar *g_option_context_get_help(GOptionContext * context,
                                gboolean main_help,
                                GOptionGroup * group);
extern gboolean g_option_context_get_help_enabled(GOptionContext *
context);

```

```

extern                                     gboolean
g_option_context_get_ignore_unknown_options(GOptionContext
                                             * context);
extern                                     GOptionGroup
*g_option_context_get_main_group(GOptionContext *
                                  context);
extern const gchar *g_option_context_get_summary(GOptionContext *
context);
extern GOptionContext *g_option_context_new(const gchar *
                                             parameter_string);
extern gboolean g_option_context_parse(GOptionContext * context,
                                       gint * argc, gchar * **argv,
                                       GError * *error);
extern void g_option_context_set_description(GOptionContext *
context,
                                             const gchar * description);
extern void g_option_context_set_help_enabled(GOptionContext *
context,
                                             gboolean help_enabled);
extern                                     void
g_option_context_set_ignore_unknown_options(GOptionContext *
                                             context,
                                             gboolean
                                             ignore_unknown);
extern void g_option_context_set_main_group(GOptionContext *
context,
                                             GOptionGroup * group);
extern void g_option_context_set_summary(GOptionContext * context,
                                         const gchar * summary);
extern void g_option_context_set_translate_func(GOptionContext *
context,
                                             GTranslateFunc func,
                                             gpointer data,
                                             GDestroyNotify
                                             destroy_notify);
extern void g_option_context_set_translation_domain(GOptionContext
*
                                             context,
                                             const gchar * domain);
extern GQuark g_option_error_quark(void);
extern void g_option_group_add_entries(GOptionGroup * group,
                                       const GOptionEntry * entries);
extern void g_option_group_free(GOptionGroup * group);
extern GOptionGroup *g_option_group_new(const gchar * name,
                                         const gchar * description,
                                         const gchar * help_description,
                                         gpointer user_data,
                                         GDestroyNotify destroy);
extern void g_option_group_set_error_hook(GOptionGroup * group,
                                           GOptionErrorFunc error_func);
extern void g_option_group_set_parse_hooks(GOptionGroup * group,
                                           GOptionParseFunc
pre_parse_func,
                                           GOptionParseFunc
post_parse_func);
extern void g_option_group_set_translate_func(GOptionGroup * group,
                                              GTranslateFunc func,
                                              gpointer data,
                                              GDestroyNotify
                                              destroy_notify);
extern void g_option_group_set_translation_domain(GOptionGroup *
group,
                                                  const gchar * domain);
extern guint g_parse_debug_string(const gchar * string,
                                  const GDebugKey * keys, guint nkeys);
extern gchar *g_path_get_basename(const gchar * file_name);

```

```

extern gchar *g_path_get_dirname(const gchar * file_name);
extern gboolean g_path_is_absolute(const gchar * file_name);
extern const gchar *g_path_skip_root(const gchar * file_name);
extern gboolean g_pattern_match(GPatternSpec * pspec, guint
string_length,
                                const gchar * string,
                                const gchar * string_reversed);
extern gboolean g_pattern_match_simple(const gchar * pattern,
                                const gchar * string);
extern gboolean g_pattern_match_string(GPatternSpec * pspec,
                                const gchar * string);
extern gboolean g_pattern_spec_equal(GPatternSpec * pspec1,
                                GPatternSpec * pspec2);
extern void g_pattern_spec_free(GPatternSpec * pspec);
extern GPatternSpec *g_pattern_spec_new(const gchar * pattern);
extern void g_pointer_bit_lock(volatile void *address, gint
lock_bit);
extern gboolean g_pointer_bit_trylock(volatile void *address,
                                gint lock_bit);
extern void g_pointer_bit_unlock(volatile void *address, gint
lock_bit);
extern gint g_poll(GPollFD * fds, guint nfds, gint timeout);
extern void g_prefix_error(GError * *err, const gchar * format, ...);
extern void g_print(const gchar * format, ...);
extern void g_printerr(const gchar * format, ...);
extern gsize g_printf_string_upper_bound(const gchar * format,
                                va_list args);
extern void *g_private_get(GPrivate * key);
extern GPrivate *g_private_new(GDestroyNotify notify);
extern void g_private_replace(GPrivate * key, void *value);
extern void g_private_set(GPrivate * key, void *value);
extern void g_propagate_error(GError * *dest, GError * src);
extern void g_propagate_prefixed_error(GError * *dest, GError * src,
                                const gchar * format, ...);
extern void g_ptr_array_add(GPtrArray * array, gpointer data);
extern void g_ptr_array_foreach(GPtrArray * array, GFunc func,
                                gpointer user_data);
extern gpointer *g_ptr_array_free(GPtrArray * array, gboolean
free_seg);
extern GPtrArray *g_ptr_array_new(void);
extern GPtrArray *g_ptr_array_new_full(guint reserved_size,
                                GDestroyNotify element_free_func);
extern GPtrArray *g_ptr_array_new_with_free_func(GDestroyNotify
                                element_free_func);
extern GPtrArray *g_ptr_array_ref(GPtrArray * array);
extern gboolean g_ptr_array_remove(GPtrArray * array, gpointer
data);
extern gboolean g_ptr_array_remove_fast(GPtrArray * array, gpointer
data);
extern gpointer g_ptr_array_remove_index(GPtrArray * array, guint
index_);
extern gpointer g_ptr_array_remove_index_fast(GPtrArray * array,
                                guint index_);
extern void g_ptr_array_remove_range(GPtrArray * array, guint
index_,
                                guint length);
extern void g_ptr_array_set_free_func(GPtrArray * array,
                                GDestroyNotify element_free_func);
extern void g_ptr_array_set_size(GPtrArray * array, gint length);
extern GPtrArray *g_ptr_array_sized_new(guint reserved_size);
extern void g_ptr_array_sort(GPtrArray * array, GCompareFunc
compare_func);
extern void g_ptr_array_sort_with_data(GPtrArray * array,
                                GCompareDataFunc compare_func,
                                gpointer user_data);
extern void g_ptr_array_unref(GPtrArray * array);

```

```

extern void g_qsort_with_data(gconstpointer pbase, gint total_elems,
                             gsize size, GCompareDataFunc compare_func,
                             gpointer user_data);
extern GQuark g_quark_from_static_string(const gchar * string);
extern GQuark g_quark_from_string(const gchar * string);
extern const gchar *g_quark_to_string(GQuark quark);
extern GQuark g_quark_try_string(const gchar * string);
extern void g_queue_clear(GQueue * queue);
extern GQueue *g_queue_copy(GQueue * queue);
extern void g_queue_delete_link(GQueue * queue, GList * link_);
extern GList *g_queue_find(GQueue * queue, gconstpointer data);
extern GList *g_queue_find_custom(GQueue * queue, gconstpointer
data,
                                GCompareFunc func);
extern void g_queue_foreach(GQueue * queue, GFunc func,
                             gpointer user_data);
extern void g_queue_free(GQueue * queue);
extern void g_queue_free_full(GQueue * queue, GDestroyNotify
free_func);
extern guint g_queue_get_length(GQueue * queue);
extern gint g_queue_index(GQueue * queue, gconstpointer data);
extern void g_queue_init(GQueue * queue);
extern void g_queue_insert_after(GQueue * queue, GList * sibling,
                                gpointer data);
extern void g_queue_insert_before(GQueue * queue, GList * sibling,
                                gpointer data);
extern void g_queue_insert_sorted(GQueue * queue, gpointer data,
                                GCompareDataFunc func,
                                gpointer user_data);
extern gboolean g_queue_is_empty(GQueue * queue);
extern gint g_queue_link_index(GQueue * queue, GList * link_);
extern GQueue *g_queue_new(void);
extern gpointer g_queue_peek_head(GQueue * queue);
extern GList *g_queue_peek_head_link(GQueue * queue);
extern gpointer g_queue_peek_nth(GQueue * queue, guint n);
extern GList *g_queue_peek_nth_link(GQueue * queue, guint n);
extern gpointer g_queue_peek_tail(GQueue * queue);
extern GList *g_queue_peek_tail_link(GQueue * queue);
extern gpointer g_queue_pop_head(GQueue * queue);
extern GList *g_queue_pop_head_link(GQueue * queue);
extern gpointer g_queue_pop_nth(GQueue * queue, guint n);
extern GList *g_queue_pop_nth_link(GQueue * queue, guint n);
extern gpointer g_queue_pop_tail(GQueue * queue);
extern GList *g_queue_pop_tail_link(GQueue * queue);
extern void g_queue_push_head(GQueue * queue, gpointer data);
extern void g_queue_push_head_link(GQueue * queue, GList * link_);
extern void g_queue_push_nth(GQueue * queue, gpointer data, gint
n);
extern void g_queue_push_nth_link(GQueue * queue, gint n, GList *
link_);
extern void g_queue_push_tail(GQueue * queue, gpointer data);
extern void g_queue_push_tail_link(GQueue * queue, GList * link_);
extern gboolean g_queue_remove(GQueue * queue, gconstpointer data);
extern guint g_queue_remove_all(GQueue * queue, gconstpointer data);
extern void g_queue_reverse(GQueue * queue);
extern void g_queue_sort(GQueue * queue, GCompareDataFunc
compare_func,
                        gpointer user_data);
extern void g_queue_unlink(GQueue * queue, GList * link_);
extern GRandom *g_rand_copy(GRandom * rand_);
extern gdouble g_rand_double(GRandom * rand_);
extern gdouble g_rand_double_range(GRandom * rand_, gdouble begin,
                                gdouble end);
extern void g_rand_free(GRandom * rand_);
extern guint32 g_rand_int(GRandom * rand_);

```

```

extern gint32 g_rand_int_range(GRand * rand_, gint32 begin, gint32
end);
extern GRand *g_rand_new(void);
extern GRand *g_rand_new_with_seed(guint32 seed);
extern GRand *g_rand_new_with_seed_array(const guint32 * seed,
guint seed_length);
extern void g_rand_set_seed(GRand * rand_, guint32 seed);
extern void g_rand_set_seed_array(GRand * rand_, const guint32 *
seed,
guint seed_length);
extern gdouble g_random_double(void);
extern gdouble g_random_double_range(gdouble begin, gdouble end);
extern guint32 g_random_int(void);
extern gint32 g_random_int_range(gint32 begin, gint32 end);
extern void g_random_set_seed(guint32 seed);
extern gpointer g_realloc(gpointer mem, gulong n_bytes);
extern void *g_realloc_n(void *mem, gsize n_blocks, gsize
n_block_bytes);
extern void g_rec_mutex_clear(GRecMutex * rec_mutex);
extern void g_rec_mutex_init(GRecMutex * rec_mutex);
extern void g_rec_mutex_lock(GRecMutex * rec_mutex);
extern gboolean g_rec_mutex_trylock(GRecMutex * rec_mutex);
extern void g_rec_mutex_unlock(GRecMutex * rec_mutex);
extern gboolean g_regex_check_replacement(const gchar * replacement,
gboolean * has_references,
GError * *error);
extern GQuark g_regex_error_quark(void);
extern gchar *g_regex_escape_nul(const gchar * string, gint length);
extern gchar *g_regex_escape_string(const gchar * string, gint
length);
extern gint g_regex_get_capture_count(const GRegex * regex);
extern GRegexCompileFlags g_regex_get_compile_flags(const GRegex *
regex);
extern GRegexMatchFlags g_regex_get_match_flags(const GRegex *
regex);
extern gint g_regex_get_max_backref(const GRegex * regex);
extern const gchar *g_regex_get_pattern(const GRegex * regex);
extern gint g_regex_get_string_number(const GRegex * regex,
const gchar * name);
extern gboolean g_regex_match(const GRegex * regex, const gchar *
string,
GRegexMatchFlags match_options,
GMatchInfo * *match_info);
extern gboolean g_regex_match_all(const GRegex * regex,
const gchar * string,
GRegexMatchFlags match_options,
GMatchInfo * *match_info);
extern gboolean g_regex_match_all_full(const GRegex * regex,
const gchar * string,
gssize string_len,
gint start_position,
GRegexMatchFlags match_options,
GMatchInfo * *match_info,
GError * *error);
extern gboolean g_regex_match_full(const GRegex * regex,
const gchar * string, gssize
string_len,
gint start_position,
GRegexMatchFlags match_options,
GMatchInfo * *match_info,
GError * *error);
extern gboolean g_regex_match_simple(const gchar * pattern,
const gchar * string,
GRegexCompileFlags compile_options,
GRegexMatchFlags match_options);
extern GRegex *g_regex_new(const gchar * pattern,

```

```

        GRegexCompileFlags compile_options,
        GRegexMatchFlags match_options,
        GError * *error);
extern GRegex *g_regex_ref(GRegex * regex);
extern gchar *g_regex_replace(const GRegex * regex, const gchar *
string,
        gssize string_len, gint start_position,
        const gchar * replacement,
        GRegexMatchFlags match_options,
        GError * *error);
extern gchar *g_regex_replace_eval(const GRegex * regex,
        const gchar * string, gssize
string_len,
        gint start_position,
        GRegexMatchFlags match_options,
        GRegexEvalCallback eval,
        void *user_data, GError * *error);
extern gchar *g_regex_replace_literal(const GRegex * regex,
        const gchar * string,
        gssize string_len,
        gint start_position,
        const gchar * replacement,
        GRegexMatchFlags match_options,
        GError * *error);
extern gchar **g_regex_split(const GRegex * regex, const gchar *
string,
        GRegexMatchFlags match_options);
extern gchar **g_regex_split_full(const GRegex * regex,
        const gchar * string, gssize
string_len,
        gint start_position,
        GRegexMatchFlags match_options,
        gint max_tokens, GError * *error);
extern gchar **g_regex_split_simple(const gchar * pattern,
        const gchar * string,
        GRegexCompileFlags compile_options,
        GRegexMatchFlags match_options);
extern void g_regex_unref(GRegex * regex);
extern gint g_relation_count(GRelation * relation, gconstpointer
key,
        gint field);
extern gint g_relation_delete(GRelation * relation, gconstpointer
key,
        gint field);
extern void g_relation_destroy(GRelation * relation);
extern gboolean g_relation_exists(GRelation * relation, ...);
extern void g_relation_index(GRelation * relation, gint field,
        GHashFunc hash_func,
        GEqualFunc key_equal_func);
extern void g_relation_insert(GRelation * relation, ...);
extern GRelation *g_relation_new(gint fields);
extern void g_relation_print(GRelation * relation);
extern GTuples *g_relation_select(GRelation * relation,
        gconstpointer key,
        gint field);
extern void g_reload_user_special_dirs_cache(void);
extern void g_return_if_fail_warning(const char *log_domain,
        const char *pretty_function,
        const char *expression);
extern void g_rw_lock_clear(GRWLock * rw_lock);
extern void g_rw_lock_init(GRWLock * rw_lock);
extern void g_rw_lock_reader_lock(GRWLock * rw_lock);
extern gboolean g_rw_lock_reader_trylock(GRWLock * rw_lock);
extern void g_rw_lock_reader_unlock(GRWLock * rw_lock);
extern void g_rw_lock_writer_lock(GRWLock * rw_lock);
extern gboolean g_rw_lock_writer_trylock(GRWLock * rw_lock);

```



```

extern void g_rw_lock_writer_unlock(GRWLock * rw_lock);
extern guint g_scanner_cur_line(GScanner * scanner);
extern guint g_scanner_cur_position(GScanner * scanner);
extern GTokenType g_scanner_cur_token(GScanner * scanner);
extern GTokenValue g_scanner_cur_value(GScanner * scanner);
extern void g_scanner_destroy(GScanner * scanner);
extern gboolean g_scanner_eof(GScanner * scanner);
extern void g_scanner_error(GScanner * scanner, const gchar *
format, ...);
extern GTokenType g_scanner_get_next_token(GScanner * scanner);
extern void g_scanner_input_file(GScanner * scanner, gint input_fd);
extern void g_scanner_input_text(GScanner * scanner, const gchar *
text,
                                guint text_len);
extern gpointer g_scanner_lookup_symbol(GScanner * scanner,
                                        const gchar * symbol);
extern GScanner *g_scanner_new(const GScannerConfig * config_template);
extern GTokenType g_scanner_peek_next_token(GScanner * scanner);
extern void g_scanner_scope_add_symbol(GScanner * scanner, guint
scope_id,
                                        const gchar * symbol,
                                        gpointer value);
extern void g_scanner_scope_foreach_symbol(GScanner * scanner,
                                           guint scope_id, GFunc func,
                                           gpointer user_data);
extern gpointer g_scanner_scope_lookup_symbol(GScanner * scanner,
                                              guint scope_id,
                                              const gchar * symbol);
extern void g_scanner_scope_remove_symbol(GScanner * scanner,
                                           guint scope_id,
                                           const gchar * symbol);
extern guint g_scanner_set_scope(GScanner * scanner, guint
scope_id);
extern void g_scanner_sync_file_offset(GScanner * scanner);
extern void g_scanner_unexp_token(GScanner * scanner,
                                  GTokenType expected_token,
                                  const gchar * identifier_spec,
                                  const gchar * symbol_spec,
                                  const gchar * symbol_name,
                                  const gchar * message, gint is_error);
extern void g_scanner_warn(GScanner * scanner, const gchar *
format, ...);
extern GSequenceIter *g_sequence_append(GSequence * seq, void
*data);
extern void g_sequence_foreach(GSequence * seq, GFunc func,
                               void *user_data);
extern void g_sequence_foreach_range(GSequenceIter * begin,
                                     GSequenceIter * end, GFunc func,
                                     void *user_data);
extern void g_sequence_free(GSequence * seq);
extern void *g_sequence_get(GSequenceIter * iter);
extern GSequenceIter *g_sequence_get_begin_iter(GSequence * seq);
extern GSequenceIter *g_sequence_get_end_iter(GSequence * seq);
extern GSequenceIter *g_sequence_get_iter_at_pos(GSequence * seq,
                                                  gint pos);
extern gint g_sequence_get_length(GSequence * seq);
extern GSequenceIter *g_sequence_insert_before(GSequenceIter *
iter,
                                              void *data);
extern GSequenceIter *g_sequence_insert_sorted(GSequence * seq,
                                              void *data,
                                              GCompareDataFunc cmp_func,
                                              void *cmp_data);
extern GSequenceIter *g_sequence_insert_sorted_iter(GSequence *
seq,
                                                    void *data,

```



```

GSequenceIterCompareFunc
    iter_cmp,
    void *cmp_data);
extern gint g_sequence_iter_compare(GSequenceIter * a,
GSequenceIter * b);
extern gint g_sequence_iter_get_position(GSequenceIter * iter);
extern GSequence *g_sequence_iter_get_sequence(GSequenceIter *
iter);
extern gboolean g_sequence_iter_is_begin(GSequenceIter * iter);
extern gboolean g_sequence_iter_is_end(GSequenceIter * iter);
extern GSequenceIter *g_sequence_iter_move(GSequenceIter * iter,
    gint delta);
extern GSequenceIter *g_sequence_iter_next(GSequenceIter * iter);
extern GSequenceIter *g_sequence_iter_prev(GSequenceIter * iter);
extern GSequenceIter *g_sequence_lookup(GSequence * seq, void *data,
    GCompareDataFunc cmp_func,
    void *cmp_data);
extern GSequenceIter *g_sequence_lookup_iter(GSequence * seq, void
*data,
    GSequenceIterCompareFunc
    iter_cmp, void *cmp_data);
extern void g_sequence_move(GSequenceIter * src, GSequenceIter *
dest);
extern void g_sequence_move_range(GSequenceIter * dest,
    GSequenceIter * begin,
    GSequenceIter * end);
extern GSequence *g_sequence_new(GDestroyNotify data_destroy);
extern GSequenceIter *g_sequence_prepend(GSequence * seq, void
*data);
extern GSequenceIter *g_sequence_range_get_midpoint(GSequenceIter
* begin,
    GSequenceIter * end);
extern void g_sequence_remove(GSequenceIter * iter);
extern void g_sequence_remove_range(GSequenceIter * begin,
    GSequenceIter * end);
extern GSequenceIter *g_sequence_search(GSequence * seq, void *data,
    GCompareDataFunc cmp_func,
    void *cmp_data);
extern GSequenceIter *g_sequence_search_iter(GSequence * seq, void
*data,
    GSequenceIterCompareFunc
    iter_cmp, void *cmp_data);
extern void g_sequence_set(GSequenceIter * iter, void *data);
extern void g_sequence_sort(GSequence * seq, GCompareDataFunc
cmp_func,
    void *cmp_data);
extern void g_sequence_sort_changed(GSequenceIter * iter,
    GCompareDataFunc cmp_func,
    void *cmp_data);
extern void g_sequence_sort_changed_iter(GSequenceIter * iter,
    GSequenceIterCompareFunc
    iter_cmp,
    void *cmp_data);
extern void g_sequence_sort_iter(GSequence * seq,
    GSequenceIterCompareFunc cmp_func,
    void *cmp_data);
extern void g_sequence_swap(GSequenceIter * a, GSequenceIter * b);
extern void g_set_application_name(const gchar * application_name);
extern void g_set_error(GError * *err, GQuark domain, gint code,
    const gchar * format, ...);
extern void g_set_error_literal(GError * *err, GQuark domain, gint
code,
    const gchar * message);
extern void g_set_prgrname(const gchar * prgrname);
extern GPrintFunc g_set_print_handler(GPrintFunc func);

```

```

extern GPrintFunc g_set_printerr_handler(GPrintFunc func);
extern gboolean g_setenv(const gchar * variable, const gchar *
value,
                        gboolean overwrite);
extern GQuark g_shell_error_quark(void);
extern gboolean g_shell_parse_argv(const gchar * command_line,
                                gint * argcp, gchar * **argvp,
                                GError * *error);
extern gchar *g_shell_quote(const gchar * unquoted_string);
extern gchar *g_shell_unquote(const gchar * quoted_string,
                              GError * *error);
extern gpointer g_slice_alloc(gsize block_size);
extern gpointer g_slice_alloc0(gsize block_size);
extern void *g_slice_copy(gsize, gconstpointer);
extern void g_slice_free1(gsize block_size, gpointer mem_block);
extern void g_slice_free_chain_with_offset(gsize block_size,
                                           gpointer mem_chain,
                                           gsize next_offset);

extern GSList *g_slist_alloc(void);
extern GSList *g_slist_append(GSList * list, gpointer data);
extern GSList *g_slist_concat(GSList * list1, GSList * list2);
extern GSList *g_slist_copy(GSList * list);
extern GSList *g_slist_delete_link(GSList * list, GSList * link_);
extern GSList *g_slist_find(GSList * list, gconstpointer data);
extern GSList *g_slist_find_custom(GSList * list, gconstpointer
data,
                                GCompareFunc func);
extern void g_slist_foreach(GSList * list, GFunc func, gpointer
user_data);
extern void g_slist_free(GSList * list);
extern void g_slist_free_1(GSList * list);
extern void g_slist_free_full(GSList * list, GDestroyNotify
free_func);
extern gint g_slist_index(GSList * list, gconstpointer data);
extern GSList *g_slist_insert(GSList * list, gpointer data, gint
position);
extern GSList *g_slist_insert_before(GSList * slist, GSList *
sibling,
                                gpointer data);
extern GSList *g_slist_insert_sorted(GSList * list, gpointer data,
                                GCompareFunc func);
extern GSList *g_slist_insert_sorted_with_data(GSList * list,
                                gpointer data,
                                GCompareDataFunc func,
                                gpointer user_data);
extern GSList *g_slist_last(GSList * list);
extern guint g_slist_length(GSList * list);
extern GSList *g_slist_nth(GSList * list, guint n);
extern gpointer g_slist_nth_data(GSList * list, guint n);
extern void g_slist_pop_allocator(void);
extern gint g_slist_position(GSList * list, GSList * llink);
extern GSList *g_slist_prepend(GSList * list, gpointer data);
extern void g_slist_push_allocator(GAllocator * dummy);
extern GSList *g_slist_remove(GSList * list, gconstpointer data);
extern GSList *g_slist_remove_all(GSList * list, gconstpointer
data);
extern GSList *g_slist_remove_link(GSList * list, GSList * link_);
extern GSList *g_slist_reverse(GSList * list);
extern GSList *g_slist_sort(GSList * list, GCompareFunc
compare_func);
extern GSList *g_slist_sort_with_data(GSList * list,
                                GCompareDataFunc compare_func,
                                gpointer user_data);
extern void g_source_add_child_source(GSource * source,
                                GSource * child_source);
extern void g_source_add_poll(GSource * source, GPollFD * fd);

```

```

extern guint g_source_attach(GSource * source, GMainContext *
context);
extern void g_source_destroy(GSource * source);
extern gboolean g_source_get_can_recurse(GSource * source);
extern GMainContext *g_source_get_context(GSource * source);
extern void g_source_get_current_time(GSource * source,
GTimeVal * timeval);
extern guint g_source_get_id(GSource * source);
extern const char *g_source_get_name(GSource * source);
extern gint g_source_get_priority(GSource * source);
extern gint64 g_source_get_time(GSource * source);
extern gboolean g_source_is_destroyed(GSource * source);
extern GSource *g_source_new(GSourceFuncs * source_funcs,
guint struct_size);
extern GSource *g_source_ref(GSource * source);
extern gboolean g_source_remove(guint tag);
extern gboolean g_source_remove_by_funcs_user_data(GSourceFuncs *
funcs,
gpointer user_data);
extern gboolean g_source_remove_by_user_data(gpointer user_data);
extern void g_source_remove_child_source(GSource * source,
GSource * child_source);
extern void g_source_remove_poll(GSource * source, GPollFD * fd);
extern void g_source_set_callback(GSource * source, GSourceFunc
func,
gpointer data, GDestroyNotify notify);
extern void g_source_set_callback_indirect(GSource * source,
gpointer callback_data,
GSourceCallbackFuncs *
callback_funcs);
extern void g_source_set_can_recurse(GSource * source,
gboolean can_recurse);
extern void g_source_set_funcs(GSource * source, GSourceFuncs *
funcs);
extern void g_source_set_name(GSource * source, const char *name);
extern void g_source_set_name_by_id(guint tag, const char *name);
extern void g_source_set_priority(GSource * source, gint priority);
extern void g_source_unref(GSource * source);
extern guint g_spaced_primes_closest(guint num);
extern gboolean g_spawn_async(const gchar * working_directory,
gchar * *argv, gchar * *envp,
GSpawnFlags flags,
GSpawnChildSetupFunc child_setup,
gpointer user_data, GPid * child_pid,
GError * *error);
extern gboolean g_spawn_async_with_pipes(const gchar *
working_directory,
gchar * *argv, gchar * *envp,
GSpawnFlags flags,
GSpawnChildSetupFunc
child_setup,
gpointer user_data,
GPid * child_pid,
gint * standard_input,
gint * standard_output,
gint * standard_error,
GError * *error);
extern void g_spawn_close_pid(GPid pid);
extern gboolean g_spawn_command_line_async(const gchar *
command_line,
GError * *error);
extern gboolean g_spawn_command_line_sync(const gchar *
command_line,
gchar * *standard_output,
gchar * *standard_error,
gint * exit_status,

```

```

        GError * *error);
extern GQuark g_spawn_error_quark(void);
extern gboolean g_spawn_sync(const gchar * working_directory,
                             gchar * *argv, gchar * *envp,
                             GSpawnFlags flags,
                             GSpawnChildSetupFunc child_setup,
                             gpointer user_data, gchar *
*standard_output,
                             gchar * *standard_error, gint *
exit_status,
                             GError * *error);
extern void g_static_mutex_free(GStaticMutex * mutex);
extern GMutex *g_static_mutex_get_mutex_impl(GMutex * *mutex);
extern void g_static_mutex_init(GStaticMutex * mutex);
extern void g_static_private_free(GStaticPrivate * private_key);
extern gpointer g_static_private_get(GStaticPrivate * private_key);
extern void g_static_private_init(GStaticPrivate * private_key);
extern void g_static_private_set(GStaticPrivate * private_key,
                                gpointer data, GDestroyNotify notify);
extern void g_static_rec_mutex_free(GStaticRecMutex * mutex);
extern void g_static_rec_mutex_init(GStaticRecMutex * mutex);
extern void g_static_rec_mutex_lock(GStaticRecMutex * mutex);
extern void g_static_rec_mutex_lock_full(GStaticRecMutex * mutex,
                                          guint depth);
extern gboolean g_static_rec_mutex_trylock(GStaticRecMutex *
mutex);
extern void g_static_rec_mutex_unlock(GStaticRecMutex * mutex);
extern guint g_static_rec_mutex_unlock_full(GStaticRecMutex *
mutex);
extern void g_static_rw_lock_free(GStaticRWLock * lock);
extern void g_static_rw_lock_init(GStaticRWLock * lock);
extern void g_static_rw_lock_reader_lock(GStaticRWLock * lock);
extern gboolean g_static_rw_lock_reader_trylock(GStaticRWLock *
lock);
extern void g_static_rw_lock_reader_unlock(GStaticRWLock * lock);
extern void g_static_rw_lock_writer_lock(GStaticRWLock * lock);
extern gboolean g_static_rw_lock_writer_trylock(GStaticRWLock *
lock);
extern void g_static_rw_lock_writer_unlock(GStaticRWLock * lock);
extern gchar *g_strdup(gchar * dest, const gchar * src);
extern gboolean g_str_equal(gconstpointer v1, gconstpointer v2);
extern gboolean g_str_has_prefix(const gchar * str, const gchar *
prefix);
extern gboolean g_str_has_suffix(const gchar * str, const gchar *
suffix);
extern guint g_str_hash(gconstpointer v);
extern gchar *g_strcanon(gchar * string, const gchar * valid_chars,
                        gchar substitutor);
extern gint g_strcasecmp(const char *s1, const char *s2);
extern gchar *g_strchomp(gchar * string);
extern gchar *g_strchug(gchar * string);
extern int g_strcmp0(const char *str1, const char *str2);
extern gchar *g_strcompress(const gchar * source);
extern gchar *g_strconcat(const gchar * string1, ...);
extern gchar *g_strdelimit(gchar * string, const gchar * delimiters,
                           gchar new_delimiter);
extern gchar *g_strdown(gchar * string);
extern gchar *g_strdup(const gchar * str);
extern gchar *g_strdup_printf(const gchar * format, ...);
extern gchar *g_strdup_vprintf(const gchar * format, va_list args);
extern gchar **g_strdupv(gchar * *str_array);
extern const gchar *g_strerror(gint errnum);
extern gchar *g_strescape(const gchar * source, const gchar *
exceptions);
extern void g_strfreev(gchar * *str_array);

```

```

extern GString *g_string_append(GString * string, const gchar *
val);
extern GString *g_string_append_c(GString * string, gchar c);
extern GString *g_string_append_len(GString * string, const gchar
* val,
                                gssize len);
extern void g_string_append_printf(GString * string, const gchar *
format,
                                ...);
extern GString *g_string_append_unichar(GString * string, gunichar
wc);
extern GString *g_string_append_uri_escaped(GString * string,
const gchar * unescaped,
const gchar *
reserved_chars_allowed,
gboolean allow_utf8);
extern void g_string_append_vprintf(GString * string, const gchar
* format,
                                va_list args);
extern GString *g_string_ascii_down(GString * string);
extern GString *g_string_ascii_up(GString * string);
extern GString *g_string_assign(GString * string, const gchar *
rval);
extern void g_string_chunk_clear(GStringChunk * chunk);
extern void g_string_chunk_free(GStringChunk * chunk);
extern gchar *g_string_chunk_insert(GStringChunk * chunk,
const gchar * string);
extern gchar *g_string_chunk_insert_const(GStringChunk * chunk,
const gchar * string);
extern gchar *g_string_chunk_insert_len(GStringChunk * chunk,
const gchar * string, gssize len);
extern GStringChunk *g_string_chunk_new(gsize size);
extern GString *g_string_down(GString * string);
extern gboolean g_string_equal(const GString * v, const GString *
v2);
extern GString *g_string_erase(GString * string, gssize pos, gssize
len);
extern gchar *g_string_free(GString * string, gboolean
free_segment);
extern guint g_string_hash(const GString * str);
extern GString *g_string_insert(GString * string, gssize pos,
const gchar * val);
extern GString *g_string_insert_c(GString * string, gssize pos,
gchar c);
extern GString *g_string_insert_len(GString * string, gssize pos,
const gchar * val, gssize len);
extern GString *g_string_insert_unichar(GString * string, gssize
pos,
                                gunichar wc);
extern GString *g_string_new(const gchar * init);
extern GString *g_string_new_len(const gchar * init, gssize len);
extern GString *g_string_overwrite(GString * string, gsize pos,
const gchar * val);
extern GString *g_string_overwrite_len(GString * string, gsize pos,
const gchar * val, gssize len);
extern GString *g_string_prepend(GString * string, const gchar *
val);
extern GString *g_string_prepend_c(GString * string, gchar c);
extern GString *g_string_prepend_len(GString * string, const gchar
* val,
                                gssize len);
extern GString *g_string_prepend_unichar(GString * string, gunichar
wc);
extern void g_string_printf(GString * string, const gchar *
format, ...);
extern GString *g_string_set_size(GString * string, gsize len);

```

```

extern GString *g_string_sized_new(gsize dfl_size);
extern GString *g_string_truncate(GString * string, gsize len);
extern GString *g_string_up(GString * string);
extern void g_string_vprintf(GString * string, const gchar * format,
                             va_list args);
extern const gchar *g_strip_context(const gchar * msgid,
                                    const gchar * msgval);
extern gchar *g_strjoin(const gchar * separator, ...);
extern gchar *g_strjoinv(const gchar * separator, gchar *
                          *str_array);
extern gsize g_strlcat(gchar * dest, const gchar * src, gsize
dest_size);
extern gsize g_strlcpy(gchar * dest, const gchar * src, gsize
dest_size);
extern gint g_strncasecmp(const char *s1, const char *s2, guint n);
extern gchar *g_strndup(const gchar * str, gsize n);
extern gchar *g_strnfill(gsize length, gchar fill_char);
extern gchar *g_strreverse(gchar * string);
extern gchar *g_strrstr(const gchar * haystack, const gchar *
needle);
extern gchar *g_strrstr_len(const gchar * haystack, gssize
haystack_len,
                           const gchar * needle);
extern const gchar *g_strsignal(gint signum);
extern gchar **g_strsplit(const gchar * string, const gchar *
delimiter,
                          gint max_tokens);
extern gchar **g_strsplit_set(const gchar * string,
                              const gchar * delimiters, gint max_tokens);
extern gchar *g_strstr_len(const gchar * haystack, gssize
haystack_len,
                           const gchar * needle);
extern gdouble g_strtod(const gchar * nptr, gchar * *endptr);
extern gchar *g_strup(gchar * string);
extern guint g_strv_length(gchar * *str_array);
extern void g_test_add_data_func(const char *testpath,
                                 gconstpointer test_data,
                                 GTestDataFunc test_func);
extern void g_test_add_func(const char *testpath, GTestFunc
test_func);
extern void g_test_add_vtable(const char *testpath, gsize data_size,
                              gconstpointer test_data,
                              GTestFixtureFunc data_setup,
                              GTestFixtureFunc data_test,
                              GTestFixtureFunc data_teardown);
extern void g_test_bug(const char *bug_uri_snippet);
extern void g_test_bug_base(const char *uri_pattern);
extern const GTestConfig *const g_test_config_vars;
extern GTestCase *g_test_create_case(const char *test_name,
                                      gsize data_size,
                                      gconstpointer test_data,
                                      GTestFixtureFunc data_setup,
                                      GTestFixtureFunc data_test,
                                      GTestFixtureFunc data_teardown);
extern GTestSuite *g_test_create_suite(const char *suite_name);
extern void g_test_fail(void);
extern GTestSuite *g_test_get_root(void);
extern void g_test_init(int *argc, char ***argv, ...);
extern void g_test_log_buffer_free(GTestLogBuffer * tbuffer);
extern GTestLogBuffer *g_test_log_buffer_new(void);
extern GTestLogMsg *g_test_log_buffer_pop(GTestLogBuffer *
tbuffer);
extern void g_test_log_buffer_push(GTestLogBuffer * tbuffer, guint
n_bytes,
                                  const unsigned char *bytes);
extern void g_test_log_msg_free(GTestLogMsg * tmsg);

```

```

extern void g_test_log_set_fatal_handler(GTestLogFatalFunc
log_func,
                                gpointer user_data);
extern const char *g_test_log_type_name(GTestLogType log_type);
extern void g_test_maximized_result(double maximized_quantity,
                                const char *format, ...);
extern void g_test_message(const char *format, ...);
extern void g_test_minimized_result(double minimized_quantity,
                                const char *format, ...);
extern void g_test_queue_destroy(GDestroyNotify destroy_func,
                                gpointer destroy_data);
extern void g_test_queue_free(gpointer gfree_pointer);
extern double g_test_rand_double(void);
extern double g_test_rand_double_range(double range_start,
                                double range_end);
extern gint32 g_test_rand_int(void);
extern gint32 g_test_rand_int_range(gint32 begin, gint32 end);
extern int g_test_run(void);
extern int g_test_run_suite(GTestSuite * suite);
extern void g_test_suite_add(GTestSuite * suite, GTestCase *
test_case);
extern void g_test_suite_add_suite(GTestSuite * suite,
                                GTestSuite * nestedsuite);
extern double g_test_timer_elapsed(void);
extern double g_test_timer_last(void);
extern void g_test_timer_start(void);
extern void g_test_trap_assertions(const char *domain, const char
*file,
                                int line, const char *func,
                                guint64 assertion_flags,
                                const char *pattern);
extern gboolean g_test_trap_fork(guint64 usec_timeout,
                                GTestTrapFlags test_trap_flags);
extern gboolean g_test_trap_has_passed(void);
extern gboolean g_test_trap_reached_timeout(void);
extern GThread *g_thread_create(GThreadFunc func, void *data,
                                gboolean joinable, GError * *error);
extern GThread *g_thread_create_full(GThreadFunc func, gpointer
data,
                                gulong stack_size, gboolean
joinable,
                                gboolean bound,
                                GThreadPriority priority,
                                GError * *error);
extern GQuark g_thread_error_quark(void);
extern void g_thread_exit(gpointer retval);
extern void g_thread_foreach(GFunc thread_func, gpointer user_data);
extern GThreadFunctions g_thread_functions_for_glib_use;
extern gboolean g_thread_get_initialized(void);
extern guint64 *g_thread_gettime(void);
extern void g_thread_init(GThreadFunctions * vtable);
extern void g_thread_init_with_errorcheck_mutexes(GThreadFunctions
*
                                vtable);
extern gpointer g_thread_join(GThread * thread);
extern GThread *g_thread_new(const char *name, GThreadFunc func,
                                void *data);
extern void g_thread_pool_free(GThreadPool * pool, gboolean
immediate,
                                gboolean wait);
extern guint g_thread_pool_get_max_idle_time(void);
extern gint g_thread_pool_get_max_threads(GThreadPool * pool);
extern gint g_thread_pool_get_max_unused_threads(void);
extern guint g_thread_pool_get_num_threads(GThreadPool * pool);
extern guint g_thread_pool_get_num_unused_threads(void);

```



```

extern GThreadPool *g_thread_pool_new(GFunc func, gpointer
user_data,
                                gint max_threads, gboolean
exclusive,
                                GError * *error);
extern gboolean g_thread_pool_push(GThreadPool * pool, gpointer
data,
                                GError * *error);
extern void g_thread_pool_set_max_idle_time(guint interval);
extern gboolean g_thread_pool_set_max_threads(GThreadPool * pool,
                                gint max_threads,
                                GError * *error);
extern void g_thread_pool_set_max_unused_threads(gint max_threads);
extern void g_thread_pool_set_sort_function(GThreadPool * pool,
                                GCompareDataFunc func,
                                gpointer user_data);
extern void g_thread_pool_stop_unused_threads(void);
extern guint g_thread_pool_unprocessed(GThreadPool * pool);
extern GThread *g_thread_ref(GThread * thread);
extern GThread *g_thread_self(void);
extern void g_thread_set_priority(GThread * thread,
                                GThreadPriority priority);
extern GThread *g_thread_try_new(const char *name, GThreadFunc func,
                                void *data, GError * *error);
extern void g_thread_unref(GThread * thread);
extern gboolean g_thread_use_default_impl;
extern void g_thread_yield(void);
extern gboolean g_threads_got_initialized;
extern void g_time_val_add(GTimeVal * time_, gulong microseconds);
extern gboolean g_time_val_from_iso8601(const gchar * iso_date,
                                GTimeVal * time);
extern gchar *g_time_val_to_iso8601(GTimeVal * time);
extern gint g_time_zone_adjust_time(GTimeZone * tz, GTimeType type,
                                gint64 * time_);
extern gint g_time_zone_find_interval(GTimeZone * tz, GTimeType
type,
                                gint64 time_);
extern const gchar *g_time_zone_get_abbreviation(GTimeZone * tz,
                                gint interval);
extern gint32 g_time_zone_get_offset(GTimeZone * tz, gint interval);
extern gboolean g_time_zone_is_dst(GTimeZone * tz, gint interval);
extern GTimeZone *g_time_zone_new(const gchar * identifier);
extern GTimeZone *g_time_zone_new_local(void);
extern GTimeZone *g_time_zone_new_utc(void);
extern GTimeZone *g_time_zone_ref(GTimeZone * tz);
extern void g_time_zone_unref(GTimeZone * tz);
extern guint g_timeout_add(guint32 interval, GSourceFunc function,
                                gpointer data);
extern guint g_timeout_add_full(gint priority, guint interval,
                                GSourceFunc function, gpointer data,
                                GDestroyNotify notify);
extern guint g_timeout_add_seconds(guint interval, GSourceFunc
function,
                                void *data);
extern guint g_timeout_add_seconds_full(gint priority, guint
interval,
                                GSourceFunc function, void *data,
                                GDestroyNotify notify);
extern GSourceFuncs g_timeout_funcs;
extern GSource *g_timeout_source_new(guint interval);
extern GSource *g_timeout_source_new_seconds(guint interval);
extern void g_timer_continue(GTimer * timer);
extern void g_timer_destroy(GTimer * timer);
extern gdouble g_timer_elapsed(GTimer * timer, gulong *
microseconds);
extern GTimer *g_timer_new(void);

```



```

extern void g_timer_reset(GTimer * timer);
extern void g_timer_start(GTimer * timer);
extern void g_timer_stop(GTimer * timer);
extern guint g_trash_stack_height(GTrashStack * *stack_p);
extern gpointer g_trash_stack_peek(GTrashStack * *stack_p);
extern gpointer g_trash_stack_pop(GTrashStack * *stack_p);
extern void g_trash_stack_push(GTrashStack * *stack_p, gpointer
data_p);
extern void g_tree_destroy(GTree * tree);
extern void g_tree_foreach(GTree * tree, GTraverseFunc func,
gpointer user_data);
extern gint g_tree_height(GTree * tree);
extern void g_tree_insert(GTree * tree, gpointer key, gpointer
value);
extern gpointer g_tree_lookup(GTree * tree, gconstpointer key);
extern gboolean g_tree_lookup_extended(GTree * tree,
gconstpointer lookup_key,
gpointer * orig_key,
gpointer * value);
extern GTree *g_tree_new(GCompareFunc key_compare_func);
extern GTree *g_tree_new_full(GCompareDataFunc key_compare_func,
gpointer key_compare_data,
GDestroyNotify key_destroy_func,
GDestroyNotify value_destroy_func);
extern GTree *g_tree_new_with_data(GCompareDataFunc
key_compare_func,
gpointer key_compare_data);
extern gint g_tree_nnodes(GTree * tree);
extern GTree *g_tree_ref(GTree * tree);
extern gboolean g_tree_remove(GTree * tree, gconstpointer key);
extern void g_tree_replace(GTree * tree, gpointer key, gpointer
value);
extern gpointer g_tree_search(GTree * tree, GCompareFunc
search_func,
gconstpointer user_data);
extern gboolean g_tree_steal(GTree * tree, gconstpointer key);
extern void g_tree_traverse(GTree * tree, GTraverseFunc
traverse_func,
GTraverseType traverse_type, void
*user_data);
extern void g_tree_unref(GTree * tree);
extern gpointer g_try_malloc(gulong n_bytes);
extern void *g_try_malloc0(gsize n_bytes);
extern void *g_try_malloc0_n(gsize n_blocks, gsize n_block_bytes);
extern void *g_try_malloc_n(gsize n_blocks, gsize n_block_bytes);
extern gpointer g_try_realloc(gpointer mem, gulong n_bytes);
extern void *g_try_realloc_n(void *mem, gsize n_blocks,
gsize n_block_bytes);
extern void g_tuples_destroy(GTuples * tuples);
extern gpointer g_tuples_index(GTuples * tuples, gint index_, gint
field);
extern gunichar2 *g_ucs4_to_utf16(const gunichar * str, glong len,
glong * items_read,
glong * items_written, GError *
*error);
extern gchar *g_ucs4_to_utf8(const gunichar * str, glong len,
glong * items_read, glong * items_written,
GError * *error);
extern GUnicodeBreakType g_unichar_break_type(gunichar c);
extern gint g_unichar_combining_class(gunichar uc);
extern gboolean g_unichar_compose(gunichar a, gunichar b, gunichar
* ch);
extern gboolean g_unichar_decompose(gunichar ch, gunichar * a,
gunichar * b);
extern gint g_unichar_digit_value(gunichar c);
extern gsize g_unichar_fully_decompose(gunichar ch, gboolean compat,

```

```

        gunichar * result,
        gsize result_len);
extern gboolean g_unichar_get_mirror_char(gunichar ch,
        gunichar * mirrored_ch);
extern GUnicodeScript g_unichar_get_script(gunichar ch);
extern gboolean g_unichar_isalnum(gunichar c);
extern gboolean g_unichar_isalpha(gunichar c);
extern gboolean g_unichar_iscntrl(gunichar c);
extern gboolean g_unichar_isdefined(gunichar c);
extern gboolean g_unichar_isdigit(gunichar c);
extern gboolean g_unichar_isgraph(gunichar c);
extern gboolean g_unichar_islower(gunichar c);
extern gboolean g_unichar_ismark(gunichar c);
extern gboolean g_unichar_isprint(gunichar c);
extern gboolean g_unichar_ispunct(gunichar c);
extern gboolean g_unichar_isspace(gunichar c);
extern gboolean g_unichar_istitle(gunichar c);
extern gboolean g_unichar_isupper(gunichar c);
extern gboolean g_unichar_iswide(gunichar c);
extern gboolean g_unichar_iswide_cjk(gunichar c);
extern gboolean g_unichar_isxdigit(gunichar c);
extern gboolean g_unichar_iszerowidth(gunichar c);
extern gint g_unichar_to_utf8(gunichar c, gchar * outbuf);
extern gunichar g_unichar_tolower(gunichar c);
extern gunichar g_unichar_totitle(gunichar c);
extern gunichar g_unichar_toupper(gunichar c);
extern GUnicodeType g_unichar_type(gunichar c);
extern gboolean g_unichar_validate(gunichar ch);
extern gint g_unichar_xdigit_value(gunichar c);
extern gunichar *g_unicode_canonical_decomposition(gunichar ch,
        gsize * result_len);

extern void g_unicode_canonical_ordering(gunichar * string, gsize
len);
extern GUnicodeScript g_unicode_script_from_iso15924(guint32
iso15924);
extern guint32 g_unicode_script_to_iso15924(GUnicodeScript script);
extern void g_unsetenv(const gchar * variable);
extern char *g_uri_escape_string(const char *unescaped,
        const char *reserved_chars_allowed,
        gboolean allow_utf8);
extern gchar **g_uri_list_extract_uris(const gchar * uri_list);
extern char *g_uri_parse_scheme(const char *uri);
extern char *g_uri_unescape_segment(const char *escaped_string,
        const char *escaped_string_end,
        const char *illegal_characters);
extern char *g_uri_unescape_string(const char *escaped_string,
        const char *illegal_characters);
extern void g_usleep(gulong microseconds);
extern gunichar *g_utf16_to_ucs4(const gunichar2 * str, glong len,
        glong * items_read, glong *
items_written,
        GError * *error);
extern gchar *g_utf16_to_utf8(const gunichar2 * str, glong len,
        glong * items_read, glong * items_written,
        GError * *error);
extern gchar *g_utf8_casefold(const gchar * str, gssize len);
extern gint g_utf8_collate(const gchar * str1, const gchar * str2);
extern gchar *g_utf8_collate_key(const gchar * str, gssize len);
extern gchar *g_utf8_collate_key_for_filename(const gchar * str,
        gssize len);
extern gchar *g_utf8_find_next_char(const gchar * p, const gchar *
end);
extern gchar *g_utf8_find_prev_char(const gchar * str, const gchar
* p);
extern gunichar g_utf8_get_char(const gchar * p);

```

```

extern gunichar g_utf8_get_char_validated(const gchar * p, gssize
max_len);
extern gchar *g_utf8_normalize(const gchar * str, gssize len,
                               GNormalizeMode mode);
extern gchar *g_utf8_offset_to_pointer(const gchar * str, glong
offset);
extern glong g_utf8_pointer_to_offset(const gchar * str,
                                      const gchar * pos);
extern gchar *g_utf8_prev_char(const gchar * p);
extern const gchar *const g_utf8_skip;
extern gchar *g_utf8_strchr(const gchar * p, gssize len, gunichar
c);
extern gchar *g_utf8_strdown(const gchar * str, gssize len);
extern glong g_utf8_strlen(const gchar * p, gssize max);
extern gchar *g_utf8_strncpy(gchar * dest, const gchar * src, gsize
n);
extern gchar *g_utf8_strrchr(const gchar * p, gssize len, gunichar
c);
extern gchar *g_utf8_strreverse(const gchar * str, gssize len);
extern gchar *g_utf8_strup(const gchar * str, gssize len);
extern gchar *g_utf8_substring(const gchar * str, glong start_pos,
                               glong end_pos);
extern gunichar *g_utf8_to_ucs4(const gchar * str, glong len,
                                glong * items_read, glong *
items_written,
                                GError * *error);
extern gunichar *g_utf8_to_ucs4_fast(const gchar * str, glong len,
                                     glong * items_written);
extern gunichar2 *g_utf8_to_utf16(const gchar * str, glong len,
                                   glong * items_read,
                                   glong * items_written, GError *
*error);
extern gboolean g_utf8_validate(const gchar * str, gssize max_len,
                                const gchar * *end);
extern void g_variant_builder_add(GVariantBuilder * builder,
                                  const gchar * format_string, ...);
extern void g_variant_builder_add_parsed(GVariantBuilder * builder,
                                          const gchar * format, ...);
extern void g_variant_builder_add_value(GVariantBuilder * builder,
                                         GVariant * value);
extern void g_variant_builder_clear(GVariantBuilder * builder);
extern void g_variant_builder_close(GVariantBuilder * builder);
extern GVariant *g_variant_builder_end(GVariantBuilder * builder);
extern void g_variant_builder_init(GVariantBuilder * builder,
                                   const GVariantType * type);
extern GVariantBuilder *g_variant_builder_new(const GVariantType *
type);
extern void g_variant_builder_open(GVariantBuilder * builder,
                                   const GVariantType * type);
extern GVariantBuilder *g_variant_builder_ref(GVariantBuilder *
builder);
extern void g_variant_builder_unref(GVariantBuilder * builder);
extern GVariant *g_variant_byteswap(GVariant * value);
extern GVariantClass g_variant_classify(GVariant * value);
extern gint g_variant_compare(gconstpointer one, gconstpointer two);
extern gchar *g_variant_dup_bytestring(GVariant * value, gsize *
length);
extern gchar **g_variant_dup_bytestring_array(GVariant * value,
                                               gsize * length);
extern gchar **g_variant_dup_objv(GVariant * value, gsize * length);
extern gchar *g_variant_dup_string(GVariant * value, gsize *
length);
extern gchar **g_variant_dup_strv(GVariant * value, gsize * length);
extern gboolean g_variant_equal(gconstpointer one, gconstpointer
two);

```

```

extern void g_variant_get(GVariant * value, const gchar *
format_string,
    ...);
extern gboolean g_variant_get_boolean(GVariant * value);
extern gchar g_variant_get_byte(GVariant * value);
extern const gchar *g_variant_get_bytestring(GVariant * value);
extern const gchar **g_variant_get_bytestring_array(GVariant *
value,
    gsize * length);
extern void g_variant_get_child(GVariant * value, gsize index_,
    const gchar * format_string, ...);
extern GVariant *g_variant_get_child_value(GVariant * value, gsize
index_);
extern gconstpointer g_variant_get_data(GVariant * value);
extern gdouble g_variant_get_double(GVariant * value);
extern gconstpointer g_variant_get_fixed_array(GVariant * value,
    gsize * n_elements,
    gsize element_size);
extern gint32 g_variant_get_handle(GVariant * value);
extern gint16 g_variant_get_int16(GVariant * value);
extern gint32 g_variant_get_int32(GVariant * value);
extern gint64 g_variant_get_int64(GVariant * value);
extern GVariant *g_variant_get_maybe(GVariant * value);
extern GVariant *g_variant_get_normal_form(GVariant * value);
extern const gchar **g_variant_get_objv(GVariant * value, gsize *
length);
extern gsize g_variant_get_size(GVariant * value);
extern const gchar *g_variant_get_string(GVariant * value, gsize *
length);
extern const gchar **g_variant_get_strv(GVariant * value, gsize *
length);
extern const GVariantType *g_variant_get_type(GVariant * value);
extern const gchar *g_variant_get_type_string(GVariant * value);
extern guint16 g_variant_get_uint16(GVariant * value);
extern guint32 g_variant_get_uint32(GVariant * value);
extern guint64 g_variant_get_uint64(GVariant * value);
extern void g_variant_get_va(GVariant * value, const gchar *
format_string,
    const gchar * *endptr, va_list * app);
extern GVariant *g_variant_get_variant(GVariant * value);
extern guint g_variant_hash(gconstpointer value);
extern gboolean g_variant_is_container(GVariant * value);
extern gboolean g_variant_is_floating(GVariant * value);
extern gboolean g_variant_is_normal_form(GVariant * value);
extern gboolean g_variant_is_object_path(const gchar * string);
extern gboolean g_variant_is_of_type(GVariant * value,
    const GVariantType * type);
extern gboolean g_variant_is_signature(const gchar * string);
extern GVariantIter *g_variant_iter_copy(GVariantIter * iter);
extern void g_variant_iter_free(GVariantIter * iter);
extern gsize g_variant_iter_init(GVariantIter * iter, GVariant *
value);
extern gboolean g_variant_iter_loop(GVariantIter * iter,
    const gchar * format_string, ...);
extern gsize g_variant_iter_n_children(GVariantIter * iter);
extern GVariantIter *g_variant_iter_new(GVariant * value);
extern gboolean g_variant_iter_next(GVariantIter * iter,
    const gchar * format_string, ...);
extern GVariant *g_variant_iter_next_value(GVariantIter * iter);
extern gboolean g_variant_lookup(GVariant * dictionary, const gchar
* key,
    const gchar * format_string, ...);
extern GVariant *g_variant_lookup_value(GVariant * dictionary,
    const gchar * key,
    const GVariantType *
expected_type);

```

```

extern gsize g_variant_n_children(GVariant * value);
extern GVariant *g_variant_new(const gchar * format_string, ...);
extern GVariant *g_variant_new_array(const GVariantType *
child_type,
                                GVariant * const *children,
                                gsize n_children);
extern GVariant *g_variant_new_boolean(gboolean value);
extern GVariant *g_variant_new_byte(guchar value);
extern GVariant *g_variant_new_bytestring(const gchar * string);
extern GVariant *g_variant_new_bytestring_array(const gchar * const
*strv,
                                gssize length);
extern GVariant *g_variant_new_dict_entry(GVariant * key,
                                GVariant * value);
extern GVariant *g_variant_new_double(gdouble value);
extern GVariant *g_variant_new_fixed_array(const GVariantType *
element_type,
                                gconstpointer elements,
                                gsize n_elements,
                                gsize element_size);
extern GVariant *g_variant_new_from_data(const GVariantType * type,
                                gconstpointer data, gsize size,
                                gboolean trusted,
                                GDestroyNotify notify,
                                void *user_data);
extern GVariant *g_variant_new_handle(gint32 value);
extern GVariant *g_variant_new_int16(gint16 value);
extern GVariant *g_variant_new_int32(gint32 value);
extern GVariant *g_variant_new_int64(gint64 value);
extern GVariant *g_variant_new_maybe(const GVariantType *
child_type,
                                GVariant * child);
extern GVariant *g_variant_new_object_path(const gchar *
object_path);
extern GVariant *g_variant_new_objv(const gchar * const *strv,
                                gssize length);
extern GVariant *g_variant_new_parsed(const gchar * format, ...);
extern GVariant *g_variant_new_parsed_va(const gchar * format,
                                va_list * app);
extern GVariant *g_variant_new_signature(const gchar * signature);
extern GVariant *g_variant_new_string(const gchar * string);
extern GVariant *g_variant_new_strv(const gchar * const *strv,
                                gssize length);
extern GVariant *g_variant_new_tuple(GVariant * const *children,
                                gsize n_children);
extern GVariant *g_variant_new_uint16(guint16 value);
extern GVariant *g_variant_new_uint32(guint32 value);
extern GVariant *g_variant_new_uint64(guint64 value);
extern GVariant *g_variant_new_va(const gchar * format_string,
                                const gchar * *endptr, va_list * app);
extern GVariant *g_variant_new_variant(GVariant * value);
extern GVariant *g_variant_parse(const GVariantType * type,
                                const char *text, const char *limit,
                                const char **endptr, GError * *error);
extern GQuark g_variant_parser_get_error_quark(void);
extern gchar *g_variant_print(GVariant * value, gboolean
type_annotate);
extern GString *g_variant_print_string(GVariant * value, GString *
string,
                                gboolean type_annotate);
extern GVariant *g_variant_ref(GVariant * value);
extern GVariant *g_variant_ref_sink(GVariant * value);
extern void g_variant_store(GVariant * value, void *data);
extern GVariant *g_variant_take_ref(GVariant * value);
extern const GVariantType *g_variant_type_checked_(const gchar *);
extern GVariantType *g_variant_type_copy(const GVariantType * type);

```

```

extern gchar *g_variant_type_dup_string(const GVariantType * type);
extern      const      GVariantType      *g_variant_type_element(const
GVariantType *
                                type);
extern gboolean g_variant_type_equal(gconstpointer type1,
                                gconstpointer type2);
extern const GVariantType *g_variant_type_first(const GVariantType
* type);
extern void g_variant_type_free(GVariantType * type);
extern gsize g_variant_type_get_string_length(const GVariantType *
type);
extern guint g_variant_type_hash(gconstpointer type);
extern gboolean g_variant_type_is_array(const GVariantType * type);
extern gboolean g_variant_type_is_basic(const GVariantType * type);
extern gboolean g_variant_type_is_container(const GVariantType *
type);
extern gboolean g_variant_type_is_definite(const GVariantType *
type);
extern gboolean g_variant_type_is_dict_entry(const GVariantType *
type);
extern gboolean g_variant_type_is_maybe(const GVariantType * type);
extern gboolean g_variant_type_is_subtype_of(const GVariantType *
type,
                                const GVariantType *
                                supertype);
extern gboolean g_variant_type_is_tuple(const GVariantType * type);
extern gboolean g_variant_type_is_variant(const GVariantType *
type);
extern const GVariantType *g_variant_type_key(const GVariantType *
type);
extern gsize g_variant_type_n_items(const GVariantType * type);
extern GVariantType *g_variant_type_new(const gchar * type_string);
extern GVariantType *g_variant_type_new_array(const GVariantType *
element);
extern      GVariantType      *g_variant_type_new_dict_entry(const
GVariantType *
                                key,
                                const GVariantType *
                                value);
extern GVariantType *g_variant_type_new_maybe(const GVariantType *
element);
extern GVariantType *g_variant_type_new_tuple(const GVariantType *
const *items, gint length);
extern const GVariantType *g_variant_type_next(const GVariantType
* type);
extern const char *g_variant_type_peek_string(const GVariantType *
type);
extern gboolean g_variant_type_string_is_valid(const gchar *
type_string);
extern gboolean g_variant_type_string_scan(const gchar * string,
                                const gchar * limit,
                                const gchar * *endptr);
extern const GVariantType *g_variant_type_value(const GVariantType
* type);
extern void g_variant_unref(GVariant * value);
extern void g_warn_message(const char *domain, const char *file,
int line,
                                const char *func, const char *warnexpr);
extern const guint glib_binary_age;
extern const gchar *glib_check_version(guint required_major,
                                guint required_minor,
                                guint required_micro);
extern const guint glib_interface_age;
extern const guint glib_major_version;
extern GMemVTable *glib_mem_profiler_table;
extern const guint glib_micro_version;

```

```
extern const quint glib_minor_version;
```

17.3.2 glib-2.0/glib/gi18n.h

```
#define __G_I18N_H__
#define NC_(Context,String)      (String)
#define N_(String)              (String)
#define _(String)               gettext (String)
#define Q_(String)              g_strip_context ((String), gettext (String))
```

17.3.3 glib-2.0/glib/gprintf.h

```
extern gint g_fprintf(FILE * file, const gchar * format, ...);
extern gint g_printf(const gchar * format, ...);
extern gint g_snprintf(gchar * string, gulong n, const gchar *
format,
...);
extern gint g_sprintf(gchar * string, const gchar * format, ...);
extern gint g_vasprintf(gchar * *string, const gchar * format,
va_list args);
extern gint g_vfprintf(FILE * file, const gchar * format, va_list
args);
extern gint g_vprintf(const gchar * format, va_list args);
extern gint g_vsnprintf(gchar * string, gulong n, const gchar *
format,
va_list args);
extern gint g_vsprintf(gchar * string, const gchar * format, va_list
args);
```

17.3.4 glib-2.0/glib/gstdio.h

```
#define g_access      access
#define g_chdir      chdir
#define g_chmod      chmod
#define g_creat      creat
#define g_fopen      fopen
#define g_freopen      freopen
#define g_lstat      lstat
#define g_mkdir      mkdir
#define g_open      open
#define g_remove      remove
#define g_rename      rename
#define g_rmdir      rmdir
#define g_stat      stat
#define g_unlink      unlink
typedef struct stat GStatBuf;
```

17.4 Interface Definitions for libglib-2.0

The interfaces defined on the following pages are included in libglib-2.0 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 17.2 shall behave as described in the referenced base document.

g_cache_value_foreach

Name

`g_cache_value_foreach` — call specified interface for each value in GCache (DEPRECATED)

Synopsis

```
#include <glib-2.0/glib.h>
void g_cache_value_foreach(GCache * cache, GHFunc func, gpointer user_data);
```

Description

The interface `g_cache_value_foreach()` shall behave as described in Glib 2.32 Reference Manual. However, this interface is deprecated and should not be used in newly-written code, because it passes pointers to unspecified internal data structures to *func*. Use `g_cache_key_foreach()` instead.

17.5 Interfaces for libgmodule-2.0

Table 17-74 defines the library name and shared object name for the libgmodule-2.0 library

Table 17-74 libgmodule-2.0 Definition

Library:	libgmodule-2.0
SONAME:	libgmodule-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Glib 2.32] Glib 2.32 Reference Manual

17.5.1 Glib-Dynamic loading of Modules

17.5.1.1 Interfaces for Glib-Dynamic loading of Modules

An LSB conforming implementation shall provide the generic functions for Glib-Dynamic loading of Modules specified in Table 17-75, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-75 libgmodule-2.0 - Glib-Dynamic loading of Modules Function Interfaces

<code>g_module_build_path</code> [Glib 2.32]	<code>g_module_close</code> [Glib 2.32]
<code>g_module_error</code> [Glib 2.32]	<code>g_module_make_resident</code> [Glib 2.32]
<code>g_module_name</code> [Glib 2.32]	<code>g_module_open</code> [Glib 2.32]
<code>g_module_supported</code> [Glib 2.32]	<code>g_module_symbol</code> [Glib 2.32]

17.6 Data Definitions for libgmodule-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libgmodule-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a

convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.6.1 glib-2.0/gmodule.h

```
#define G_MODULE_EXPORT
#define G_MODULE_IMPORT extern

typedef struct _GModule GModule;
typedef enum {
    G_MODULE_BIND_LAZY = 1 << 0,
    G_MODULE_BIND_LOCAL = 1 << 1,
    G_MODULE_BIND_MASK = 0x03
} GModuleFlags;
typedef void (*GModuleUnload) (GModule *);
typedef const gchar *(*GModuleCheckInit) (GModule *);
extern gchar *g_module_build_path(const gchar * directory,
                                  const gchar * module_name);
extern gboolean g_module_close(GModule * module);
extern const gchar *g_module_error(void);
extern void g_module_make_resident(GModule * module);
extern const gchar *g_module_name(GModule * module);
extern GModule *g_module_open(const gchar * file_name, GModuleFlags
flags);
extern gboolean g_module_supported(void);
extern gboolean g_module_symbol(GModule * module,
                                const gchar * symbol_name,
                                gpointer * symbol);
```

17.7 Interfaces for libgobject-2.0

Table 17-76 defines the library name and shared object name for the libgobject-2.0 library

Table 17-76 libgobject-2.0 Definition

Library:	libgobject-2.0
SONAME:	libgobject-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

- [Gobject 2.32] Gobject 2.32 Reference Manual
- [LSB] This Specification

17.7.1 GObject GTypeModule

17.7.1.1 Interfaces for GObject GTypeModule

An LSB conforming implementation shall provide the generic functions for GObject GTypeModule specified in Table 17-77, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-77 libgobject-2.0 - GObject GTypeModule Function Interfaces

g_binding_flags_get_type [Gobject 2.32]	g_binding_get_type [Gobject 2.32]
g_cclosure_marshal_generic [Gobject 2.32]	g_cclosure_marshal_generic_va [LSB]
g_clear_object [Gobject 2.32]	g_object_class_install_properties [Gobject 2.32]
g_object_compat_control [LSB]	g_object_get_type [Gobject 2.32]
g_object_notify_by_pspec [Gobject 2.32]	g_param_spec_variant [Gobject 2.32]
g_signal_accumulator_first_wins [Gobject 2.32]	g_signal_chain_from_overridden_handler [Gobject 2.32]
g_signal_new_class_handler [Gobject 2.32]	g_signal_override_class_handler [Gobject 2.32]
g_signal_set_va_marshaller [LSB]	g_source_set_dummy_callback [Gobject 2.32]
g_type_add_class_private [Gobject 2.32]	g_type_class_get_private [LSB]
g_type_module_add_interface [Gobject 2.32]	g_type_module_get_type [Gobject 2.32]
g_type_module_register_enum [Gobject 2.32]	g_type_module_register_flags [Gobject 2.32]
g_type_module_register_type [Gobject 2.32]	g_type_module_set_name [Gobject 2.32]
g_type_module_unuse [Gobject 2.32]	g_type_module_use [Gobject 2.32]
g_value_dup_variant [Gobject 2.32]	g_value_get_schar [Gobject 2.32]
g_value_get_variant [Gobject 2.32]	g_value_set_boxed_take_ownership [Gobject 2.32]
g_value_set_object_take_ownership [Gobject 2.32]	g_value_set_param_take_ownership [Gobject 2.32]
g_value_set_schar [Gobject 2.32]	g_value_set_string_take_ownership [Gobject 2.32]
g_value_set_variant [Gobject 2.32]	g_value_take_variant [Gobject 2.32]
g_weak_ref_clear [Gobject 2.32]	g_weak_ref_get [Gobject 2.32]
g_weak_ref_init [Gobject 2.32]	g_weak_ref_set [Gobject 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for GObject GTypeModule specified in Table 17-78, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-78 libgobject-2.0 - GObject GTypeModule Deprecated Function Interfaces

g_value_set_boxed_take_ownership [Gobject 2.32]	g_value_set_object_take_ownership [Gobject 2.32]
g_value_set_param_take_ownership [Gobject 2.32]	g_value_set_string_take_ownership [Gobject 2.32]

17.7.2 GObject Enums and Flags

17.7.2.1 Interfaces for GObject Enums and Flags

An LSB conforming implementation shall provide the generic functions for GObject Enums and Flags specified in Table 17-79, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-79 libgobject-2.0 - GObject Enums and Flags Function Interfaces

g_enum_complete_type_info [Gobject 2.32]	g_enum_get_value [Gobject 2.32]
g_enum_get_value_by_name [Gobject 2.32]	g_enum_get_value_by_nick [Gobject 2.32]
g_enum_register_static [Gobject 2.32]	g_flags_complete_type_info [Gobject 2.32]
g_flags_get_first_value [Gobject 2.32]	g_flags_get_value_by_name [Gobject 2.32]
g_flags_get_value_by_nick [Gobject 2.32]	g_flags_register_static [Gobject 2.32]

17.7.3 GObject Signals

17.7.3.1 Interfaces for GObject Signals

An LSB conforming implementation shall provide the generic functions for GObject Signals specified in Table 17-80, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-80 libgobject-2.0 - GObject Signals Function Interfaces

g_signal_accumulator_true_handled [Gobject 2.32]	g_signal_add_emission_hook [Gobject 2.32]
g_signal_chain_from_overridden [Gobject 2.32]	g_signal_connect_closure [Gobject 2.32]
g_signal_connect_closure_by_id [Gobject 2.32]	g_signal_connect_data [Gobject 2.32]

g_signal_connect_object [Gobject 2.32]	g_signal_emit [Gobject 2.32]
g_signal_emit_by_name [Gobject 2.32]	g_signal_emit_valist [Gobject 2.32]
g_signal_emitv [Gobject 2.32]	g_signal_get_invocation_hint [Gobject 2.32]
g_signal_handler_block [Gobject 2.32]	g_signal_handler_disconnect [Gobject 2.32]
g_signal_handler_find [Gobject 2.32]	g_signal_handler_is_connected [Gobject 2.32]
g_signal_handler_unblock [Gobject 2.32]	g_signal_handlers_block_matched [Gobject 2.32]
g_signal_handlers_disconnect_matched [Gobject 2.32]	g_signal_handlers_unblock_matched [Gobject 2.32]
g_signal_has_handler_pending [Gobject 2.32]	g_signal_list_ids [Gobject 2.32]
g_signal_lookup [Gobject 2.32]	g_signal_name [Gobject 2.32]
g_signal_new [Gobject 2.32]	g_signal_new_valist [Gobject 2.32]
g_signal_newv [Gobject 2.32]	g_signal_override_class_closure [Gobject 2.32]
g_signal_parse_name [Gobject 2.32]	g_signal_query [Gobject 2.32]
g_signal_remove_emission_hook [Gobject 2.32]	g_signal_stop_emission [Gobject 2.32]
g_signal_stop_emission_by_name [Gobject 2.32]	g_signal_type_cclosure_new [Gobject 2.32]

17.7.4 Gobject Value arrays

17.7.4.1 Interfaces for Gobject Value arrays

An LSB conforming implementation shall provide the generic functions for Gobject Value arrays specified in Table 17-81, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-81 libgobject-2.0 - Gobject Value arrays Function Interfaces

g_value_array_append [Gobject 2.32]	g_value_array_copy [Gobject 2.32]
g_value_array_free [Gobject 2.32]	g_value_array_get_nth [Gobject 2.32]
g_value_array_insert [Gobject 2.32]	g_value_array_new [Gobject 2.32]
g_value_array_prepend [Gobject 2.32]	g_value_array_remove [Gobject 2.32]
g_value_array_sort [Gobject 2.32]	g_value_array_sort_with_data [Gobject 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Gobject Value arrays specified in Table 17-82, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-82 libgobject-2.0 - Gobject Value arrays Deprecated Function Interfaces

g_value_array_append [Gobject 2.32]	g_value_array_copy [Gobject 2.32]
g_value_array_free [Gobject 2.32]	g_value_array_get_nth [Gobject 2.32]
g_value_array_insert [Gobject 2.32]	g_value_array_new [Gobject 2.32]
g_value_array_prepend [Gobject 2.32]	g_value_array_remove [Gobject 2.32]
g_value_array_sort [Gobject 2.32]	g_value_array_sort_with_data [Gobject 2.32]

17.7.5 Gobject Generic Values

17.7.5.1 Interfaces for Gobject Generic Values

An LSB conforming implementation shall provide the generic functions for Gobject Generic Values specified in Table 17-83, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-83 libgobject-2.0 - Gobject Generic Values Function Interfaces

g_strdup_value_contents [Gobject 2.32]	g_value_array_get_type [Gobject 2.32]
g_value_copy [Gobject 2.32]	g_value_fits_pointer [Gobject 2.32]
g_value_get_type [Gobject 2.32]	g_value_init [Gobject 2.32]
g_value_peek_pointer [Gobject 2.32]	g_value_register_transform_func [Gobject 2.32]
g_value_reset [Gobject 2.32]	g_value_set_instance [LSB]
g_value_transform [Gobject 2.32]	g_value_type_compatible [Gobject 2.32]
g_value_type_transformable [Gobject 2.32]	g_value_unset [Gobject 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Gobject Generic Values specified in Table 17-84, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-84 libgobject-2.0 - GObject Generic Values Deprecated Function Interfaces

g_value_array_get_type [Gobject 2.32]	

17.7.6 GObject Parameters and Values

17.7.6.1 Interfaces for GObject Parameters and Values

An LSB conforming implementation shall provide the generic functions for GObject Parameters and Values specified in Table 17-85, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-85 libgobject-2.0 - GObject Parameters and Values Function Interfaces

g_param_spec_boolean [Gobject 2.32]	g_param_spec_boxed [Gobject 2.32]
g_param_spec_char [Gobject 2.32]	g_param_spec_double [Gobject 2.32]
g_param_spec_enum [Gobject 2.32]	g_param_spec_flags [Gobject 2.32]
g_param_spec_float [Gobject 2.32]	g_param_spec_int [Gobject 2.32]
g_param_spec_int64 [Gobject 2.32]	g_param_spec_long [Gobject 2.32]
g_param_spec_object [Gobject 2.32]	g_param_spec_override [Gobject 2.32]
g_param_spec_param [Gobject 2.32]	g_param_spec_pointer [Gobject 2.32]
g_param_spec_string [Gobject 2.32]	g_param_spec_uchar [Gobject 2.32]
g_param_spec_uint [Gobject 2.32]	g_param_spec_uint64 [Gobject 2.32]
g_param_spec_ulong [Gobject 2.32]	g_param_spec_unichar [Gobject 2.32]
g_param_spec_value_array [Gobject 2.32]	g_value_dup_boxed [Gobject 2.32]
g_value_dup_object [Gobject 2.32]	g_value_dup_param [Gobject 2.32]
g_value_dup_string [Gobject 2.32]	g_value_get_boolean [Gobject 2.32]
g_value_get_boxed [Gobject 2.32]	g_value_get_char [Gobject 2.32]
g_value_get_double [Gobject 2.32]	g_value_get_enum [Gobject 2.32]
g_value_get_flags [Gobject 2.32]	g_value_get_float [Gobject 2.32]
g_value_get_gtype [Gobject 2.32]	g_value_get_int [Gobject 2.32]
g_value_get_int64 [Gobject 2.32]	g_value_get_long [Gobject 2.32]
g_value_get_object [Gobject 2.32]	g_value_get_param [Gobject 2.32]
g_value_get_pointer [Gobject 2.32]	g_value_get_string [Gobject 2.32]
g_value_get_uchar [Gobject 2.32]	g_value_get_uint [Gobject 2.32]
g_value_get_uint64 [Gobject 2.32]	g_value_get_ulong [Gobject 2.32]
g_value_set_boolean [Gobject 2.32]	g_value_set_boxed [Gobject 2.32]
g_value_set_char [Gobject 2.32]	g_value_set_double [Gobject 2.32]

<code>g_value_set_enum</code> [Gobject 2.32]	<code>g_value_set_flags</code> [Gobject 2.32]
<code>g_value_set_float</code> [Gobject 2.32]	<code>g_value_set_gtype</code> [Gobject 2.32]
<code>g_value_set_int</code> [Gobject 2.32]	<code>g_value_set_int64</code> [Gobject 2.32]
<code>g_value_set_long</code> [Gobject 2.32]	<code>g_value_set_object</code> [Gobject 2.32]
<code>g_value_set_param</code> [Gobject 2.32]	<code>g_value_set_pointer</code> [Gobject 2.32]
<code>g_value_set_static_boxed</code> [Gobject 2.32]	<code>g_value_set_static_string</code> [Gobject 2.32]
<code>g_value_set_string</code> [Gobject 2.32]	<code>g_value_set_uchar</code> [Gobject 2.32]
<code>g_value_set_uint</code> [Gobject 2.32]	<code>g_value_set_uint64</code> [Gobject 2.32]
<code>g_value_set_ulong</code> [Gobject 2.32]	<code>g_value_take_boxed</code> [Gobject 2.32]
<code>g_value_take_object</code> [Gobject 2.32]	<code>g_value_take_param</code> [Gobject 2.32]
<code>g_value_take_string</code> [Gobject 2.32]	

An LSB conforming implementation shall provide the generic deprecated functions for Gobject Parameters and Values specified in Table 17-86, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-86 libgobject-2.0 - Gobject Parameters and Values Deprecated Function Interfaces

<code>g_value_get_char</code> [Gobject 2.32]	<code>g_value_set_char</code> [Gobject 2.32]

An LSB conforming implementation shall provide the generic data interfaces for Gobject Parameters and Values specified in Table 17-87, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-87 libgobject-2.0 - Gobject Parameters and Values Data Interfaces

<code>g_param_spec_types</code> [Gobject 2.32]	

17.7.7 Gobject GParamSpec

17.7.7.1 Interfaces for Gobject GParamSpec

An LSB conforming implementation shall provide the generic functions for Gobject GParamSpec specified in Table 17-88, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-88 libgobject-2.0 - Gobject GParamSpec Function Interfaces

<code>g_param_spec_get_blurb</code> [Gobject 2.32]	<code>g_param_spec_get_name</code> [Gobject 2.32]
<code>g_param_spec_get_nick</code> [Gobject 2.32]	<code>g_param_spec_get_qdata</code> [Gobject 2.32]

g_param_spec_get_redirect_target [Gobject 2.32]	g_param_spec_gtype [Gobject 2.32]
g_param_spec_internal [Gobject 2.32]	g_param_spec_pool_insert [Gobject 2.32]
g_param_spec_pool_list [Gobject 2.32]	g_param_spec_pool_list_owned [Gobject 2.32]
g_param_spec_pool_lookup [Gobject 2.32]	g_param_spec_pool_new [Gobject 2.32]
g_param_spec_pool_remove [Gobject 2.32]	g_param_spec_ref [Gobject 2.32]
g_param_spec_ref_sink [Gobject 2.32]	g_param_spec_set_qdata [Gobject 2.32]
g_param_spec_set_qdata_full [Gobject 2.32]	g_param_spec_sink [Gobject 2.32]
g_param_spec_steal_qdata [Gobject 2.32]	g_param_spec_unref [Gobject 2.32]
g_param_type_register_static [Gobject 2.32]	g_param_value_convert [Gobject 2.32]
g_param_value_defaults [Gobject 2.32]	g_param_value_set_default [Gobject 2.32]
g_param_value_validate [Gobject 2.32]	g_param_values_cmp [Gobject 2.32]

17.7.8 Gobject GBoxed

17.7.8.1 Interfaces for Gobject GBoxed

An LSB conforming implementation shall provide the generic functions for Gobject GBoxed specified in Table 17-89, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-89 libgobject-2.0 - Gobject GBoxed Function Interfaces

g_array_get_type [Gobject 2.32]	g_boxed_copy [Gobject 2.32]
g_boxed_free [Gobject 2.32]	g_boxed_type_register_static [Gobject 2.32]
g_byte_array_get_type [Gobject 2.32]	g_bytes_get_type [Gobject 2.32]
g_date_get_type [Gobject 2.32]	g_date_time_get_type [Gobject 2.32]
g_error_get_type [Gobject 2.32]	g_gstring_get_type [Gobject 2.32]
g_key_file_get_type [Gobject 2.32]	g_main_context_get_type [Gobject 2.32]
g_main_loop_get_type [Gobject 2.32]	g_match_info_get_type [Gobject 2.32]
g_pointer_type_register_static [Gobject 2.32]	g_ptr_array_get_type [Gobject 2.32]

<code>g_regex_get_type</code> [Gobject 2.32]	<code>g_source_get_type</code> [Gobject 2.32]
<code>g_strv_get_type</code> [Gobject 2.32]	<code>g_variant_builder_get_type</code> [Gobject 2.32]
<code>g_variant_get_gtype</code> [Gobject 2.32]	<code>g_variant_type_get_gtype</code> [Gobject 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for Gobject GBoxed specified in Table 17-90, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-90 libgobject-2.0 - Gobject GBoxed Deprecated Function Interfaces

<code>g_variant_get_gtype</code> [Gobject 2.32]	

17.7.9 Gobject Closures

17.7.9.1 Interfaces for Gobject Closures

An LSB conforming implementation shall provide the generic functions for Gobject Closures specified in Table 17-91, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-91 libgobject-2.0 - Gobject Closures Function Interfaces

<code>g_cclosure_marshal_BOOLEAN__FLAG</code> [Gobject 2.32]	<code>g_cclosure_marshal_STRING__OBJECT_POINTER</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__BOOLEAN</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__BOOLEANv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__BOXED</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__BOXEDv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__CHAR</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__CHARv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__DOUBLE</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__DOUBLEv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__ENUM</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__ENUMv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__FLAGS</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__FLAGsv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__FLOAT</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__FLOATv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__INT</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__INTv</code> [Gobject 2.32]
<code>g_cclosure_marshal_VOID__LONG</code> [Gobject 2.32]	<code>g_cclosure_marshal_VOID__LONGv</code> [Gobject 2.32]

g_cclosure_marshal_VOID__OBJECT [Gobject 2.32]	g_cclosure_marshal_VOID__OBJECT v [Gobject 2.32]
g_cclosure_marshal_VOID__PARAM [Gobject 2.32]	g_cclosure_marshal_VOID__PARAM v [Gobject 2.32]
g_cclosure_marshal_VOID__POINTE R [Gobject 2.32]	g_cclosure_marshal_VOID__POINTE Rv [Gobject 2.32]
g_cclosure_marshal_VOID__STRING [Gobject 2.32]	g_cclosure_marshal_VOID__STRING v [Gobject 2.32]
g_cclosure_marshal_VOID__UCHAR [Gobject 2.32]	g_cclosure_marshal_VOID__UCHAR v [Gobject 2.32]
g_cclosure_marshal_VOID__UINT [Gobject 2.32]	g_cclosure_marshal_VOID__UINT_P OINTER [Gobject 2.32]
g_cclosure_marshal_VOID__UINT_P OINTERv [Gobject 2.32]	g_cclosure_marshal_VOID__UINTv [Gobject 2.32]
g_cclosure_marshal_VOID__ULONG [Gobject 2.32]	g_cclosure_marshal_VOID__ULONG v [Gobject 2.32]
g_cclosure_marshal_VOID__VARIA NT [Gobject 2.32]	g_cclosure_marshal_VOID__VARIA NTv [Gobject 2.32]
g_cclosure_marshal_VOID__VOID [Gobject 2.32]	g_cclosure_marshal_VOID__VOIDv [Gobject 2.32]
g_cclosure_new [Gobject 2.32]	g_cclosure_new_object [Gobject 2.32]
g_cclosure_new_object_swap [Gobject 2.32]	g_cclosure_new_swap [Gobject 2.32]
g_closure_add_finalize_notifier [Gobject 2.32]	g_closure_add_invalidate_notifier [Gobject 2.32]
g_closure_add_marshal_guards [Gobject 2.32]	g_closure_invalidate [Gobject 2.32]
g_closure_invoke [Gobject 2.32]	g_closure_new_object [Gobject 2.32]
g_closure_new_simple [Gobject 2.32]	g_closure_ref [Gobject 2.32]
g_closure_remove_finalize_notifier [Gobject 2.32]	g_closure_remove_invalidate_notifie r [Gobject 2.32]
g_closure_set_marshal [Gobject 2.32]	g_closure_set_meta_marshal [Gobject 2.32]
g_closure_sink [Gobject 2.32]	g_closure_unref [Gobject 2.32]
g_source_set_closure [Gobject 2.32]	

17.7.10 Gobject Gobject

17.7.10.1 Interfaces for Gobject Gobject

An LSB conforming implementation shall provide the generic functions for Gobject Gobject specified in Table 17-92, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-92 libgobject-2.0 - GObject GObject Function Interfaces

g_object_add_toggle_ref [Gobject 2.32]	g_object_add_weak_pointer [Gobject 2.32]
g_object_class_find_property [Gobject 2.32]	g_object_class_install_property [Gobject 2.32]
g_object_class_list_properties [Gobject 2.32]	g_object_class_override_property [Gobject 2.32]
g_object_connect [Gobject 2.32]	g_object_disconnect [Gobject 2.32]
g_object_force_floating [Gobject 2.32]	g_object_freeze_notify [Gobject 2.32]
g_object_get [Gobject 2.32]	g_object_get_data [Gobject 2.32]
g_object_get_property [Gobject 2.32]	g_object_get_qdata [Gobject 2.32]
g_object_get_valist [Gobject 2.32]	g_object_interface_find_property [Gobject 2.32]
g_object_interface_install_property [Gobject 2.32]	g_object_interface_list_properties [Gobject 2.32]
g_object_is_floating [Gobject 2.32]	g_object_new [Gobject 2.32]
g_object_new_valist [Gobject 2.32]	g_object_newv [Gobject 2.32]
g_object_notify [Gobject 2.32]	g_object_ref [Gobject 2.32]
g_object_ref_sink [Gobject 2.32]	g_object_remove_toggle_ref [Gobject 2.32]
g_object_remove_weak_pointer [Gobject 2.32]	g_object_run_dispose [Gobject 2.32]
g_object_set [Gobject 2.32]	g_object_set_data [Gobject 2.32]
g_object_set_data_full [Gobject 2.32]	g_object_set_property [Gobject 2.32]
g_object_set_qdata [Gobject 2.32]	g_object_set_qdata_full [Gobject 2.32]
g_object_set_valist [Gobject 2.32]	g_object_steal_data [Gobject 2.32]
g_object_steal_qdata [Gobject 2.32]	g_object_thaw_notify [Gobject 2.32]
g_object_unref [Gobject 2.32]	g_object_watch_closure [Gobject 2.32]
g_object_weak_ref [Gobject 2.32]	g_object_weak_unref [Gobject 2.32]

17.7.11 GObject GType

17.7.11.1 Interfaces for GObject GType

An LSB conforming implementation shall provide the generic functions for GObject GType specified in Table 17-93, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-93 libgobject-2.0 - GObject GType Function Interfaces

g_closure_get_type [Gobject 2.32]	g_gtype_get_type [Gobject 2.32]

g_hash_table_get_type [Gobject 2.32]	g_initially_unowned_get_type [Gobject 2.32]
g_io_channel_get_type [Gobject 2.32]	g_io_condition_get_type [Gobject 2.32]
g_type_add_class_cache_func [Gobject 2.32]	g_type_add_interface_check [Gobject 2.32]
g_type_add_interface_dynamic [Gobject 2.32]	g_type_add_interface_static [Gobject 2.32]
g_type_check_class_cast [Gobject 2.32]	g_type_check_class_is_a [Gobject 2.32]
g_type_check_instance [Gobject 2.32]	g_type_check_instance_cast [Gobject 2.32]
g_type_check_instance_is_a [Gobject 2.32]	g_type_check_is_value_type [Gobject 2.32]
g_type_check_value [Gobject 2.32]	g_type_check_value_holds [Gobject 2.32]
g_type_children [Gobject 2.32]	g_type_class_add_private [Gobject 2.32]
g_type_class_peek [Gobject 2.32]	g_type_class_peek_parent [Gobject 2.32]
g_type_class_peek_static [Gobject 2.32]	g_type_class_ref [Gobject 2.32]
g_type_class_unref [Gobject 2.32]	g_type_class_unref_uncached [Gobject 2.32]
g_type_create_instance [Gobject 2.32]	g_type_default_interface_peek [Gobject 2.32]
g_type_default_interface_ref [Gobject 2.32]	g_type_default_interface_unref [Gobject 2.32]
g_type_depth [Gobject 2.32]	g_type_free_instance [Gobject 2.32]
g_type_from_name [Gobject 2.32]	g_type_fundamental [Gobject 2.32]
g_type_fundamental_next [Gobject 2.32]	g_type_get_plugin [Gobject 2.32]
g_type_get_qdata [Gobject 2.32]	g_type_init [Gobject 2.32]
g_type_init_with_debug_flags [Gobject 2.32]	g_type_instance_get_private [Gobject 2.32]
g_type_interface_add_prerequisite [Gobject 2.32]	g_type_interface_get_plugin [Gobject 2.32]
g_type_interface_peek [Gobject 2.32]	g_type_interface_peek_parent [Gobject 2.32]
g_type_interface_prerequisites [Gobject 2.32]	g_type_interfaces [Gobject 2.32]

g_type_is_a [Gobject 2.32]	g_type_name [Gobject 2.32]
g_type_name_from_class [LSB]	g_type_name_from_instance [LSB]
g_type_next_base [Gobject 2.32]	g_type_parent [Gobject 2.32]
g_type_qname [Gobject 2.32]	g_type_query [Gobject 2.32]
g_type_register_dynamic [Gobject 2.32]	g_type_register_fundamental [Gobject 2.32]
g_type_register_static [Gobject 2.32]	g_type_register_static_simple [Gobject 2.32]
g_type_remove_class_cache_func [Gobject 2.32]	g_type_remove_interface_check [Gobject 2.32]
g_type_set_qdata [Gobject 2.32]	g_type_test_flags [Gobject 2.32]
g_type_value_table_peek [Gobject 2.32]	

17.7.12 Gobject GTypePlugin

17.7.12.1 Interfaces for Gobject GTypePlugin

An LSB conforming implementation shall provide the generic functions for Gobject GTypePlugin specified in Table 17-94, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-94 libgobject-2.0 - Gobject GTypePlugin Function Interfaces

g_type_plugin_complete_interface_info [Gobject 2.32]	g_type_plugin_complete_type_info [Gobject 2.32]
g_type_plugin_get_type [Gobject 2.32]	g_type_plugin_unuse [Gobject 2.32]
g_type_plugin_use [Gobject 2.32]	

17.7.13 Gobject GBinding

17.7.13.1 Interfaces for Gobject GBinding

An LSB conforming implementation shall provide the generic functions for Gobject GBinding specified in Table 17-95, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-95 libgobject-2.0 - Gobject GBinding Function Interfaces

g_binding_get_flags [Gobject 2.32]	g_binding_get_source [Gobject 2.32]
g_binding_get_source_property [Gobject 2.32]	g_binding_get_target [Gobject 2.32]
g_binding_get_target_property [Gobject 2.32]	g_object_bind_property [Gobject 2.32]
g_object_bind_property_full [Gobject 2.32]	g_object_bind_property_with_closures [Gobject 2.32]

17.8 Data Definitions for libobject-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libobject-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.8.1 glib-2.0/glib-object.h

```
#define G_CLOSURE_NEEDS_MARSHAL(closure)
    (((GClosure*) (closure))->marshal == NULL)
#define G_TYPE_FROM_INTERFACE(g_iface) \
    (((GTypeInterface*) (g_iface))->g_type)
#define G_TYPE_CCC(cp,gt,ct) \
    ((ct*) g_type_check_class_cast ((GTypeClass*) cp, gt))
#define G_TYPE_CIC(ip,gt,ct) \
    ((ct*) g_type_check_instance_cast ((GTypeInstance*) ip, gt))
#define G_TYPE_IGI(ip,gt,ct) \
    ((ct*) g_type_interface_peek ((GTypeInstance*) ip)-
    >g_class, gt))
#define G_TYPE_INSTANCE_GET_PRIVATE(instance,g_type,c_type) \
    ((c_type*) g_type_instance_get_private ((GTypeInstance*)
    (instance), \
    (g_type)))
#define G_TYPE_MAKE_FUNDAMENTAL(x) \
    ((GType) ((x) << G_TYPE_FUNDAMENTAL_SHIFT))
#define G_ENUM_CLASS(class) \
    (G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_ENUM, GEnumClass))
#define G_FLAGS_CLASS(class) \
    (G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_FLAGS,
    GFlagsClass))
#define G_OBJECT_CLASS(class) \
    (G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_OBJECT,
    GObjectClass))
#define G_TYPE_MODULE_CLASS(class) \
    (G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_TYPE_MODULE, \
    GTypeModuleClass))
#define G_PARAM_SPEC_CLASS(pclass) \
    (G_TYPE_CHECK_CLASS_CAST ((pclass), G_TYPE_PARAM,
    GParamSpecClass))
#define G_TYPE_PLUGIN_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), G_TYPE_TYPE_PLUGIN, \
    GTypePluginClass))
#define G_IS_ENUM_CLASS(class) \
    (G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_ENUM))
#define G_IS_FLAGS_CLASS(class) \
    (G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_FLAGS))
#define G_IS_OBJECT_CLASS(class) \
    (G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_OBJECT))
#define G_IS_TYPE_MODULE_CLASS(class) \
    (G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_TYPE_MODULE))
```

```

#define G_IS_PARAM_SPEC_CLASS(pclass) \
    (G_TYPE_CHECK_CLASS_TYPE ((pclass), G_TYPE_PARAM))
#define G_IS_TYPE_PLUGIN_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_TYPE ((vtable), G_TYPE_TYPE_PLUGIN))
#define G_TYPE_PLUGIN(inst) \
    (G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_TYPE_PLUGIN, \
    GTypePlugin))
#define G_TYPE_MODULE(module) \
    (G_TYPE_CHECK_INSTANCE_CAST ((module), G_TYPE_TYPE_MODULE, \
    \
    GTypeModule))
#define G_OBJECT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), G_TYPE_OBJECT, \
    GObject))
#define G_PARAM_SPEC(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM, \
    GParamSpec))
#define G_PARAM_SPEC_BOOLEAN(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_BOOLEAN, \
    \
    GParamSpecBoolean))
#define G_PARAM_SPEC_BOXED(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_BOXED, \
    GParamSpecBoxed))
#define G_PARAM_SPEC_CHAR(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_CHAR, \
    GParamSpecChar))
#define G_PARAM_SPEC_DOUBLE(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_DOUBLE, \
    \
    GParamSpecDouble))
#define G_PARAM_SPEC_ENUM(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_ENUM, \
    GParamSpecEnum))
#define G_PARAM_SPEC_FLAGS(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_FLAGS, \
    GParamSpecFlags))
#define G_PARAM_SPEC_FLOAT(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_FLOAT, \
    GParamSpecFloat))
#define G_PARAM_SPEC_INT(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_INT, \
    GParamSpecInt))
#define G_PARAM_SPEC_INT64(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_INT64, \
    GParamSpecInt64))
#define G_PARAM_SPEC_LONG(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_LONG, \
    GParamSpecLong))
#define G_PARAM_SPEC_OBJECT(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_OBJECT, \
    \
    GParamSpecObject))
#define G_PARAM_SPEC_OVERRIDE(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_OVERRIDE, \
    \
    GParamSpecOverride))
#define G_PARAM_SPEC_PARAM(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_PARAM, \
    GParamSpecParam))
#define G_PARAM_SPEC_POINTER(pspec) \
    (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_POINTER, \
    \
    GParamSpecPointer))
#define G_PARAM_SPEC_STRING(pspec) \
    \

```



```

        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_STRING,
\
        GParamSpecString))
#define G_PARAM_SPEC_UCHAR(pspec) \
        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_UCHAR, \
        GParamSpecUChar))
#define G_PARAM_SPEC_UINT(pspec) \
        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_UINT, \
        GParamSpecUInt))
#define G_PARAM_SPEC_UINT64(pspec) \
        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_UINT64,
\
        GParamSpecUInt64))
#define G_PARAM_SPEC_ULONG(pspec) \
        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_ULONG, \
        GParamSpecULong))
#define G_PARAM_SPEC_UNICHAR(pspec) \
        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_UNICHAR,
\
        GParamSpecUnichar))
#define G_PARAM_SPEC_VALUE_ARRAY(pspec) \
        (G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_VALUE_ARRAY, \
        GParamSpecValueArray)) /* : DEPRECATED */
#define G_IS_TYPE_PLUGIN(inst) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_TYPE_PLUGIN))
#define G_IS_TYPE_MODULE(module) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((module), G_TYPE_TYPE_MODULE))
#define G_IS_OBJECT(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), G_TYPE_OBJECT))
#define G_IS_PARAM_SPEC(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM))
#define G_IS_PARAM_SPEC_BOOLEAN(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_BOOLEAN))
#define G_IS_PARAM_SPEC_BOXED(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_BOXED))
#define G_IS_PARAM_SPEC_CHAR(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_CHAR))
#define G_IS_PARAM_SPEC_DOUBLE(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_DOUBLE))
#define G_IS_PARAM_SPEC_ENUM(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_ENUM))
#define G_IS_PARAM_SPEC_FLAGS(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_FLAGS))
#define G_IS_PARAM_SPEC_FLOAT(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_FLOAT))
#define G_IS_PARAM_SPEC_INT(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_INT))
#define G_IS_PARAM_SPEC_INT64(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_INT64))
#define G_IS_PARAM_SPEC_LONG(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_LONG))
#define G_IS_PARAM_SPEC_OBJECT(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_OBJECT))
#define G_IS_PARAM_SPEC_OVERRIDE(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_OVERRIDE))
#define G_IS_PARAM_SPEC_PARAM(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_PARAM))
#define G_IS_PARAM_SPEC_POINTER(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_POINTER))
#define G_IS_PARAM_SPEC_STRING(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_STRING))
#define G_IS_PARAM_SPEC_UCHAR(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_UCHAR))
#define G_IS_PARAM_SPEC_UINT(pspec) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_UINT))

```



```

(G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_UINT))
#define G_IS_PARAM_SPEC_UINT64(pspec) \
(G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_UINT64))
#define G_IS_PARAM_SPEC_ULONG(pspec) \
(G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_ULONG))
#define G_IS_PARAM_SPEC_UNICHAR(pspec) \
(G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_UNICHAR))
#define G_IS_PARAM_SPEC_VALUE_ARRAY(pspec) \
(G_TYPE_CHECK_INSTANCE_TYPE ((pspec),
G_TYPE_PARAM_VALUE_ARRAY)) /* : DEPRECATED */
#define G_VALUE HOLDS(value,type) \
(G_TYPE_CHECK_VALUE_TYPE ((value), (type)))
#define G_VALUE HOLDS_BOOLEAN(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_BOOLEAN))
#define G_VALUE HOLDS_BOXED(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_BOXED))
#define G_VALUE HOLDS_CHAR(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_CHAR))
#define G_VALUE HOLDS_DOUBLE(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_DOUBLE))
#define G_VALUE HOLDS_ENUM(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_ENUM))
#define G_VALUE HOLDS_FLAGS(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_FLAGS))
#define G_VALUE HOLDS_FLOAT(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_FLOAT))
#define G_VALUE HOLDS_INT(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_INT))
#define G_VALUE HOLDS_INT64(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_INT64))
#define G_VALUE HOLDS_LONG(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_LONG))
#define G_VALUE HOLDS_OBJECT(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_OBJECT))
#define G_VALUE HOLDS_PARAM(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_PARAM))
#define G_VALUE HOLDS_POINTER(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_POINTER))
#define G_VALUE HOLDS_STRING(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_STRING))
#define G_VALUE HOLDS_UCHAR(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_UCHAR))
#define G_VALUE HOLDS_UINT(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_UINT))
#define G_VALUE HOLDS_UINT64(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_UINT64))
#define G_VALUE HOLDS_ULONG(value) \
(G_TYPE_CHECK_VALUE_TYPE ((value), G_TYPE_ULONG))
#define G_TYPE_FROM_INSTANCE(instance) \
(G_TYPE_FROM_CLASS (((GTypeInstance*) (instance))->g_class))
#define G_TYPE_IS_INTERFACE(type) \
(G_TYPE_FUNDAMENTAL (type) == G_TYPE_INTERFACE)
#define G_TYPE_IS_OBJECT(type) \
(G_TYPE_FUNDAMENTAL (type) == G_TYPE_OBJECT)
#define G_TYPE_MODULE_GET_CLASS(module) \
(G_TYPE_INSTANCE_GET_CLASS ((module), G_TYPE_TYPE_MODULE, \
GTypeModuleClass))
#define G_OBJECT_GET_CLASS(object) \
(G_TYPE_INSTANCE_GET_CLASS ((object), G_TYPE_OBJECT,
GObjectClass))
#define G_PARAM_SPEC_GET_CLASS(pspec) \
(G_TYPE_INSTANCE_GET_CLASS ((pspec), G_TYPE_PARAM,
GParamSpecClass))
#define G_TYPE_PLUGIN_GET_CLASS(inst) \
(G_TYPE_INSTANCE_GET_INTERFACE ((inst), G_TYPE_TYPE_PLUGIN,
\

```

```

        GTypePluginClass))
#define G_ENUM_CLASS_TYPE_NAME(class) \
    (g_type_name (G_ENUM_CLASS_TYPE (class)))
#define G_FLAGS_CLASS_TYPE_NAME(class) \
    (g_type_name (G_FLAGS_TYPE (class)))
#define G_OBJECT_CLASS_NAME(class) \
    (g_type_name (G_OBJECT_CLASS_TYPE (class)))
#define G_PARAM_SPEC_TYPE_NAME(pspec) \
    (g_type_name (G_PARAM_SPEC_TYPE (pspec)))
#define G_TYPE_IS_ABSTRACT(type) \
    (g_type_test_flags ((type), G_TYPE_FLAG_ABSTRACT))
#define G_TYPE_IS_CLASSED(type) \
    (g_type_test_flags ((type), G_TYPE_FLAG_CLASSED))
#define G_TYPE_IS_DEEP_DERIVABLE(type) \
    (g_type_test_flags ((type), G_TYPE_FLAG_DEEP_DERIVABLE))
#define G_TYPE_IS_DERIVABLE(type) \
    (g_type_test_flags ((type), G_TYPE_FLAG_DERIVABLE))
#define G_TYPE_IS_INSTANTIATABLE(type) \
    (g_type_test_flags ((type), G_TYPE_FLAG_INSTANTIATABLE))
#define G_TYPE_IS_VALUE_ABSTRACT(type) \
    (g_type_test_flags ((type), G_TYPE_FLAG_VALUE_ABSTRACT))
#define G_TYPE_HAS_VALUE_TABLE(type) \
    (g_type_value_table_peek (type) != NULL)
#define G_TYPE_CHECK_CLASS_CAST(g_class,g_type,c_type) \
    (_G_TYPE_CCC ((g_class), (g_type), c_type))
#define G_TYPE_CHECK_CLASS_TYPE(g_class,g_type) \
    (_G_TYPE_CCT ((g_class), (g_type)))
#define G_TYPE_CHECK_INSTANCE(instance) \
    (_G_TYPE_CHI ((GTypeInstance*) (instance)))
#define G_TYPE_CHECK_INSTANCE_CAST(instance,g_type,c_type) \
    (_G_TYPE_CIC ((instance), (g_type), c_type))
#define G_TYPE_CHECK_INSTANCE_TYPE(instance,g_type) \
    (_G_TYPE_CIT ((instance), (g_type)))
#define G_TYPE_CHECK_VALUE_TYPE(value,g_type) \
    (_G_TYPE_CVH ((value), (g_type)))
#define G_TYPE_INSTANCE_GET_CLASS(instance,g_type,c_type) \
    (_G_TYPE_IGC ((instance), (g_type), c_type))
#define G_TYPE_INSTANCE_GET_INTERFACE(instance,g_type,c_type) \
    (_G_TYPE_IGI ((instance), (g_type), c_type))
#define G_DEFINE_TYPE_WITH_CODE(TN,t_n,T_P,_C_) \
    G_DEFINE_TYPE_EXTENDED (TN, t_n, T_P, 0, _C_)
#define G_DEFINE_TYPE(TN,t_n,T_P) \
    G_DEFINE_TYPE_EXTENDED (TN, t_n, T_P, 0, {})
#define G_DEFINE_ABSTRACT_TYPE(TN,t_n,T_P) \
    G_DEFINE_TYPE_EXTENDED (TN, t_n, T_P, G_TYPE_FLAG_ABSTRACT,
    {})
#define G_OBJECT_WARN_INVALID_PROPERTY_ID(object,property_id,pspec) \
    \
    G_OBJECT_WARN_INVALID_PSPEC ((object), "property",
    (property_id), \
    (pspec))
#define g_signal_connect(instance,detailed_signal,c_handler,data) \
    \
    g_signal_connect_data ((instance), (detailed_signal),
    (c_handler), \
    (data), NULL, (GConnectFlags) 0)
#define
g_signal_connect_after(instance,detailed_signal,c_handler,data) \
    g_signal_connect_data ((instance), (detailed_signal),
    (c_handler), \
    (data), NULL, G_CONNECT_AFTER)
#define
g_signal_connect_swapped(instance,detailed_signal,c_handler,data) \
    \
    g_signal_connect_data ((instance), (detailed_signal),
    (c_handler), \

```

```

        (data), NULL, G_CONNECT_SWAPPED)
#define g_signal_handlers_block_by_func(instance,func,data) \
    g_signal_handlers_block_matched          ((instance),
(GSignalMatchType) \
    (G_SIGNAL_MATCH_FUNC | G_SIGNAL_MATCH_DATA), 0, 0, NULL,
(func), \
    (data))
#define g_signal_handlers_disconnect_by_func(instance,func,data)
\
    g_signal_handlers_disconnect_matched          ((instance),
(GSignalMatchType) \
    (G_SIGNAL_MATCH_FUNC | G_SIGNAL_MATCH_DATA), 0, 0, NULL,
(func), \
    (data))
#define g_signal_handlers_unblock_by_func(instance,func,data) \
    g_signal_handlers_unblock_matched          ((instance),
(GSignalMatchType) \
    (G_SIGNAL_MATCH_FUNC | G_SIGNAL_MATCH_DATA), 0, 0, NULL,
(func), \
    (data))
#define G_CCLOSURE_SWAP_DATA(cclosure) \
    (((GClosure*) (cclosure))->derivative flag)
#define G_OBJECT_WARN_INVALID_PSPEC(object,pname,property_id,pspec)
\
G_STMT_START { \
    GObject * object = (GObject*) (object); \
    GParamSpec * _pspec = (GParamSpec*) (pspec); \
    guint _property_id = (property_id); \
    g_warning ("%s: invalid %s id %u for \"%s\" of type %s in %s", \
        G_STRLOC, \
        (pname), \
        _property_id, \
        _pspec->name, \
        g_type_name (G_PARAM_SPEC_TYPE (_pspec)), \
        G_OBJECT_TYPE_NAME (_object)); \
} G_STMT_END
#define G_CLOSURE_N_NOTIFIERS(cl)          (((cl)->n_guards << 1L) + \
    (cl)->n_fnotifiers + (cl)->n_inotifiers)
#define G_TYPE_FROM_CLASS(g_class)          (((GTypeClass*) (g_class))-
>g_type)
#define G_VALUE_TYPE(value)          (((GValue*) (value))->g_type)
#define G_TYPE_IGC(ip,gt,ct)          ((ct*) (((GTypeInstance*) ip)-
>g_class))
#define G_TYPE_CLASS_GET_PRIVATE(klass,g_type,c_type) ((c_type*)
g_type_class_get_private ((GTypeClass*) (klass), (g_type)))
#define G_CALLBACK(f)          ((GCallback) (f))
#define G_REAL_CLOSURE(_c)          ((GRealClosure *)G_STRUCT_MEMBER_P
(( _c ), -G_STRUCT_OFFSET (GRealClosure, closure)))
#define G_TYPE_FLAG_RESERVED_ID_BIT          ((GType) (1 << 0))
#define G_TYPE_IS_FUNDAMENTAL(type)          ((type) <=
G_TYPE_FUNDAMENTAL_MAX)
#define G_TYPE_IS_DERIVED(type) ((type) > G_TYPE_FUNDAMENTAL_MAX)
#define G_ATOMIC_ARRAY_GET_LOCKED(_array,_type)          ((_type
*) ((_array)->data))
#define G_ATOMIC_ARRAY_DATA_SIZE(mem)          (*((gsize *) (mem) - 1))
#define G_PARAM_MASK          (0x000000ff)
#define G_VALUE_NOCOPY_CONTENTS (1 << 27)
#define G_TYPE_FUNDAMENTAL_SHIFT          (2)
#define G_TYPE_RESERVED_GLIB_FIRST          (22)
#define G_TYPE_FUNDAMENTAL_MAX (255 << G_TYPE_FUNDAMENTAL_SHIFT)
#define G_TYPE_RESERVED_GLIB_LAST          (31)
#define G_TYPE_RESERVED_BSE_FIRST          (32)
#define G_TYPE_RESERVED_BSE_LAST          (48)
#define G_TYPE_RESERVED_USER_FIRST          (49)
#define G_PARAM_USER_SHIFT          (8)
#define G_TYPE_ARRAY          (g_array_get_type ())

```

```

#define G_TYPE_BINDING_FLAGS (g_binding_flags_get_type ())
#define G_TYPE_BINDING (g_binding_get_type ())
#define G_TYPE_BYTES (g_bytes_get_type ())
#define G_TYPE_BYTE_ARRAY (g_byte_array_get_type ())
#define G_TYPE_CLOSURE (g_closure_get_type ())
#define G_TYPE_DATE (g_date_get_type ())
#define G_TYPE_DATE_TIME (g_date_time_get_type ())
#define G_TYPE_ERROR (g_error_get_type ())
#define G_TYPE_GSTRING (g_gstring_get_type ())
#define G_TYPE_GTYPE (g_gtype_get_type ())
#define G_TYPE_HASH_TABLE (g_hash_table_get_type ())
#define G_TYPE_INITIALLY_UNOWNED (g_initially_unowned_get_type ())
#define G_TYPE_IO_CHANNEL (g_io_channel_get_type ())
#define G_TYPE_IO_CONDITION (g_io_condition_get_type ())
#define G_TYPE_KEY_FILE (g_key_file_get_type ())
#define G_TYPE_MAIN_CONTEXT (g_main_context_get_type ())
#define G_TYPE_MAIN_LOOP (g_main_loop_get_type ())
#define G_TYPE_MATCH_INFO (g_match_info_get_type ())
#define G_PARAM_READWRITE (G_PARAM_READABLE | G_PARAM_WRITABLE)
#define G_PARAM_SPEC_VALUE_TYPE(pspec) (G_PARAM_SPEC (pspec) ->value_type)
#define G_TYPE_PARAM_CHAR (g_param_spec_types[10])
#define G_TYPE_PARAM_ENUM (g_param_spec_types[10])
#define G_TYPE_PARAM_FLAGS (g_param_spec_types[11])
#define G_TYPE_PARAM_FLOAT (g_param_spec_types[12])
#define G_TYPE_PARAM_DOUBLE (g_param_spec_types[13])
#define G_TYPE_PARAM_STRING (g_param_spec_types[14])
#define G_TYPE_PARAM_PARAM (g_param_spec_types[15])
#define G_TYPE_PARAM_BOXED (g_param_spec_types[16])
#define G_TYPE_PARAM_POINTER (g_param_spec_types[17])
#define G_TYPE_PARAM_VALUE_ARRAY (g_param_spec_types[18])
/* : DEPRECATED */
#define G_TYPE_PARAM_OBJECT (g_param_spec_types[19])
#define G_TYPE_PARAM_UCHAR (g_param_spec_types[1])
#define G_TYPE_PARAM_OVERRIDE (g_param_spec_types[20])
#define G_TYPE_PARAM_GTYPE (g_param_spec_types[21])
#define G_TYPE_PARAM_VARIANT (g_param_spec_types[22])
#define G_TYPE_PARAM_BOOLEAN (g_param_spec_types[2])
#define G_TYPE_PARAM_INT (g_param_spec_types[3])
#define G_TYPE_PARAM_UINT (g_param_spec_types[4])
#define G_TYPE_PARAM_LONG (g_param_spec_types[5])
#define G_TYPE_PARAM_ULONG (g_param_spec_types[6])
#define G_TYPE_PARAM_INT64 (g_param_spec_types[7])
#define G_TYPE_PARAM_UINT64 (g_param_spec_types[8])
#define G_TYPE_PARAM_UNICHAR (g_param_spec_types[9])
#define G_PARAM_STATIC_STRINGS (G_PARAM_STATIC_NAME | G_PARAM_STATIC_NICK | G_PARAM_STATIC_BLURB)
#define G_TYPE_PTR_ARRAY (g_ptr_array_get_type ())
#define G_TYPE_REGEX (g_regex_get_type ())
#define G_TYPE_SOURCE (g_source_get_type ())
#define G_TYPE_STRV (g_strv_get_type ())
#define G_INITIALLY_UNOWNED_CLASS(class) (G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_INITIALLY_UNOWNED, GInitiallyUnownedClass))
#define _G_TYPE_CCT(cp,gt) (g_type_check_class_is_a ((GTypeClass*) cp, gt))
#define G_IS_INITIALLY_UNOWNED_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_INITIALLY_UNOWNED))
#define _G_TYPE_CHI(ip) (g_type_check_instance ((GTypeInstance*) ip))
#define G_BINDING(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj), G_TYPE_BINDING, GBinding))

```

```

#define G_INITIALLY_UNOWNED(object)
(G_TYPE_CHECK_INSTANCE_CAST ((object), G_TYPE_INITIALLY_UNOWNED,
GInitiallyUnowned))
#define G_PARAM_SPEC_GTYPE(pspec) (G_TYPE_CHECK_INSTANCE_CAST
((pspec), G_TYPE_PARAM_GTYPE, GParamSpecGType))
#define G_PARAM_SPEC_VARIANT(pspec)
(G_TYPE_CHECK_INSTANCE_CAST ((pspec), G_TYPE_PARAM_VARIANT,
GParamSpecVariant))
#define G_TYPE_CIT(ip,gt) (g_type_check_instance_is_a
((GTypeInstance*) ip, gt))
#define G_IS_BINDING(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_BINDING))
#define G_IS_INITIALLY_UNOWNED(object)
(G_TYPE_CHECK_INSTANCE_TYPE ((object), G_TYPE_INITIALLY_UNOWNED))
#define G_IS_PARAM_SPEC_GTYPE(pspec)
(G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_GTYPE))
#define G_IS_PARAM_SPEC_VARIANT(pspec)
(G_TYPE_CHECK_INSTANCE_TYPE ((pspec), G_TYPE_PARAM_VARIANT))
#define G_TYPE_IS_VALUE(type) (g_type_check_is_value_type (type))
#define G_TYPE_IS_VALUE_TYPE(type) (g_type_check_is_value_type
(type))
#define G_TYPE_CHV(vl) (g_type_check_value ((GValue*) vl))
#define G_IS_VALUE(value) (G_TYPE_CHECK_VALUE (value))
#define G_TYPE_CVH(vl,gt) (g_type_check_value_holds ((GValue*)
vl, gt))
#define G_VALUE_HOLDS_GTYPE(value) (G_TYPE_CHECK_VALUE_TYPE
((value), G_TYPE_GTYPE))
#define G_VALUE_HOLDS_VARIANT(value) (G_TYPE_CHECK_VALUE_TYPE
((value), G_TYPE_VARIANT))
#define G_SIGNAL_TYPE_STATIC_SCOPE
(G_TYPE_FLAG_RESERVED_ID_BIT)
#define G_ENUM_CLASS_TYPE(class) (G_TYPE_FROM_CLASS (class))
#define G_FLAGS_CLASS_TYPE(class) (G_TYPE_FROM_CLASS (class))
#define G_OBJECT_CLASS_TYPE(class) (G_TYPE_FROM_CLASS (class))
#define G_OBJECT_TYPE(object) (G_TYPE_FROM_INSTANCE (object))
#define G_PARAM_SPEC_TYPE(pspec) (G_TYPE_FROM_INSTANCE
(pspec))
#define G_TYPE_IS_BOXED(type) (G_TYPE_FUNDAMENTAL (type) ==
G_TYPE_BOXED)
#define G_TYPE_IS_ENUM(type) (G_TYPE_FUNDAMENTAL (type) ==
G_TYPE_ENUM)
#define G_TYPE_IS_FLAGS(type) (G_TYPE_FUNDAMENTAL (type) ==
G_TYPE_FLAGS)
#define G_TYPE_IS_PARAM(type) (G_TYPE_FUNDAMENTAL (type) ==
G_TYPE_PARAM)
#define G_TYPE_FUNDAMENTAL(type) (g_type_fundamental (type))
#define G_INITIALLY_UNOWNED_GET_CLASS(object)
(G_TYPE_INSTANCE_GET_CLASS ((object), G_TYPE_INITIALLY_UNOWNED,
GInitiallyUnownedClass))
#define G_TYPE_TYPE_MODULE (g_type_module_get_type ())
#define G_OBJECT_TYPE_NAME(object) (g_type_name (G_OBJECT_TYPE
(object)))
#define G_VALUE_TYPE_NAME(value) (g_type_name (G_VALUE_TYPE
(value)))
#define G_TYPE_TYPE_PLUGIN (g_type_plugin_get_type ())
#define G_TYPE_VALUE_ARRAY (g_value_array_get_type ()) /* :
DEPRECATED */
#define G_TYPE_VALUE (g_value_get_type ())
#define G_TYPE_VARIANT_BUILDER (g_variant_builder_get_type ())
#define G_TYPE_VARIANT_TYPE (g_variant_type_get_gtype ())
#define G_TYPE_CHECK_VALUE(value) (_G_TYPE_CHV ((value)))
#define G_SIGNAL_FLAGS_MASK 0x1fff
#define G_SIGNAL_MATCH_MASK 0x3f
#define G_OBJECT_VAR extern
#define
_G_DEFINE_BOXED_TYPE_BEGIN(TypeName,type_name,copy_func,free_func)

```

```

GType type_name ##_get_type (void) { static volatile gsize
g_define_type_id_volatile = 0; if (g_once_init_enter
(&g_define_type_id_volatile)) { GType g_define_type_id =
g_boxed_type_register_static (g_intern_static_string (#TypeName),
(GBoxedCopyFunc) copy_func, (GBoxedFreeFunc) free_func); {
#define G_DEFINE_POINTER_TYPE_BEGIN(TypeName,type_name)
GType type_name ##_get_type (void) { static volatile gsize
g_define_type_id_volatile = 0; if (g_once_init_enter
(&g_define_type_id_volatile)) { GType g_define_type_id =
g_pointer_type_register_static (g_intern_static_string
(#TypeName)); {
#define g_cclosure_marshal_BOOL__BOXED_BOXED
g_cclosure_marshal_BOOLEAN__BOXED_BOXED
#define g_cclosure_marshal_BOOL__FLAGS
g_cclosure_marshal_BOOLEAN__FLAGS
#define G_DEFINE_BOXED_TYPE(TypeName,type_name,copy_func,free_func)
G_DEFINE_BOXED_TYPE_WITH_CODE (TypeName, type_name, copy_func,
free_func, {})
#define G_DEFINE_DYNAMIC_TYPE(TN,t_n,T_P)
G_DEFINE_DYNAMIC_TYPE_EXTENDED (TN, t_n, T_P, 0, {})
#define G_DEFINE_INTERFACE(TN,t_n,T_P)
G_DEFINE_INTERFACE_WITH_CODE(TN, t_n, T_P, ; )
#define G_DEFINE_POINTER_TYPE(TypeName,type_name)
G_DEFINE_POINTER_TYPE_WITH_CODE (TypeName, type_name, {})
#define g_signal_handlers_disconnect_by_data(instance,data)
g_signal_handlers_disconnect_matched ((instance),
G_SIGNAL_MATCH_DATA, 0, 0, NULL, NULL, (data))
#define g_clear_object(object_ptr) G_STMT_START { gpointer *p
= (gpointer) (object_ptr); gpointer _o; do _o =
g_atomic_pointer_get (p); while G_UNLIKELY
(!g_atomic_pointer_compare_and_exchange (_p, _o, NULL)); if (_o)
g_object_unref (_o); } G_STMT_END
#define G_ATOMIC_ARRAY_DO_TRANSACTION(_array,type,C_)
G_STMT_START { volatile gpointer *_datap = &(_array)->data; type
*transaction_data, *_check; _check = g_atomic_pointer_get
(_datap); do { transaction_data = _check; { C_; } _check =
g_atomic_pointer_get (_datap); } while (transaction_data !=
_check); } G_STMT_END
#define G_TYPE_INVALID G_TYPE_MAKE_FUNDAMENTAL (0)
#define G_TYPE_NONE G_TYPE_MAKE_FUNDAMENTAL (1)
#define G_TYPE_INT64 G_TYPE_MAKE_FUNDAMENTAL (10)
#define G_TYPE_UINT64 G_TYPE_MAKE_FUNDAMENTAL (11)
#define G_TYPE_ENUM G_TYPE_MAKE_FUNDAMENTAL (12)
#define G_TYPE_FLAGS G_TYPE_MAKE_FUNDAMENTAL (13)
#define G_TYPE_FLOAT G_TYPE_MAKE_FUNDAMENTAL (14)
#define G_TYPE_DOUBLE G_TYPE_MAKE_FUNDAMENTAL (15)
#define G_TYPE_STRING G_TYPE_MAKE_FUNDAMENTAL (16)
#define G_TYPE_POINTER G_TYPE_MAKE_FUNDAMENTAL (17)
#define G_TYPE_BOXED G_TYPE_MAKE_FUNDAMENTAL (18)
#define G_TYPE_PARAM G_TYPE_MAKE_FUNDAMENTAL (19)
#define G_TYPE_INTERFACE G_TYPE_MAKE_FUNDAMENTAL (2)
#define G_TYPE_OBJECT G_TYPE_MAKE_FUNDAMENTAL (20)
#define G_TYPE_VARIANT G_TYPE_MAKE_FUNDAMENTAL (21)
#define G_TYPE_CHAR G_TYPE_MAKE_FUNDAMENTAL (3)
#define G_TYPE_UCHAR G_TYPE_MAKE_FUNDAMENTAL (4)
#define G_TYPE_BOOLEAN G_TYPE_MAKE_FUNDAMENTAL (5)
#define G_TYPE_INT G_TYPE_MAKE_FUNDAMENTAL (6)
#define G_TYPE_UINT G_TYPE_MAKE_FUNDAMENTAL (7)
#define G_TYPE_LONG G_TYPE_MAKE_FUNDAMENTAL (8)
#define G_TYPE_ULONG G_TYPE_MAKE_FUNDAMENTAL (9)
#define
G_DEFINE_DYNAMIC_TYPE_EXTENDED (TypeName,type_name,TYPE_PARENT,fla
gs,CODE) \
static void type_name##_init (TypeName *self);
\

```


[illegible]

```

        g_type_interface_add_prerequisite      (g_define_type_id,
TYPE_PREREQ); \
    {
        /* custom code follows */
#define
_G_DEFINE_TYPE_EXTENDED_BEGIN(Typename,type_name,TYPE_PARENT,flag
s) \
    \
static void      type_name##_init              (Typename      *self);
\
static void      type_name##_class_init        (Typename##Class
*klass); \
static gpointer type_name##_parent_class = NULL; \
static void      type_name##_class_intern_init (gpointer klass) \
{ \
    type_name##_parent_class = g_type_class_peek_parent (klass); \
    type_name##_class_init ((Typename##Class*) klass); \
} \
\
GType \
type_name##_get_type (void) \
{ \
    static volatile gsize g_define_type_id__volatile = 0; \
    if (g_once_init_enter (&g_define_type_id__volatile)) \
    { \
        GType g_define_type_id = \
            g_type_register_static_simple (TYPE_PARENT, \
                g_intern_static_string
(#Typename), \
                sizeof (Typename##Class), \
                (GClassInitFunc)
type_name##_class_intern_init, \
                sizeof (Typename), \
                (GInstanceInitFunc)
type_name##_init, \
                (GTypeFlags) flags); \
        { /* custom code follows */;
#define
G_DEFINE_BOXED_TYPE_WITH_CODE(Typename,type_name,copy_func,free_f
unc,_C_) _G_DEFINE_BOXED_TYPE_BEGIN (Typename, type_name,
copy_func, free_func){_C_}; _G_DEFINE_TYPE_EXTENDED_END()
#define _G_DEFINE_INTERFACE_WITH_CODE(TN,t_n,T_P,_C_)
_G_DEFINE_INTERFACE_EXTENDED_BEGIN(TN, t_n, T_P) {_C_};
_G_DEFINE_INTERFACE_EXTENDED_END()
#define _G_DEFINE_POINTER_TYPE_WITH_CODE(Typename,type_name,_C_)
_G_DEFINE_POINTER_TYPE_BEGIN (Typename, type_name) {_C_};
_G_DEFINE_TYPE_EXTENDED_END()
#define _G_DEFINE_ABSTRACT_TYPE_WITH_CODE(TN,t_n,T_P,_C_)
_G_DEFINE_TYPE_EXTENDED_BEGIN (TN, t_n, T_P, G_TYPE_FLAG_ABSTRACT)
{_C_}; _G_DEFINE_TYPE_EXTENDED_END()
#define _G_DEFINE_TYPE_EXTENDED(TN,t_n,T_P,_f_,_C_)
_G_DEFINE_TYPE_EXTENDED_BEGIN (TN, t_n, T_P, _f_) {_C_};
_G_DEFINE_TYPE_EXTENDED_END()
#define G_VALUE_INIT { 0, { { 0 } } }
#define G_IMPLEMENT_INTERFACE_DYNAMIC(TYPE_IFACE,iface_init)
{ const GInterfaceInfo g_implement_interface_info =
{ (GInterfaceInitFunc) iface_init, NULL, NULL };
g_type_module_add_interface (type_module, g_define_type_id,
TYPE_IFACE, &g_implement_interface_info); }
#define G_IMPLEMENT_INTERFACE(TYPE_IFACE,iface_init) { \
    const GInterfaceInfo g_implement_interface_info = { \
        (GInterfaceInitFunc) iface_init, NULL, NULL \
    }; \
    g_type_add_interface_static (g_define_type_id, TYPE_IFACE,
&g_implement_interface_info); \
}

```



```

#define _G_DEFINE_INTERFACE_EXTENDED_END()      } g_once_init_leave
(&g_define_type_id__volatile,    g_define_type_id);    }    return
g_define_type_id__volatile; }
#define _G_DEFINE_TYPE_EXTENDED_END()          } g_once_init_leave
(&g_define_type_id__volatile,    g_define_type_id);    }    return
g_define_type_id__volatile; }

typedef gulong GType;
typedef struct _GTypeClass GTypeClass;
typedef struct _GTypeInstance {
    GTypeClass *g_class;
} GTypeInstance;
typedef struct _GValue GValue;
typedef enum {
    G_PARAM_READABLE = 1,
    G_PARAM_WRITABLE = 2,
    G_PARAM_CONSTRUCT = 4,
    G_PARAM_CONSTRUCT_ONLY = 8,
    G_PARAM_LAX_VALIDATION = 16,
    G_PARAM_STATIC_NAME = 32,
    G_PARAM_PRIVATE = G_PARAM_STATIC_NAME,
    G_PARAM_STATIC_NICK = 64,
    G_PARAM_STATIC_BLURB = 128,
    G_PARAM_DEPRECATED = 1 << 31
} GParamFlags;
typedef struct _GParamSpec {
    GTypeInstance g_type_instance;
    gchar *name;
    GParamFlags flags;
    GType value_type;
    GType owner_type;
    gchar *_nick;
    gchar *_blurb;
    GData *qdata;
    guint ref_count;
    guint param_id;
} GParamSpec;
typedef struct _GClosure {
    volatile guint ref_count:15;
    volatile guint meta_marshall_nouse:1;
    volatile guint n_guards:1;
    volatile guint n_fnotifiers:2;
    volatile guint n_inotifiers:8;
    volatile guint in_inotify:1;
    volatile guint floating:1;
    volatile guint derivative_flag:1;
    volatile guint in_marshall:1;
    volatile guint is_invalid:1;
    void (*marshall) (GClosure * closure, GValue * return_value,
                     guint n_param_values, const GValue * param_values,
                     gpointer invocation_hint, gpointer marshall_data);
    gpointer data;
    GClosureNotifyData *notifiers;
} GClosure;
typedef void (*GClosureNotify) (gpointer data, GClosure * closure);
typedef struct _GClosureNotifyData GClosureNotifyData;
typedef gboolean(*GTypeClassCacheFunc) (gpointer, GTypeClass *);

typedef struct _GObject {
    GTypeInstance g_type_instance;
    volatile guint ref_count;
    GData *qdata;
} GObject;
struct _GValue {
    GType g_type;
    union {

```

```

        gint v_int;
        guint v_uint;
        glong v_long;
        gulong v_ulong;
        gint64 v_int64;
        guint64 v_uint64;
        gfloat v_float;
        gdouble v_double;
        gpointer v_pointer;
    } data[2];
};

typedef struct _GObjectConstructParam {
    GParamSpec *pspec;
    GValue *value;
} GObjectConstructParam;

typedef struct _GObjectClass {
    GTypeClass g_type_class;
    GSList *construct_properties;
    GObject      *(*constructor)      (GType      type,      guint
n_construct_properties,      GObjectConstructParam      *
construct_properties);
    void (*set_property) (GObject * object, guint property_id,
                        const GValue * value, GParamSpec * pspec);
    void (*get_property) (GObject * object, guint property_id,
                        GValue * value, GParamSpec * pspec);
    void (*dispose) (GObject * object);
    void (*finalize) (GObject * object);
    void (*dispatch_properties_changed) (GObject * object, guint
n_pspecs,
                                GParamSpec * *pspecs);
    void (*notify) (GObject * object, GParamSpec * pspec);
    void (*constructed) (GObject * object);
    gsize flags;
    gpointer pdummy[6];
} GObjectClass;

typedef void (*GTypeInterfaceCheckFunc) (gpointer, gpointer);

typedef struct _GValueArray {
    guint n_values;
    GValue *values;
    guint n_preallocated;
} GValueArray;

typedef struct _GEnumValue {
    gint value;
    const gchar *value_name;
    const gchar *value_nick;
} GEnumValue;

typedef struct _GEnumClass {
    GTypeClass g_type_class;
    gint minimum;
    gint maximum;
    guint n_values;
    GEnumValue *values;
} GEnumClass;

typedef struct _GParamSpecPool GParamSpecPool;

typedef enum {
    G_TYPE_DEBUG_NONE = 0,
    G_TYPE_DEBUG_OBJECTS = 1,
    G_TYPE_DEBUG_SIGNALS = 2,
    G_TYPE_DEBUG_MASK = 3
} GTypeDebugFlags;

typedef struct _GTypePlugin GTypePlugin;

typedef enum {
    G_SIGNAL_MATCH_ID = 1,
    G_SIGNAL_MATCH_DETAIL = 2,
    G_SIGNAL_MATCH_CLOSURE = 4,

```

```

    G_SIGNAL_MATCH_FUNC = 8,
    G_SIGNAL_MATCH_DATA = 16,
    G_SIGNAL_MATCH_UNBLOCKED = 32
} GSignalMatchType;
typedef struct _GFlagsValue {
    guint value;
    const gchar *value_name;
    const gchar *value_nick;
} GFlagsValue;
typedef void (*GClosureMarshal) (GClosure * closure, GValue *
return_value,
                                guint n_param_values,
                                const GValue * param_values,
                                gpointer invocation_hint,
                                gpointer marshal_data);

typedef void (*GCallback) (void);
typedef struct _GFlagsClass {
    GTypeClass g_type_class;
    guint mask;
    guint n_values;
    GFlagsValue *values;
} GFlagsClass;
typedef enum {
    G_SIGNAL_RUN_FIRST = 1,
    G_SIGNAL_RUN_LAST = 2,
    G_SIGNAL_RUN_CLEANUP = 4,
    G_SIGNAL_NO_RECURSE = 8,
    G_SIGNAL_DETAILED = 16,
    G_SIGNAL_ACTION = 32,
    G_SIGNAL_NO_HOOKS = 64,
    G_SIGNAL_MUST_COLLECT = 1 << 7,
    G_SIGNAL_DEPRECATED = 1 << 8
} GSignalFlags;
typedef struct _GSignalInvocationHint {
    guint signal_id;
    GQuark detail;
    GSignalFlags run_type;
} GSignalInvocationHint;
typedef void (*GWeakNotify) (gpointer data,
                             GObject * where_the_object_was);

typedef void (*GBaseInitFunc) (gpointer);
typedef void (*GBaseFinalizeFunc) (gpointer);
typedef void (*GClassInitFunc) (gpointer g_class, gpointer
class_data);
typedef void (*GClassFinalizeFunc) (gpointer g_class, gpointer
class_data);
typedef void (*GInstanceInitFunc) (GTypeInstance * instance,
                                   gpointer g_class);

typedef union _GTypeCValue {
    gint v_int;
    glong v_long;
    gint64 v_int64;
    gdouble v_double;
    gpointer v_pointer;
} GTypeCValue;
typedef struct _GTypeValueTable {
    void (*value_init) (GValue * value);
    void (*value_free) (GValue * value);
    void (*value_copy) (const GValue * src_value, GValue *
dest_value);
    gpointer (*value_peek_pointer) (const GValue * value);
    gchar *collect_format;
    gchar *(*collect_value) (GValue * value, guint n_collect_values,
                             GTypeCValue * collect_values,
                             guint collect_flags);
    gchar *lcopy_format;

```

```

    gchar    *(*lcopy_value)    (const    GValue    *    value,    guint
n_collect_values,
                                GTypeCValue * collect_values,
                                guint collect_flags);
} GTypeValueTable;
typedef struct _GTypeInfo {
    guint16 class_size;
    GBaseInitFunc base_init;
    GBaseFinalizeFunc base_finalize;
    GClassInitFunc class_init;
    GClassFinalizeFunc class_finalize;
    gconstpointer class_data;
    guint16 instance_size;
    guint16 n_preallocs;
    GInstanceInitFunc instance_init;
    const GTypeValueTable *value_table;
} GTypeInfo;
typedef enum {
    G_TYPE_FLAG_ABSTRACT = 16,
    G_TYPE_FLAG_VALUE_ABSTRACT = 32
} GTypeFlags;
typedef struct _GTypeModule {
    GObject parent_instance;
    guint use_count;
    GSList *type_infos;
    GSList *interface_infos;
    gchar *name;
} GTypeModule;
typedef void (*GInterfaceInitFunc) (gpointer g_iface, gpointer
iface_data);
typedef void (*GInterfaceFinalizeFunc) (gpointer g_iface,
gpointer iface_data);
typedef struct _GInterfaceInfo {
    GInterfaceInitFunc interface_init;
    GInterfaceFinalizeFunc interface_finalize;
    gpointer interface_data;
} GInterfaceInfo;
typedef gboolean(*GSignalAccumulator) (GSignalInvocationHint *
ihint,
                                GValue * return_accu,
                                const GValue * handler_return,
                                gpointer data);
typedef GClosureMarshal GSignalCMarshal;
typedef enum {
    G_CONNECT_AFTER = 1,
    G_CONNECT_SWAPPED = 2
} GConnectFlags;
typedef enum {
    G_TYPE_FLAG_CLASSSED = 1,
    G_TYPE_FLAG_INSTANTIATABLE = 2,
    G_TYPE_FLAG_DERIVABLE = 4,
    G_TYPE_FLAG_DEEP_DERIVABLE = 8
} GTypeFundamentalFlags;
typedef struct _GTypeFundamentalInfo {
    GTypeFundamentalFlags type_flags;
} GTypeFundamentalInfo;
typedef struct _GTypeQuery {
    GType type;
    const gchar *type_name;
    guint class_size;
    guint instance_size;
} GTypeQuery;
typedef gboolean(*GSignalEmissionHook) (GSignalInvocationHint *
ihint,
                                guint n_param_values,
                                const GValue * param_values,

```

```

                                gpointer data);
typedef void (*GValueTransform) (const GValue * src_value,
                                GValue * dest_value);
typedef struct _GParameter {
    const gchar *name;
    GValue value;
} GParameter;
typedef struct _GParamSpecTypeInfo {
    guint16 instance_size;
    guint16 n_preallocs;
    void (*instance_init) (GParamSpec * pspec);
    GType value_type;
    void (*finalize) (GParamSpec * pspec);
    void (*value_set_default) (GParamSpec * pspec, GValue * value);
    gboolean (*value_validate) (GParamSpec * pspec, GValue * value);
    gint (*values_cmp) (GParamSpec * pspec, const GValue * value1,
                       const GValue * value2);
} GParamSpecTypeInfo;
typedef gpointer (*GBoxedCopyFunc) (gpointer);
typedef void (*GBoxedFreeFunc) (gpointer);
typedef struct _GSignalQuery {
    guint signal_id;
    const gchar *signal_name;
    GType itype;
    GSignalFlags signal_flags;
    GType return_type;
    guint n_params;
    const GType *param_types;
} GSignalQuery;
typedef struct _GTypeInterface {
    GType g_type;
    GType g_instance_type;
} GTypeInterface;
typedef void (*GTypePluginCompleteInterfaceInfo) (GTypePlugin *
plugin,
                                                    GType instance_type,
                                                    GType interface_type,
                                                    GInterfaceInfo * info);

typedef struct _GParamSpecObject {
    GParamSpec parent_instance;
} GParamSpecObject;
typedef void (*GTypePluginUnuse) (GTypePlugin * plugin);
typedef void (*GTypePluginUse) (GTypePlugin * plugin);
typedef void (*GTypePluginCompleteTypeInfo) (GTypePlugin * plugin,
                                              GType gtype, GTypeInfo * info,
                                              GTypeValueTable *
                                              value_table);

typedef struct _GTypePluginClass {
    GTypeInterface base_iface;
    GTypePluginUse use_plugin;
    GTypePluginUnuse unuse_plugin;
    GTypePluginCompleteTypeInfo complete_type_info;
    GTypePluginCompleteInterfaceInfo complete_interface_info;
} GTypePluginClass;
typedef struct _GCClosure {
    GClosure closure;
    gpointer callback;
} GCClosure;
typedef struct _GParamSpecUnichar {
    GParamSpec parent_instance;
    gunichar default_value;
} GParamSpecUnichar;
typedef struct _GParamSpecUInt64 {
    GParamSpec parent_instance;
    guint64 minimum;
    guint64 maximum;

```

```

    guint64 default_value;
} GParamSpecUInt64;
typedef struct _GParamSpecBoxed {
    GParamSpec parent_instance;
} GParamSpecBoxed;
typedef struct _GParamSpecOverride {
    GParamSpec parent_instance;
    GParamSpec *overridden;
} GParamSpecOverride;
typedef struct _GParamSpecClass {
    GTypeClass g_type_class;
    GType value_type;
    void (*finalize) (GParamSpec * pspec);
    void (*value_set_default) (GParamSpec * pspec, GValue * value);
    gboolean (*value_validate) (GParamSpec * pspec, GValue * value);
    gint (*values_cmp) (GParamSpec * pspec, const GValue * value1,
                       const GValue * value2);
    gpointer dummy[4];
} GParamSpecClass;
typedef struct _GParamSpecLong {
    GParamSpec parent_instance;
    glong minimum;
    glong maximum;
    glong default_value;
} GParamSpecLong;
typedef struct _GParamSpecEnum {
    GParamSpec parent_instance;
    GEnumClass *enum_class;
    gint default_value;
} GParamSpecEnum;
typedef struct _GParamSpecFloat {
    GParamSpec parent_instance;
    gfloat minimum;
    gfloat maximum;
    gfloat default_value;
    gfloat epsilon;
} GParamSpecFloat;
typedef struct _GParamSpecString {
    GParamSpec parent_instance;
    gchar *default_value;
    gchar *cset_first;
    gchar *cset_nth;
    gchar substitutor;
    guint null_fold_if_empty:1;
    guint ensure_non_null:1;
} GParamSpecString;
typedef struct _GParamSpecDouble {
    GParamSpec parent_instance;
    gdouble minimum;
    gdouble maximum;
    gdouble default_value;
    gdouble epsilon;
} GParamSpecDouble;
typedef struct _GParamSpecParam {
    GParamSpec parent_instance;
} GParamSpecParam;
typedef struct _GParamSpecValueArray {
    GParamSpec parent_instance;
    GParamSpec *element_spec;
    guint fixed_n_elements;
} GParamSpecValueArray;
typedef struct _GParamSpecFlags {
    GParamSpec parent_instance;
    GFlagsClass *flags_class;
    guint default_value;
} GParamSpecFlags;

```

```

typedef struct _GParamSpecInt64 {
    GParamSpec parent_instance;
    gint64 minimum;
    gint64 maximum;
    gint64 default_value;
} GParamSpecInt64;
typedef struct _GParamSpecPointer {
    GParamSpec parent_instance;
} GParamSpecPointer;
typedef struct _GParamSpecInt {
    GParamSpec parent_instance;
    gint minimum;
    gint maximum;
    gint default_value;
} GParamSpecInt;
typedef struct _GTypeModuleClass {
    GObjectClass parent_class;
    gboolean(*load) (GTypeModule * module);
    void(*unload) (GTypeModule * module);
    void(*reserved1) (void);
    void(*reserved2) (void);
    void(*reserved3) (void);
    void(*reserved4) (void);
} GTypeModuleClass;
typedef struct _GParamSpecUInt {
    GParamSpec parent_instance;
    guint minimum;
    guint maximum;
    guint default_value;
} GParamSpecUInt;
typedef struct _GParamSpecUChar {
    GParamSpec parent_instance;
    guint8 minimum;
    guint8 maximum;
    guint8 default_value;
} GParamSpecUChar;
typedef struct _GParamSpecULong {
    GParamSpec parent_instance;
    gulong minimum;
    gulong maximum;
    gulong default_value;
} GParamSpecULong;
typedef struct _GParamSpecChar {
    GParamSpec parent_instance;
    gint8 minimum;
    gint8 maximum;
    gint8 default_value;
} GParamSpecChar;
typedef struct _GParamSpecBoolean {
    GParamSpec parent_instance;
    gboolean default_value;
} GParamSpecBoolean;
typedef void (*GObjectFinalizeFunc) (GObject * object);
typedef void (*GObjectGetPropertyFunc) (GObject * object,
                                         quint property_id, GValue * value,
                                         GParamSpec * pspec);
typedef void (*GObjectSetPropertyFunc) (GObject * object,
                                         quint property_id,
                                         const GValue * value,
                                         GParamSpec * pspec);

typedef gchar *gchararray;
typedef gchar **GStrv;
typedef void (*GToggleNotify) (gpointer data, GObject * object,
                               gboolean is_last_ref);
typedef struct _GBinding GBinding;
typedef gboolean(*GBindingTransformFunc) (GBinding * binding,

```

```

const GValue * source_value,
GValue * target_value,
gpointer user_data);

typedef enum {
    G_BINDING_DEFAULT = 0,
    G_BINDING_BIDIRECTIONAL = 1 << 0,
    G_BINDING_SYNC_CREATE = 1 << 1,
    G_BINDING_INVERT_BOOLEAN = 1 << 2
} GBindingFlags;
typedef struct _GObject {
    GTypeInstance g_type_instance;
    volatile guint ref_count;
    GData *qdata;
} GInitiallyUnowned;
typedef struct _GObjectClass {
    GTypeClass g_type_class;
    GSList *construct_properties;
    GObject      *(*constructor)      (GType      type,      guint
n_construct_properties,      GObjectConstructParam      *
construct_properties);
    void (*set_property) (GObject * object, guint property_id,
                        const GValue * value, GParamSpec * pspec);
    void (*get_property) (GObject * object, guint property_id,
                        GValue * value, GParamSpec * pspec);
    void (*dispose) (GObject * object);
    void (*finalize) (GObject * object);
    void (*dispatch_properties_changed) (GObject * object, guint
n_pspecs,
                        GParamSpec * *pspecs);
    void (*notify) (GObject * object, GParamSpec * pspec);
    void (*constructed) (GObject * object);
    gsize flags;
    gpointer pdummy[6];
} GInitiallyUnownedClass;
struct GWeakRef {
    union {
        void *p;
    } priv;
};
typedef struct _GParamSpecGType {
    GParamSpec parent_instance;
    GType is_a_type;
} GParamSpecGType;
typedef struct _GParamSpecVariant {
    GParamSpec parent_instance;
    GVariantType *type;
    GVariant *default_value;
    gpointer padding[4];
} GParamSpecVariant;
typedef void (*GVaClosureMarshal) (GClosure * closure,
    GValue * return_value,
    gpointer instance, va_list * args,
    gpointer marshal_data, int n_params,
    GType * param_types);

typedef struct _GAtomicArray {
    volatile gpointer data;
} GAtomicArray;
typedef struct _GRealClosure {
    GClosureMarshal meta_marshal;
    gpointer meta_marshal_data;
    GVaClosureMarshal va_meta_marshal;
    GVaClosureMarshal va_marshal;
    GClosure closure;
} GRealClosure;
typedef GVaClosureMarshal GSignalCVaMarshaller;

```



```

extern GType g_array_get_type(void);
extern GType g_binding_flags_get_type(void);
extern GBindingFlags g_binding_get_flags(GBinding * binding);
extern GObject *g_binding_get_source(GBinding * binding);
extern const gchar *g_binding_get_source_property(GBinding *
binding);
extern GObject *g_binding_get_target(GBinding * binding);
extern const gchar *g_binding_get_target_property(GBinding *
binding);
extern GType g_binding_get_type(void);
extern gpointer g_boxed_copy(GType boxed_type, gpointer
src_boxed);
extern void g_boxed_free(GType boxed_type, gpointer boxed);
extern GType g_boxed_type_register_static(const gchar * name,
GBoxedCopyFunc boxed_copy,
GBoxedFreeFunc boxed_free);
extern GType g_byte_array_get_type(void);
extern GType g_bytes_get_type(void);
extern void g_cclosure_marshal_BOOLEAN__FLAGS(GClosure *, GValue *,
guint,
const GValue *, gpointer,
gpointer);
extern void g_cclosure_marshal_STRING__OBJECT_POINTER(GClosure *,
GValue *,
guint,
const GValue *,
gpointer, gpointer);
extern void g_cclosure_marshal_VOID__BOOLEAN(GClosure * closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__BOOLEANv(GClosure * closure,
GValue * return_value,
gpointer instance,
va_list args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__BOXED(GClosure * closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__BOXEDv(GClosure * closure,
GValue * return_value,
gpointer instance,
va_list args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__CHAR(GClosure * closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__CHARv(GClosure * closure,
GValue * return_value,
gpointer instance, va_list
args,
gpointer marshal_data,
int n_params,
GType * param_types);

```

```

extern void g_cclosure_marshal_VOID__DOUBLE(GClosure * closure,
                                             GValue * return_value,
                                             guint n_param_values,
                                             const GValue * param_value,
                                             gpointer invocation_hint,
                                             gpointer marshal_data);
extern void g_cclosure_marshal_VOID__DOUBLEv(GClosure * closure,
                                              GValue * return_value,
                                              gpointer instance,
                                              va_list args,
                                              gpointer marshal_data,
                                              int n_params,
                                              GType * param_types);
extern void g_cclosure_marshal_VOID__ENUM(GClosure * closure,
                                           GValue * return_value,
                                           guint n_param_values,
                                           const GValue * param_value,
                                           gpointer invocation_hint,
                                           gpointer marshal_data);
extern void g_cclosure_marshal_VOID__ENUMv(GClosure * closure,
                                             GValue * return_value,
                                             gpointer instance,
                                             va_list
args,
                                             gpointer marshal_data,
                                             int n_params,
                                             GType * param_types);
extern void g_cclosure_marshal_VOID__FLAGS(GClosure * closure,
                                             GValue * return_value,
                                             guint n_param_values,
                                             const GValue * param_value,
                                             gpointer invocation_hint,
                                             gpointer marshal_data);
extern void g_cclosure_marshal_VOID__FLAGsv(GClosure * closure,
                                              GValue * return_value,
                                              gpointer instance,
                                              va_list args,
                                              gpointer marshal_data,
                                              int n_params,
                                              GType * param_types);
extern void g_cclosure_marshal_VOID__FLOAT(GClosure * closure,
                                             GValue * return_value,
                                             guint n_param_values,
                                             const GValue * param_value,
                                             gpointer invocation_hint,
                                             gpointer marshal_data);
extern void g_cclosure_marshal_VOID__FLOATv(GClosure * closure,
                                              GValue * return_value,
                                              gpointer instance,
                                              va_list args,
                                              gpointer marshal_data,
                                              int n_params,
                                              GType * param_types);
extern void g_cclosure_marshal_VOID__INT(GClosure * closure,
                                           GValue * return_value,
                                           guint n_param_values,
                                           const GValue * param_value,
                                           gpointer invocation_hint,
                                           gpointer marshal_data);
extern void g_cclosure_marshal_VOID__INTv(GClosure * closure,
                                             GValue * return_value,
                                             gpointer instance,
                                             va_list args,
                                             gpointer marshal_data,
                                             int n_params,
                                             GType * param_types);
extern void g_cclosure_marshal_VOID__LONG(GClosure * closure,
                                           GValue * return_value,

```

```

    guint n_param_values,
    const GValue * param_value,
    gpointer invocation_hint,
    gpointer marshal_data);
extern void g_cclosure_marshal_VOID__LONGv(GClosure * closure,
    GValue * return_value,
    gpointer instance, va_list
args,
    gpointer marshal_data,
    int n_params,
    GType * param_types);
extern void g_cclosure_marshal_VOID__OBJECT(GClosure * closure,
    GValue * return_value,
    guint n_param_values,
    const GValue * param_value,
    gpointer invocation_hint,
    gpointer marshal_data);
extern void g_cclosure_marshal_VOID__OBJECTv(GClosure * closure,
    GValue * return_value,
    gpointer instance,
    va_list args,
    gpointer marshal_data,
    int n_params,
    GType * param_types);
extern void g_cclosure_marshal_VOID__PARAM(GClosure * closure,
    GValue * return_value,
    guint n_param_values,
    const GValue * param_value,
    gpointer invocation_hint,
    gpointer marshal_data);
extern void g_cclosure_marshal_VOID__PARAMv(GClosure * closure,
    GValue * return_value,
    gpointer instance,
    va_list args,
    gpointer marshal_data,
    int n_params,
    GType * param_types);
extern void g_cclosure_marshal_VOID__POINTER(GClosure * closure,
    GValue * return_value,
    guint n_param_values,
    const GValue * param_value,
    gpointer invocation_hint,
    gpointer marshal_data);
extern void g_cclosure_marshal_VOID__POINTERv(GClosure * closure,
    GValue * return_value,
    gpointer instance,
    va_list args,
    gpointer marshal_data,
    int n_params,
    GType * param_types);
extern void g_cclosure_marshal_VOID__STRING(GClosure * closure,
    GValue * return_value,
    guint n_param_values,
    const GValue * param_value,
    gpointer invocation_hint,
    gpointer marshal_data);
extern void g_cclosure_marshal_VOID__STRINGv(GClosure * closure,
    GValue * return_value,
    gpointer instance,
    va_list args,
    gpointer marshal_data,
    int n_params,
    GType * param_types);
extern void g_cclosure_marshal_VOID__UCHAR(GClosure * closure,
    GValue * return_value,
    guint n_param_values,

```

```

const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__UCHARv(GClosure * closure,
GValue * return_value,
gpointer instance,
va_list args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__UINT(GClosure * closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__UINT_POINTER(GClosure *
closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__UINT_POINTERv(GClosure *
closure,
GValue * return_value,
gpointer instance,
va_list args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__UINTv(GClosure * closure,
GValue * return_value,
gpointer instance, va_list
args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__ULONG(GClosure * closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__ULONGv(GClosure * closure,
GValue * return_value,
gpointer instance,
va_list args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__VARIANT(GClosure * closure,
GValue * return_value,
guint n_param_values,
const GValue * param_value,
gpointer invocation_hint,
gpointer marshal_data);
extern void g_cclosure_marshal_VOID__VARIANTv(GClosure * closure,
GValue * return_value,
gpointer instance,
va_list args,
gpointer marshal_data,
int n_params,
GType * param_types);
extern void g_cclosure_marshal_VOID__VOID(GClosure * closure,

```

```

        GValue * return_value,
        quint n_param_values,
        const GValue * param_value,
        gpointer invocation_hint,
        gpointer marshal_data);
extern void g_cclosure_marshal_VOID__VOIDv(GClosure * closure,
        GValue * return_value,
        gpointer instance, va_list
args,
        gpointer marshal_data,
        int n_params,
        GType * param_types);
extern void g_cclosure_marshal_generic(GClosure * closure,
        GValue * return_gvalue,
        quint n_param_values,
        const GValue * param_values,
        gpointer invocation_hint,
        gpointer marshal_data);
extern void g_cclosure_marshal_generic_va(GClosure * closure,
        GValue * return_value,
        gpointer instance,
        va_list args_list,
        gpointer marshal_data,
        int n_params,
        GType * param_types);
extern GClosure *g_cclosure_new(GCallback callback_func,
        gpointer user_data,
        GClosureNotify destroy_data);
extern GClosure *g_cclosure_new_object(GCallback callback_func,
        GObject * object);
extern GClosure *g_cclosure_new_object_swap(GCallback
callback_func,
        GObject * object);
extern GClosure *g_cclosure_new_swap(GCallback callback_func,
        gpointer user_data,
        GClosureNotify destroy_data);
extern void g_clear_object(volatile GObject * *object_ptr);
extern void g_closure_add_finalize_notifier(GClosure * closure,
        gpointer notify_data,
        GClosureNotify notify_func);
extern void g_closure_add_invalidate_notifier(GClosure * closure,
        gpointer notify_data,
        GClosureNotify notify_func);
extern void g_closure_add_marshal_guards(GClosure * closure,
        gpointer pre_marshal_data,
        GClosureNotify
pre_marshal_notify,
        gpointer post_marshal_data,
        GClosureNotify
post_marshal_notify);
extern GType g_closure_get_type(void);
extern void g_closure_invalidate(GClosure * closure);
extern void g_closure_invoke(GClosure * closure, GValue *
return_value,
        quint n_param_values,
        const GValue * param_values,
        gpointer invocation_hint);
extern GClosure *g_closure_new_object(guint sizeof_closure,
        GObject * object);
extern GClosure *g_closure_new_simple(guint sizeof_closure,
        gpointer data);
extern GClosure *g_closure_ref(GClosure * closure);
extern void g_closure_remove_finalize_notifier(GClosure * closure,
        gpointer notify_data,
        GClosureNotify
notify_func);

```

```

extern void g_closure_remove_invalidate_notifier(GClosure *
closure,
                                                gpointer notify_data,
                                                GClosureNotify
                                                notify_func);
extern void g_closure_set_marshal(GClosure * closure,
GClosureMarshal marshal);
extern void g_closure_set_meta_marshal(GClosure * closure,
gpointer marshal_data,
GClosureMarshal meta_marshal);
extern void g_closure_sink(GClosure * closure);
extern void g_closure_unref(GClosure * closure);
extern GType g_date_get_type(void);
extern GType g_date_time_get_type(void);
extern void g_enum_complete_type_info(GType g_enum_type, GTypeInfo
* info,
const GEnumValue * const_values);
extern GEnumValue *g_enum_get_value(GEnumClass * enum_class, gint
value);
extern GEnumValue *g_enum_get_value_by_name(GEnumClass *
enum_class,
const gchar * name);
extern GEnumValue *g_enum_get_value_by_nick(GEnumClass *
enum_class,
const gchar * nick);
extern GType g_enum_register_static(const gchar * name,
const GEnumValue *
const_static_values);
extern GType g_error_get_type(void);
extern void g_flags_complete_type_info(GType g_flags_type,
GTypeInfo * info,
const GFlagsValue * const_values);
extern GFlagsValue *g_flags_get_first_value(GFlagsClass *
flags_class,
guint value);
extern GFlagsValue *g_flags_get_value_by_name(GFlagsClass *
flags_class,
const gchar * name);
extern GFlagsValue *g_flags_get_value_by_nick(GFlagsClass *
flags_class,
const gchar * nick);
extern GType g_flags_register_static(const gchar * name,
const GFlagsValue *
const_static_values);
extern GType g_gstring_get_type(void);
extern GType g_gtype_get_type(void);
extern GType g_hash_table_get_type(void);
extern GType g_initially_unowned_get_type(void);
extern GType g_io_channel_get_type(void);
extern GType g_io_condition_get_type(void);
extern GType g_key_file_get_type(void);
extern GType g_main_context_get_type(void);
extern GType g_main_loop_get_type(void);
extern GType g_match_info_get_type(void);
extern void g_object_add_toggle_ref(GObject * object, GToggleNotify
notify,
gpointer data);
extern void g_object_add_weak_pointer(GObject * object,
gpointer * weak_pointer_location);
extern GBinding *g_object_bind_property(gpointer source,
const gchar * source_property,
gpointer target,
const gchar * target_property,
GBindingFlags flags);
extern GBinding *g_object_bind_property_full(gpointer source,
const gchar * source_property,

```

```

gpointer target,
const gchar * target_property,
GBindingFlags flags,
GBindingTransformFunc
transform_to,
GBindingTransformFunc
transform_from,
gpointer user_data,
GDestroyNotify notify);
extern GBinding *g_object_bind_property_with_closures(gpointer
source,

const gchar *
source_property,
gpointer target,
const gchar *
target_property,
GBindingFlags flags,
GClosure *
transform_to,
GClosure *
transform_from);
extern GParamSpec *g_object_class_find_property(GObjectClass *
oclass,

const gchar *
property_name);
extern void g_object_class_install_properties(GObjectClass *
oclass,

guint n_specs,
GParamSpec * *pspecs);
extern void g_object_class_install_property(GObjectClass * oclass,
guint property_id,
GParamSpec * pspec);
extern GParamSpec **g_object_class_list_properties(GObjectClass *
oclass,

guint * n_properties);
extern void g_object_class_override_property(GObjectClass * oclass,
guint property_id,
const gchar * name);
extern gsize g_object_compat_control(gsize what, void *data);
extern gpointer g_object_connect(gpointer object,
const gchar * signal_spec, ...);
extern void g_object_disconnect(gpointer object, const gchar *
signal_spec,

...);
extern void g_object_force_floating(GObject *);
extern void g_object_freeze_notify(GObject * object);
extern void g_object_get(gpointer object,
const gchar * first_property_name, ...);
extern void *g_object_get_data(GObject * object, const gchar * key);
extern void g_object_get_property(GObject * object,
const gchar * property_name,
GValue * value);
extern void *g_object_get_qdata(GObject * object, GQuark quark);
extern GType g_object_get_type(void);
extern void g_object_get_valist(GObject * object,
const gchar * first_property_name,
va_list var_args);
extern GParamSpec *g_object_interface_find_property(gpointer
g_iface,

const gchar *
property_name);
extern void g_object_interface_install_property(gpointer g_iface,
GParamSpec * pspec);
extern GParamSpec **g_object_interface_list_properties(gpointer
g_iface,

guint *

```

```

                                n_properties_p);
extern gboolean g_object_is_floating(gpointer object);
extern gpointer g_object_new(GType object_type,
                             const gchar * first_property_name, ...);
extern GObject *g_object_new_valist(GType object_type,
                                    const gchar * first_property_name,
                                    va_list var_args);
extern gpointer g_object_newv(GType object_type, guint n_parameters,
                              GParameter * parameters);
extern void g_object_notify(GObject * object, const gchar *
property_name);
extern void g_object_notify_by_pspec(GObject * object, GParamSpec
* pspec);
extern gpointer g_object_ref(gpointer object);
extern gpointer g_object_ref_sink(gpointer object);
extern void g_object_remove_toggle_ref(GObject * object,
                                       GToggleNotify notify,
                                       gpointer data);
extern void g_object_remove_weak_pointer(GObject * object,
                                       gpointer
                                       *
weak_pointer_location);
extern void g_object_run_dispose(GObject * object);
extern void g_object_set(gpointer object,
                        const gchar * first_property_name, ...);
extern void g_object_set_data(GObject * object, const gchar * key,
                              gpointer data);
extern void g_object_set_data_full(GObject * object, const gchar *
key,
                                gpointer data, GDestroyNotify
destroy);
extern void g_object_set_property(GObject * object,
                                const gchar * property_name,
                                const GValue * value);
extern void g_object_set_qdata(GObject * object, GQuark quark,
                              gpointer data);
extern void g_object_set_qdata_full(GObject * object, GQuark quark,
                                   gpointer data, GDestroyNotify
destroy);
extern void g_object_set_valist(GObject * object,
                                const gchar * first_property_name,
                                va_list var_args);
extern void *g_object_steal_data(GObject * object, const gchar *
key);
extern void *g_object_steal_qdata(GObject * object, GQuark quark);
extern void g_object_thaw_notify(GObject * object);
extern void g_object_unref(gpointer object);
extern void g_object_watch_closure(GObject * object, GClosure *
closure);
extern void g_object_weak_ref(GObject * object, GWeakNotify notify,
                              gpointer data);
extern void g_object_weak_unref(GObject * object, GWeakNotify
notify,
                              gpointer data);
extern GParamSpec *g_param_spec_boolean(const gchar * name,
                                       const gchar * nick,
                                       const gchar * blurb,
                                       gboolean default_value,
                                       GParamFlags flags);
extern GParamSpec *g_param_spec_boxed(const gchar * name,
                                       const gchar * nick,
                                       const gchar * blurb,
                                       GType boxed_type, GParamFlags
flags);
extern GParamSpec *g_param_spec_char(const gchar * name,
                                       const gchar * nick,
                                       const gchar * blurb, gint8 minimum,

```



```

        gint8 maximum, gint8 default_value,
        GParamFlags flags);
extern GParamSpec *g_param_spec_double(const gchar * name,
        const gchar * nick,
        const gchar * blurb,
        gdouble minimum, gdouble maximum,
        gdouble default_value,
        GParamFlags flags);
extern GParamSpec *g_param_spec_enum(const gchar * name,
        const gchar * nick,
        const gchar * blurb, GType enum_type,
        gint default_value,
        GParamFlags flags);
extern GParamSpec *g_param_spec_flags(const gchar * name,
        const gchar * nick,
        const gchar * blurb,
        GType flags_type,
        guint default_value,
        GParamFlags flags);
extern GParamSpec *g_param_spec_float(const gchar * name,
        const gchar * nick,
        const gchar * blurb, gfloat minimum,
        gfloat maximum, gfloat
        default_value,
        GParamFlags flags);
extern const gchar *g_param_spec_get_blurb(GParamSpec * pspec);
extern const gchar *g_param_spec_get_name(GParamSpec * pspec);
extern const gchar *g_param_spec_get_nick(GParamSpec * pspec);
extern gpointer g_param_spec_get_qdata(GParamSpec * pspec, GQuark
quark);
extern GParamSpec *g_param_spec_get_redirect_target(GParamSpec *
pspec);
extern GParamSpec *g_param_spec_gtype(const gchar *, const gchar *,
        const gchar *, GType, GParamFlags);
extern GParamSpec *g_param_spec_int(const gchar * name, const gchar
* nick,
        const gchar * blurb, gint minimum,
        gint maximum, gint default_value,
        GParamFlags flags);
extern GParamSpec *g_param_spec_int64(const gchar * name,
        const gchar * nick,
        const gchar * blurb, gint64 minimum,
        gint64 maximum, gint64
        default_value,
        GParamFlags flags);
extern gpointer g_param_spec_internal(GType param_type, const gchar
* name,
        const gchar * nick,
        const gchar * blurb,
        GParamFlags flags);
extern GParamSpec *g_param_spec_long(const gchar * name,
        const gchar * nick,
        const gchar * blurb, glong minimum,
        glong maximum, glong default_value,
        GParamFlags flags);
extern GParamSpec *g_param_spec_object(const gchar * name,
        const gchar * nick,
        const gchar * blurb,
        GType object_type,
        GParamFlags flags);
extern GParamSpec *g_param_spec_override(const gchar * name,
        GParamSpec * overridden);
extern GParamSpec *g_param_spec_param(const gchar * name,
        const gchar * nick,
        const gchar * blurb,

```

```

                                GType    param_type,    GParamFlags
flags);
extern GParamSpec *g_param_spec_pointer(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb,
                                GParamFlags flags);
extern void g_param_spec_pool_insert(GParamSpecPool * pool,
                                GParamSpec * pspec,    GType
owner_type);
extern GParamSpec **g_param_spec_pool_list(GParamSpecPool * pool,
                                GType owner_type,
                                guint * n_pspecs_p);
extern GList *g_param_spec_pool_list_owned(GParamSpecPool * pool,
                                GType owner_type);
extern GParamSpec *g_param_spec_pool_lookup(GParamSpecPool * pool,
                                const gchar * param_name,
                                GType owner_type,
                                gboolean walk_ancestors);
extern GParamSpecPool *g_param_spec_pool_new(gboolean
type_prefixing);
extern void g_param_spec_pool_remove(GParamSpecPool * pool,
                                GParamSpec * pspec);
extern GParamSpec *g_param_spec_ref(GParamSpec * pspec);
extern GParamSpec *g_param_spec_ref_sink(GParamSpec *);
extern void g_param_spec_set_qdata(GParamSpec * pspec, GQuark quark,
                                gpointer data);
extern void g_param_spec_set_qdata_full(GParamSpec * pspec, GQuark
quark,
                                gpointer data,
                                GDestroyNotify destroy);
extern void g_param_spec_sink(GParamSpec * pspec);
extern gpointer g_param_spec_steal_qdata(GParamSpec * pspec, GQuark
quark);
extern GParamSpec *g_param_spec_string(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb,
                                const gchar * default_value,
                                GParamFlags flags);
extern GType *g_param_spec_types;
extern GParamSpec *g_param_spec_uchar(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb, guint8 minimum,
                                guint8 maximum,    guint8
default_value,
                                GParamFlags flags);
extern GParamSpec *g_param_spec_uint(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb, guint minimum,
                                guint maximum,    guint default_value,
                                GParamFlags flags);
extern GParamSpec *g_param_spec_uint64(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb,
                                guint64 minimum,    guint64 maximum,
                                guint64 default_value,
                                GParamFlags flags);
extern GParamSpec *g_param_spec_ulong(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb, gulong minimum,
                                gulong maximum,    gulong
default_value,
                                GParamFlags flags);
extern GParamSpec *g_param_spec_unichar(const gchar * name,
                                const gchar * nick,
                                const gchar * blurb,
                                gunichar default_value,

```

```

        GParamFlags flags);
extern void g_param_spec_unref(GParamSpec * pspec);
extern GParamSpec *g_param_spec_value_array(const gchar * name,
        const gchar * nick,
        const gchar * blurb,
        GParamSpec * element_spec,
        GParamFlags flags);
extern GParamSpec *g_param_spec_variant(const gchar * name,
        const gchar * nick,
        const gchar * blurb,
        GVariantType * type,
        GVariant * default_value,
        GParamFlags flags);
extern GType g_param_type_register_static(const gchar * name,
        const GParamSpecTypeInfo *
        pspec_info);
extern gboolean g_param_value_convert(GParamSpec * pspec,
        const GValue * src_value,
        GValue * dest_value,
        gboolean strict_validation);
extern gboolean g_param_value_defaults(GParamSpec * pspec, GValue
* value);
extern void g_param_value_set_default(GParamSpec * pspec, GValue *
value);
extern gboolean g_param_value_validate(GParamSpec * pspec, GValue
* value);
extern gint g_param_values_cmp(GParamSpec * pspec, const GValue *
value1,
        const GValue * value2);
extern GType g_pointer_type_register_static(const gchar * name);
extern GType g_ptr_array_get_type(void);
extern GType g_regex_get_type(void);
extern
        gboolean
g_signal_accumulator_first_wins(GSignalInvocationHint *
        ihint,
        GValue * return_accu,
        const GValue *
        handler_return,
        gpointer dummy);
extern
        gboolean
g_signal_accumulator_true_handled(GSignalInvocationHint *
        ihint,
        GValue * return_accu,
        const GValue *
        handler_return,
        gpointer dummy);
extern gulong g_signal_add_emission_hook(guint signal_id, GQuark
detail,
        GSignalEmissionHook hook_func,
        gpointer hook_data,
        GDestroyNotify data_destroy);
extern void g_signal_chain_from_overridden(const GValue *
        instance_and_params,
        GValue * return_value);
extern void g_signal_chain_from_overridden_handler(gpointer
instance, ...);
extern gulong g_signal_connect_closure(gpointer instance,
        const gchar * detailed_signal,
        GClosure * closure, gboolean
after);
extern gulong g_signal_connect_closure_by_id(gpointer instance,
        guint signal_id,
        GQuark detail,
        GClosure * closure,
        gboolean after);
extern gulong g_signal_connect_data(gpointer instance,

```

```

const gchar * detailed_signal,
GCallback c_handler, gpointer data,
GClosureNotify destroy_data,
GConnectFlags connect_flags);
extern gulong g_signal_connect_object(gpointer instance,
const gchar * detailed_signal,
GCallback c_handler,
gpointer gobject,
GConnectFlags connect_flags);
extern void g_signal_emit(gpointer instance, guint signal_id,
GQuark detail, ...);
extern void g_signal_emit_by_name(gpointer instance,
const gchar * detailed_signal, ...);
extern void g_signal_emit_valist(gpointer instance, guint signal_id,
GQuark detail, va_list var_args);
extern void g_signal_emitv(const GValue * instance_and_params,
guint signal_id, GQuark detail,
GValue * return_value);
extern GSignalInvocationHint
*g_signal_get_invocation_hint(gpointer
instance);
extern void g_signal_handler_block(gpointer instance, gulong
handler_id);
extern void g_signal_handler_disconnect(gpointer instance,
gulong handler_id);
extern gulong g_signal_handler_find(gpointer instance,
GSignalMatchType mask, guint
signal_id,
GQuark detail, GClosure * closure,
gpointer func, gpointer data);
extern gboolean g_signal_handler_is_connected(gpointer instance,
gulong handler_id);
extern void g_signal_handler_unblock(gpointer instance, gulong
handler_id);
extern guint g_signal_handlers_block_matched(gpointer instance,
GSignalMatchType mask,
guint signal_id,
GQuark detail,
GClosure * closure,
gpointer func, gpointer data);
extern guint g_signal_handlers_disconnect_matched(gpointer
instance,
GSignalMatchType mask,
guint signal_id,
GQuark detail,
GClosure * closure,
gpointer func,
gpointer data);
extern guint g_signal_handlers_unblock_matched(gpointer instance,
GSignalMatchType mask,
guint signal_id,
GQuark detail,
GClosure * closure,
gpointer func,
gpointer data);
extern gboolean g_signal_has_handler_pending(gpointer instance,
guint signal_id,
GQuark detail,
gboolean may_be_blocked);
extern guint *g_signal_list_ids(GType itype, guint * n_ids);
extern guint g_signal_lookup(const gchar * name, GType itype);
extern const gchar *g_signal_name(guint signal_id);
extern guint g_signal_new(const gchar * signal_name, GType itype,
GSignalFlags signal_flags, guint
class_offset,
GSignalAccumulator accumulator,

```

```

        gpointer accu_data,
        GSignalCMarshaller c_marshaller,
        GType return_type, guint n_params, ...);
extern guint g_signal_new_class_handler(const char *signal_name,
        GType itype,
        GSignalFlags signal_flags,
        GCallback class_handler,
        GSignalAccumulator accumulator,
        gpointer accu_data,
        GSignalCMarshaller c_marshaller,
        GType return_type, guint n_params,
        ...);
extern guint g_signal_new_valist(const gchar * signal_name, GType
itype,
        GSignalFlags signal_flags,
        GClosure * class_closure,
        GSignalAccumulator accumulator,
        gpointer accu_data,
        GSignalCMarshaller c_marshaller,
        GType return_type, guint n_params,
        va_list args);
extern guint g_signal_newv(const gchar * signal_name, GType itype,
        GSignalFlags signal_flags,
        GClosure * class_closure,
        GSignalAccumulator accumulator,
        gpointer accu_data,
        GSignalCMarshaller c_marshaller,
        GType return_type, guint n_params,
        GType * param_types);
extern void g_signal_override_class_closure(guint signal_id,
        GType instance_type,
        GClosure * class_closure);
extern void g_signal_override_class_handler(const char
*signal_name,
        GType instance_type,
        GCallback class_handler);
extern gboolean g_signal_parse_name(const gchar * detailed_signal,
        GType itype, guint * signal_id_p,
        GQuark * detail_p,
        gboolean force_detail_quark);
extern void g_signal_query(guint signal_id, GSignalQuery * query);
extern void g_signal_remove_emission_hook(guint signal_id, gulong
hook_id);
extern void g_signal_set_va_marshaller(guint signal_id,
        GType instance_type,
        GSignalCVaMarshaller
va_marshaller);
extern void g_signal_stop_emission(gpointer instance, guint
signal_id,
        GQuark detail);
extern void g_signal_stop_emission_by_name(gpointer instance,
        const gchar * detailed_signal);
extern GClosure *g_signal_type_cclosure_new(GType itype,
        guint struct_offset);
extern GType g_source_get_type(void);
extern void g_source_set_closure(GSource * source, GClosure *
closure);
extern void g_source_set_dummy_callback(GSource * source);
extern gchar *g_strdup_value_contents(const GValue * value);
extern GType g_strv_get_type(void);
extern void g_type_add_class_cache_func(gpointer cache_data,
        GTypeClassCacheFunc cache_func);
extern void g_type_add_class_private(GType class_type, gsize
private_size);
extern void g_type_add_interface_check(gpointer check_data,

```

```

                                GTypeInterfaceCheckFunc
check_func);
extern void g_type_add_interface_dynamic(GType instance_type,
                                         GType interface_type,
                                         GTypePlugin * plugin);
extern void g_type_add_interface_static(GType instance_type,
                                         GType interface_type,
                                         const GInterfaceInfo * info);
extern GTypeClass *g_type_check_class_cast(GTypeClass * g_class,
                                           GType is_a_type);
extern gboolean g_type_check_class_is_a(GTypeClass * g_class,
                                         GType is_a_type);
extern gboolean g_type_check_instance(GTypeInstance * instance);
extern GTypeInstance *g_type_check_instance_cast(GTypeInstance *
instance,
                                                GType iface_type);
extern gboolean g_type_check_instance_is_a(GTypeInstance *
instance,
                                           GType iface_type);
extern gboolean g_type_check_is_value_type(GType type);
extern gboolean g_type_check_value(GValue * value);
extern gboolean g_type_check_value_holds(GValue * value, GType
type);
extern GType *g_type_children(GType type, guint * n_children);
extern void g_type_class_add_private(gpointer g_class, gsize
private_size);
extern gpointer g_type_class_get_private(GTypeClass * klass,
                                         GType private_type);
extern gpointer g_type_class_peek(GType type);
extern gpointer g_type_class_peek_parent(gpointer g_class);
extern gpointer g_type_class_peek_static(GType type);
extern gpointer g_type_class_ref(GType type);
extern void g_type_class_unref(gpointer g_class);
extern void g_type_class_unref_uncached(gpointer g_class);
extern GTypeInstance *g_type_create_instance(GType type);
extern gpointer g_type_default_interface_peek(GType g_type);
extern gpointer g_type_default_interface_ref(GType g_type);
extern void g_type_default_interface_unref(gpointer g_iface);
extern guint g_type_depth(GType type);
extern void g_type_free_instance(GTypeInstance * instance);
extern GType g_type_from_name(const gchar * name);
extern GType g_type_fundamental(GType type_id);
extern GType g_type_fundamental_next(void);
extern GTypePlugin *g_type_get_plugin(GType type);
extern gpointer g_type_get_qdata(GType type, GQuark quark);
extern void g_type_init(void);
extern void g_type_init_with_debug_flags(GTypeDebugFlags
debug_flags);
extern gpointer g_type_instance_get_private(GTypeInstance *
instance,
                                           GType private_type);
extern void g_type_interface_add_prerequisite(GType interface_type,
                                              GType prerequisite_type);
extern GTypePlugin *g_type_interface_get_plugin(GType
instance_type,
                                                GType interface_type);
extern gpointer g_type_interface_peek(gpointer instance_class,
                                      GType iface_type);
extern gpointer g_type_interface_peek_parent(gpointer g_iface);
extern GType *g_type_interface_prerequisites(GType interface_type,
                                              guint * n_prerequisites);
extern GType *g_type_interfaces(GType type, guint * n_interfaces);
extern gboolean g_type_is_a(GType type, GType is_a_type);
extern void g_type_module_add_interface(GTypeModule * module,
                                         GType instance_type,
                                         GType interface_type,

```

```

        const GInterfaceInfo *
        interface_info);
extern GType g_type_module_get_type(void);
extern GType g_type_module_register_enum(GTypeModule * module,
        const gchar * name,
        const GEnumValue *
        const_static_values);
extern GType g_type_module_register_flags(GTypeModule * module,
        const gchar * name,
        const GFlagsValue *
        const_static_values);
extern GType g_type_module_register_type(GTypeModule * module,
        GType parent_type,
        const gchar * type_name,
        const GTypeInfo * type_info,
        GTypeFlags flags);
extern void g_type_module_set_name(GTypeModule * module,
        const gchar * name);
extern void g_type_module_unuse(GTypeModule * module);
extern gboolean g_type_module_use(GTypeModule * module);
extern const gchar *g_type_name(GType type);
extern const gchar *g_type_name_from_class(GTypeClass *g_class);
extern const gchar *g_type_name_from_instance(GTypeInstance *
instance);
extern GType g_type_next_base(GType leaf_type, GType root_type);
extern GType g_type_parent(GType type);
extern void g_type_plugin_complete_interface_info(GTypePlugin *
plugin,
        GType instance_type,
        GType interface_type,
        GInterfaceInfo * info);
extern void g_type_plugin_complete_type_info(GTypePlugin * plugin,
        GType g_type,
        GTypeInfo * info,
        GTypeValueTable *
        value_table);
extern GType g_type_plugin_get_type(void);
extern void g_type_plugin_unuse(GTypePlugin * plugin);
extern void g_type_plugin_use(GTypePlugin * plugin);
extern GQuark g_type_qname(GType type);
extern void g_type_query(GType type, GTypeQuery * query);
extern GType g_type_register_dynamic(GType parent_type,
        const gchar * type_name,
        GTypePlugin * plugin,
        GTypeFlags flags);
extern GType g_type_register_fundamental(GType type_id,
        const gchar * type_name,
        const GTypeInfo * info,
        const GTypeFundamentalInfo *
        finfo, GTypeFlags flags);
extern GType g_type_register_static(GType parent_type,
        const gchar * type_name,
        const GTypeInfo * info,
        GTypeFlags flags);
extern GType g_type_register_static_simple(GType, const gchar *,
guint,
        GClassInitFunc, guint,
        GInstanceInitFunc, GTypeFlags);
extern void g_type_remove_class_cache_func(gpointer cache_data,
        GTypeClassCacheFunc
        cache_func);
extern void g_type_remove_interface_check(gpointer check_data,
        GTypeInterfaceCheckFunc
        check_func);
extern void g_type_set_qdata(GType type, GQuark quark, gpointer
data);

```

```

extern gboolean g_type_test_flags(GType type, guint flags);
extern GTypeValueTable *g_type_value_table_peek(GType type);
extern GValueArray *g_value_array_append(GValueArray * value_array,
                                         const GValue * value);
extern GValueArray *g_value_array_copy(const GValueArray *
value_array);
extern void g_value_array_free(GValueArray * value_array);
extern GValue *g_value_array_get_nth(GValueArray * value_array,
                                     guint index_);
extern GType g_value_array_get_type(void);
extern GValueArray *g_value_array_insert(GValueArray * value_array,
                                         guint index_,
                                         const GValue * value);
extern GValueArray *g_value_array_new(guint n_preallocated);
extern GValueArray *g_value_array_prepend(GValueArray *
value_array,
                                         const GValue * value);
extern GValueArray *g_value_array_remove(GValueArray * value_array,
                                         guint index_);
extern GValueArray *g_value_array_sort(GValueArray * value_array,
                                       GCompareFunc compare_func);
extern GValueArray *g_value_array_sort_with_data(GValueArray *
value_array,
                                                GCompareDataFunc
                                                compare_func,
                                                gpointer user_data);
extern void g_value_copy(const GValue * src_value, GValue *
dest_value);
extern gpointer g_value_dup_boxed(const GValue * value);
extern void *g_value_dup_object(const GValue * value);
extern GParamSpec *g_value_dup_param(const GValue * value);
extern gchar *g_value_dup_string(const GValue * value);
extern GVariant *g_value_dup_variant(const GValue * value);
extern gboolean g_value_fits_pointer(const GValue * value);
extern gboolean g_value_get_boolean(const GValue * value);
extern gpointer g_value_get_boxed(const GValue * value);
extern gchar g_value_get_char(const GValue * value);
extern gdouble g_value_get_double(const GValue * value);
extern gint g_value_get_enum(const GValue * value);
extern guint g_value_get_flags(const GValue * value);
extern gfloat g_value_get_float(const GValue * value);
extern GType g_value_get_gtype(const GValue *);
extern gint g_value_get_int(const GValue * value);
extern gint64 g_value_get_int64(const GValue * value);
extern gulong g_value_get_long(const GValue * value);
extern void *g_value_get_object(const GValue * value);
extern GParamSpec *g_value_get_param(const GValue * value);
extern gpointer g_value_get_pointer(const GValue * value);
extern gint8 g_value_get_schar(const GValue * value);
extern const char *g_value_get_string(const GValue * value);
extern GType g_value_get_type(void);
extern gchar g_value_get_uchar(const GValue * value);
extern guint g_value_get_uint(const GValue * value);
extern guint64 g_value_get_uint64(const GValue * value);
extern gulong g_value_get_ulong(const GValue * value);
extern GVariant *g_value_get_variant(const GValue * value);
extern GValue *g_value_init(GValue * value, GType g_type);
extern gpointer g_value_peek_pointer(const GValue * value);
extern void g_value_register_transform_func(GType src_type,
                                           GType dest_type,
                                           GValueTransform
                                           transform_func);
extern GValue *g_value_reset(GValue * value);
extern void g_value_set_boolean(GValue * value, gboolean v_boolean);
extern void g_value_set_boxed(GValue * value, gpointer
v_boxed);

```



```

extern void g_value_set_boxed_take_ownership(GValue * value,
                                             gconstpointer v_boxed);
extern void g_value_set_char(GValue * value, gchar v_char);
extern void g_value_set_double(GValue * value, gdouble v_double);
extern void g_value_set_enum(GValue * value, gint v_enum);
extern void g_value_set_flags(GValue * value, guint v_flags);
extern void g_value_set_float(GValue * value, gfloat v_float);
extern void g_value_set_gtype(GValue *, GType);
extern void g_value_set_instance(GValue * value, gpointer instance);
extern void g_value_set_int(GValue * value, gint v_int);
extern void g_value_set_int64(GValue * value, gint64 v_int64);
extern void g_value_set_long(GValue * value, glong v_long);
extern void g_value_set_object(GValue * value, gpointer v_object);
extern void g_value_set_object_take_ownership(GValue * value,
                                             void *v_object);
extern void g_value_set_param(GValue * value, GParamSpec * param);
extern void g_value_set_param_take_ownership(GValue * value,
                                             GParamSpec * param);
extern void g_value_set_pointer(GValue * value, gpointer v_pointer);
extern void g_value_set_schar(GValue * value, gint8 v_char);
extern void g_value_set_static_boxed(GValue * value,
                                     gconstpointer v_boxed);
extern void g_value_set_static_string(GValue * value,
                                     const gchar * v_string);
extern void g_value_set_string(GValue * value, const gchar *
v_string);
extern void g_value_set_string_take_ownership(GValue * value,
                                             gchar * v_string);
extern void g_value_set_uchar(GValue * value, guchar v_uchar);
extern void g_value_set_uint(GValue * value, guint v_uint);
extern void g_value_set_uint64(GValue * value, guint64 v_uint64);
extern void g_value_set_ulong(GValue * value, gulong v_ulong);
extern void g_value_set_variant(GValue * value, GVariant * variant);
extern void g_value_take_boxed(GValue * value, gconstpointer
v_boxed);
extern void g_value_take_object(GValue * value, gpointer v_object);
extern void g_value_take_param(GValue * value, GParamSpec * param);
extern void g_value_take_string(GValue * value, gchar * v_string);
extern void g_value_take_variant(GValue * value, GVariant *
variant);
extern gboolean g_value_transform(const GValue * src_value,
                                 GValue * dest_value);
extern gboolean g_value_type_compatible(GType src_type, GType
dest_type);
extern gboolean g_value_type_transformable(GType src_type,
                                           GType dest_type);
extern void g_value_unset(GValue * value);
extern GType g_variant_builder_get_type(void);
extern GType g_variant_get_gtype(void);
extern GType g_variant_type_get_gtype(void);
extern void g_weak_ref_clear(struct GWeakRef *weak_ref);
extern void *g_weak_ref_get(struct GWeakRef *weak_ref);
extern void g_weak_ref_init(struct GWeakRef *weak_ref, gpointer
object);
extern void g_weak_ref_set(struct GWeakRef *weak_ref, gpointer
object);

```

17.8.2 glib-2.0/gobject/gvaluecollector.h

```

#define G_VALUE_COLLECT_FORMAT_MAX_LENGTH (8)
#define G_VALUE_LCOPY(value,var_args,flags,__error) \
G_STMT_START { \
    const GValue *_value = (value); \
    guint _flags = (flags); \
    GType _value_type = G_VALUE_TYPE (_value); \

```

```

    GTypeValueTable *_vtable = g_type_value_table_peek (_value_type);
    \
    gchar *_lcopy_format = _vtable->lcopy_format; \
    GTypeCValue      _cvalues[G_VALUE_COLLECT_FORMAT_MAX_LENGTH]      =
    { { 0, }, }; \
    guint _n_values = 0; \
    \
    while (*_lcopy_format) \
    { \
        GTypeCValue *_cvalue = _cvalues + _n_values++; \
        \
        switch (*_lcopy_format++) \
        { \
            case G_VALUE_COLLECT_INT: \
                _cvalue->v_int = va_arg ((var_args), gint); \
                break; \
            \
            case G_VALUE_COLLECT_LONG: \
                _cvalue->v_int = va_arg ((var_args), glong); \
                break; \
            \
            case G_VALUE_COLLECT_INT64: \
                _cvalue->v_int = va_arg ((var_args), gint64); \
                break; \
            \
            case G_VALUE_COLLECT_DOUBLE: \
                _cvalue->v_int = va_arg ((var_args), gdouble); \
                break; \
            \
            case G_VALUE_COLLECT_POINTER: \
                _cvalue->v_int = va_arg ((var_args), gpointer); \
                break; \
            \
            default: \
                g_assert_not_reached (); \
        } \
    } \
    \
    *(_error) = _vtable->lcopy_value (_value, _n_values, _cvalues,
    _flags); \
    \
} G_STMT_END
#define G_VALUE_COLLECT_SKIP(_value_type,var_args) \
G_STMT_START { \
    GTypeValueTable *_vtable = g_type_value_table_peek (_value_type);
    \
    gchar *_collect_format = _vtable->collect_format; \
    \
    while (*_collect_format) \
    { \
        switch (*_collect_format++) \
        { \
            case G_VALUE_COLLECT_INT: \
                va_arg ((var_args), gint); \
                break; \
            \
            case G_VALUE_COLLECT_LONG: \
                va_arg ((var_args), glong); \
                break; \
            \
            case G_VALUE_COLLECT_INT64: \
                va_arg ((var_args), gint64); \
                break; \
            \
            case G_VALUE_COLLECT_DOUBLE: \
                va_arg ((var_args), gdouble); \

```

```

        break; \
    \
    case G_VALUE_COLLECT_POINTER: \
        va_arg ((var_args), gpointer); \
        break; \
    \
    default: \
        g_assert_not_reached (); \
    } \
} \
} G_STMT_END
#define
G_VALUE_COLLECT_INIT(value, _value_type, var_args, flags, __error) \
G_STMT_START { \
    GValue *_val = (value); \
    guint _flags = (flags); \
    GTypeValueTable *_vtab = g_type_value_table_peek (_value_type); \
    \
    gchar *_collect_format = _vtab->collect_format; \
    GTypeCValue _cvalues[G_VALUE_COLLECT_FORMAT_MAX_LENGTH] = \
    { { 0, }, }; \
    guint _n_values = 0; \
    \
    _val->g_type = _value_type; \
    \
    while (*_collect_format) \
    { \
        GTypeCValue *_cvalue = _cvalues + _n_values++; \
        \
        switch (*_collect_format++) \
        { \
            case G_VALUE_COLLECT_INT: \
                _cvalue->v_int = va_arg ((var_args), gint); \
                break; \
            \
            case G_VALUE_COLLECT_LONG: \
                _cvalue->v_int = va_arg ((var_args), glong); \
                break; \
            \
            case G_VALUE_COLLECT_INT64: \
                _cvalue->v_int = va_arg ((var_args), gint64); \
                break; \
            \
            case G_VALUE_COLLECT_DOUBLE: \
                _cvalue->v_int = va_arg ((var_args), gdouble); \
                break; \
            \
            case G_VALUE_COLLECT_POINTER: \
                _cvalue->v_int = va_arg ((var_args), gpointer); \
                break; \
            \
            default: \
                g_assert_not_reached (); \
        } \
    } \
    \
    *(__error) = _vtab->collect_value (_val, _n_values, _cvalues, \
    _flags); \
    \
} G_STMT_END
#define G_VALUE_COLLECT(value, var_args, flags, __error) \
G_STMT_START { \
    GValue *_value = (value); \
    GType _value_type = G_VALUE_TYPE (_value); \
    GTypeValueTable *_vtable = g_type_value_table_peek (_value_type); \
    \

```

```

\
if (_vtable->value_free) \
    _vtable->value_free (_value); \
\
memset (_value->data, 0, sizeof (_value->data)); \
\
G_VALUE_COLLECT_INIT(value, _value_type, var_args, flags,
__error); \
\
} G_STMT_END

enum {
    G_VALUE_COLLECT_INT = 'i',
    G_VALUE_COLLECT_LONG = 'l',
    G_VALUE_COLLECT_INT64 = 'q',
    G_VALUE_COLLECT_DOUBLE = 'd',
    G_VALUE_COLLECT_POINTER = 'p'
};

```

17.9 Interface Definitions for libgobject-2.0

The interfaces defined on the following pages are included in libgobject-2.0 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 17.7 shall behave as described in the referenced base document.

g_type_name_from_class

Name

`g_type_name_from_class` — get unique type name of the class

Synopsis

```

#include <glib-object.h>
gchar *g_type_name_from_class(GTypeClass *g_class);

```

Description

The `g_type_name_from_class()` returns the unique name that is assigned to a type ID of `g_class` class structure.

Return Value

The `g_type_name_from_class()` function returns a pointer referring to the static string with the unique name.

Errors

No errors are defined.

g_type_name_from_instance

Name

g_type_name_from_instance — get name of type from the instance

Synopsis

```
#include <glib-object.h>
gchar *g_type_name_from_instance(GTypeInstance *instance);
```

Description

The g_type_name_from_instance() returns the name of type from the instance *instance* in a Camel case. Instance should be valid instance pointer.

Return Value

The g_type_name_from_instance() function returns a pointer referring to the string with the type name.

Errors

No errors are defined.

g_value_set_instance

Name

g_value_set_instance — set value from an instantiatable type

Synopsis

```
#include <glib-object.h>
void g_value_set_instance(GValue *value, gpointer instance);
```

Description

The g_value_set_instance() sets value from an instantiatable type via the value_table's collect_value() function.

Errors

No errors are defined.

17.10 Interfaces for libgthread-2.0

Table 17-96 defines the library name and shared object name for the libgthread-2.0 library

Table 17-96 libgthread-2.0 Definition

Library:	libgthread-2.0
SONAME:	libgthread-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Glib 2.32] Glib 2.32 Reference Manual

17.10.1 Glib thread abstraction

17.10.1.1 Interfaces for Glib thread abstraction

An LSB conforming implementation shall provide the generic functions for Glib thread abstraction specified in Table 17-97, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-97 libgthread-2.0 - Glib thread abstraction Function Interfaces

g_thread_init [Glib 2.32]	g_thread_init_with_errorcheck_mutes [Glib 2.32]

17.11 Interfaces for libgio-2.0

Table 17-98 defines the library name and shared object name for the libgio-2.0 library

Table 17-98 libgio-2.0 Definition

Library:	libgio-2.0
SONAME:	libgio-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

- [Gio 2.32] Gio 2.32 Reference Manual
- [Gobject 2.32] Gobject 2.32 Reference Manual
- [LSB] This Specification

17.11.1 libgio-2.0 interfaces

17.11.1.1 Interfaces for libgio-2.0 interfaces

An LSB conforming implementation shall provide the generic functions for libgio-2.0 interfaces specified in Table 17-99, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-99 libgio-2.0 - libgio-2.0 interfaces Function Interfaces

g_action_activate [Gio 2.32]	g_action_change_state [Gio 2.32]
g_action_get_enabled [Gio 2.32]	g_action_get_name [Gio 2.32]
g_action_get_parameter_type [Gio 2.32]	g_action_get_state [Gio 2.32]
g_action_get_state_hint [Gio 2.32]	g_action_get_state_type [Gio 2.32]
g_action_get_type [Gobject 2.32]	g_action_group_action_added [Gio 2.32]
g_action_group_action_enabled_changed [Gio 2.32]	g_action_group_action_removed [Gio 2.32]
g_action_group_action_state_changed [Gio 2.32]	g_action_group_activate_action [Gio 2.32]

g_action_group_change_action_state [Gio 2.32]	g_action_group_get_action_enabled [Gio 2.32]
g_action_group_get_action_parameter_type [Gio 2.32]	g_action_group_get_action_state [Gio 2.32]
g_action_group_get_action_state_hint [Gio 2.32]	g_action_group_get_action_state_type [Gio 2.32]
g_action_group_get_type [Gobject 2.32]	g_action_group_has_action [Gio 2.32]
g_action_group_list_actions [Gio 2.32]	g_action_group_query_action [Gio 2.32]
g_action_map_add_action [Gio 2.32]	g_action_map_add_action_entries [Gio 2.32]
g_action_map_get_type [Gobject 2.32]	g_action_map_lookup_action [Gio 2.32]
g_action_map_remove_action [Gio 2.32]	g_app_info_add_supports_type [Gio 2.32]
g_app_info_can_delete [Gio 2.32]	g_app_info_can_remove_supports_type [Gio 2.32]
g_app_info_create_flags_get_type [Gobject 2.32]	g_app_info_create_from_commandline [Gio 2.32]
g_app_info_delete [Gio 2.32]	g_app_info_dup [Gio 2.32]
g_app_info_equal [Gio 2.32]	g_app_info_get_all [Gio 2.32]
g_app_info_get_all_for_type [Gio 2.32]	g_app_info_get_commandline [Gio 2.32]
g_app_info_get_default_for_type [Gio 2.32]	g_app_info_get_default_for_uri_scheme [Gio 2.32]
g_app_info_get_description [Gio 2.32]	g_app_info_get_display_name [Gio 2.32]
g_app_info_get_executable [Gio 2.32]	g_app_info_get_fallback_for_type [Gio 2.32]
g_app_info_get_icon [Gio 2.32]	g_app_info_get_id [Gio 2.32]
g_app_info_get_name [Gio 2.32]	g_app_info_get_recommended_for_type [Gio 2.32]
g_app_info_get_type [Gobject 2.32]	g_app_info_launch [Gio 2.32]
g_app_info_launch_default_for_uri [Gio 2.32]	g_app_info_launch_uris [Gio 2.32]
g_app_info_remove_supports_type [Gio 2.32]	g_app_info_reset_type_associations [Gio 2.32]
g_app_info_set_as_default_for_extension [Gio 2.32]	g_app_info_set_as_default_for_type [Gio 2.32]

g_app_info_set_as_last_used_for_type [Gio 2.32]	g_app_info_should_show [Gio 2.32]
g_app_info_supports_files [Gio 2.32]	g_app_info_supports_uris [Gio 2.32]
g_app_launch_context_get_display [Gio 2.32]	g_app_launch_context_get_environment [Gio 2.32]
g_app_launch_context_get_startup_notify_id [Gio 2.32]	g_app_launch_context_get_type [Gobject 2.32]
g_app_launch_context_launch_failed [Gio 2.32]	g_app_launch_context_new [Gio 2.32]
g_app_launch_context_setenv [Gio 2.32]	g_app_launch_context_unsetenv [Gio 2.32]
g_application_activate [Gio 2.32]	g_application_command_line_get_arguments [Gio 2.32]
g_application_command_line_get_cwd [Gio 2.32]	g_application_command_line_get_environment [Gio 2.32]
g_application_command_line_get_exit_status [Gio 2.32]	g_application_command_line_get_is_remote [Gio 2.32]
g_application_command_line_get_platform_data [Gio 2.32]	g_application_command_line_get_type [Gobject 2.32]
g_application_command_line_getenv [Gio 2.32]	g_application_command_line_print [Gio 2.32]
g_application_command_line_printerr [Gio 2.32]	g_application_command_line_set_exit_status [Gio 2.32]
g_application_flags_get_type [Gobject 2.32]	g_application_get_application_id [Gio 2.32]
g_application_get_default [Gio 2.32]	g_application_get_flags [Gio 2.32]
g_application_get_inactivity_timeout [Gio 2.32]	g_application_get_is_registered [Gio 2.32]
g_application_get_is_remote [Gio 2.32]	g_application_get_type [Gobject 2.32]
g_application_hold [Gio 2.32]	g_application_id_is_valid [Gio 2.32]
g_application_new [Gio 2.32]	g_application_open [Gio 2.32]
g_application_quit [Gio 2.32]	g_application_register [Gio 2.32]
g_application_release [Gio 2.32]	g_application_run [Gio 2.32]
g_application_set_action_group [Gio 2.32]	g_application_set_application_id [Gio 2.32]
g_application_set_default [Gio 2.32]	g_application_set_flags [Gio 2.32]
g_application_set_inactivity_timeout [Gio 2.32]	g_ask_password_flags_get_type [Gobject 2.32]
g_async_initable_get_type [Gobject 2.32]	g_async_initable_init_async [Gio 2.32]

g_async_initable_init_finish [Gio 2.32]	g_async_initable_new_async [Gio 2.32]
g_async_initable_new_finish [Gio 2.32]	g_async_initable_new_valist_async [Gio 2.32]
g_async_initable_newv_async [Gio 2.32]	g_async_result_get_source_object [Gio 2.32]
g_async_result_get_type [Gobject 2.32]	g_async_result_get_user_data [Gio 2.32]
g_buffered_input_stream_fill [Gio 2.32]	g_buffered_input_stream_fill_async [Gio 2.32]
g_buffered_input_stream_fill_finish [Gio 2.32]	g_buffered_input_stream_get_available [Gio 2.32]
g_buffered_input_stream_get_buffer_size [Gio 2.32]	g_buffered_input_stream_get_type [Gobject 2.32]
g_buffered_input_stream_new [Gio 2.32]	g_buffered_input_stream_new_sized [Gio 2.32]
g_buffered_input_stream_peek [Gio 2.32]	g_buffered_input_stream_peek_buffer [Gio 2.32]
g_buffered_input_stream_read_byte [Gio 2.32]	g_buffered_input_stream_set_buffer_size [Gio 2.32]
g_buffered_output_stream_get_auto_grow [Gio 2.32]	g_buffered_output_stream_get_buffer_size [Gio 2.32]
g_buffered_output_stream_get_type [Gobject 2.32]	g_buffered_output_stream_new [Gio 2.32]
g_buffered_output_stream_new_sized [Gio 2.32]	g_buffered_output_stream_set_auto_grow [Gio 2.32]
g_buffered_output_stream_set_buffer_size [Gio 2.32]	g_bus_get [Gio 2.32]
g_bus_get_finish [Gio 2.32]	g_bus_get_sync [Gio 2.32]
g_bus_name_owner_flags_get_type [Gobject 2.32]	g_bus_name_watcher_flags_get_type [Gobject 2.32]
g_bus_own_name [Gio 2.32]	g_bus_own_name_on_connection [Gio 2.32]
g_bus_own_name_on_connection_with_closures [Gio 2.32]	g_bus_own_name_with_closures [Gio 2.32]
g_bus_type_get_type [Gobject 2.32]	g_bus_unown_name [Gio 2.32]
g_bus_unwatch_name [Gio 2.32]	g_bus_watch_name [Gio 2.32]
g_bus_watch_name_on_connection [Gio 2.32]	g_bus_watch_name_on_connection_with_closures [Gio 2.32]
g_bus_watch_name_with_closures [Gio 2.32]	g_cancellable_cancel [Gio 2.32]

g_cancellable_connect [Gio 2.32]	g_cancellable_disconnect [Gio 2.32]
g_cancellable_get_current [Gio 2.32]	g_cancellable_get_fd [Gio 2.32]
g_cancellable_get_type [Gobject 2.32]	g_cancellable_is_cancelled [Gio 2.32]
g_cancellable_make_pollfd [Gio 2.32]	g_cancellable_new [Gio 2.32]
g_cancellable_pop_current [Gio 2.32]	g_cancellable_push_current [Gio 2.32]
g_cancellable_release_fd [Gio 2.32]	g_cancellable_reset [Gio 2.32]
g_cancellable_set_error_if_cancelled [Gio 2.32]	g_cancellable_source_new [Gio 2.32]
g_charset_converter_get_num_fallbacks [Gio 2.32]	g_charset_converter_get_type [Gobject 2.32]
g_charset_converter_get_use_fallback [Gio 2.32]	g_charset_converter_new [Gio 2.32]
g_charset_converter_set_use_fallback [Gio 2.32]	g_content_type_can_be_executable [Gio 2.32]
g_content_type_equals [Gio 2.32]	g_content_type_from_mime_type [Gio 2.32]
g_content_type_get_description [Gio 2.32]	g_content_type_get_icon [Gio 2.32]
g_content_type_get_mime_type [Gio 2.32]	g_content_type_guess [Gio 2.32]
g_content_type_guess_for_tree [Gio 2.32]	g_content_type_is_a [Gio 2.32]
g_content_type_is_unknown [Gio 2.32]	g_content_types_get_registered [Gio 2.32]
g_converter_convert [Gio 2.32]	g_converter_flags_get_type [Gobject 2.32]
g_converter_get_type [Gobject 2.32]	g_converter_input_stream_get_converter [Gio 2.32]
g_converter_input_stream_get_type [Gobject 2.32]	g_converter_input_stream_new [Gio 2.32]
g_converter_output_stream_get_converter [Gio 2.32]	g_converter_output_stream_get_type [Gobject 2.32]
g_converter_output_stream_new [Gio 2.32]	g_converter_reset [Gio 2.32]
g_converter_result_get_type [Gobject 2.32]	g_credentials_get_native [Gio 2.32]
g_credentials_get_type [Gobject 2.32]	g_credentials_get_unix_user [Gio 2.32]
g_credentials_is_same_user [Gio 2.32]	g_credentials_new [Gio 2.32]

g_credentials_set_native [Gio 2.32]	g_credentials_set_unix_user [Gio 2.32]
g_credentials_to_string [Gio 2.32]	g_credentials_type_get_type [Gobject 2.32]
g_data_input_stream_get_byte_order [Gio 2.32]	g_data_input_stream_get_newline_type [Gio 2.32]
g_data_input_stream_get_type [Gobject 2.32]	g_data_input_stream_new [Gio 2.32]
g_data_input_stream_read_byte [Gio 2.32]	g_data_input_stream_read_int16 [Gio 2.32]
g_data_input_stream_read_int32 [Gio 2.32]	g_data_input_stream_read_int64 [Gio 2.32]
g_data_input_stream_read_line [Gio 2.32]	g_data_input_stream_read_line_async [Gio 2.32]
g_data_input_stream_read_line_finish [Gio 2.32]	g_data_input_stream_read_line_finish_utf8 [Gio 2.32]
g_data_input_stream_read_line_utf8 [Gio 2.32]	g_data_input_stream_read_uint16 [Gio 2.32]
g_data_input_stream_read_uint32 [Gio 2.32]	g_data_input_stream_read_uint64 [Gio 2.32]
g_data_input_stream_read_until [Gio 2.32]	g_data_input_stream_read_until_async [Gio 2.32]
g_data_input_stream_read_until_finish [Gio 2.32]	g_data_input_stream_read_upto [Gio 2.32]
g_data_input_stream_read_upto_async [Gio 2.32]	g_data_input_stream_read_upto_finish [Gio 2.32]
g_data_input_stream_set_byte_order [Gio 2.32]	g_data_input_stream_set_newline_type [Gio 2.32]
g_data_output_stream_get_byte_order [Gio 2.32]	g_data_output_stream_get_type [Gobject 2.32]
g_data_output_stream_new [Gio 2.32]	g_data_output_stream_put_byte [Gio 2.32]
g_data_output_stream_put_int16 [Gio 2.32]	g_data_output_stream_put_int32 [Gio 2.32]
g_data_output_stream_put_int64 [Gio 2.32]	g_data_output_stream_put_string [Gio 2.32]
g_data_output_stream_put_uint16 [Gio 2.32]	g_data_output_stream_put_uint32 [Gio 2.32]
g_data_output_stream_put_uint64 [Gio 2.32]	g_data_output_stream_set_byte_order [Gio 2.32]
g_data_stream_byte_order_get_type [Gobject 2.32]	g_data_stream_newline_type_get_type [Gobject 2.32]

g_dbus_action_group_get [Gio 2.32]	g_dbus_action_group_get_type [Gobject 2.32]
g_dbus_address_get_for_bus_sync [Gio 2.32]	g_dbus_address_get_stream [Gio 2.32]
g_dbus_address_get_stream_finish [Gio 2.32]	g_dbus_address_get_stream_sync [Gio 2.32]
g_dbus_annotation_info_get_type [Gobject 2.32]	g_dbus_annotation_info_lookup [Gio 2.32]
g_dbus_annotation_info_ref [Gio 2.32]	g_dbus_annotation_info_unref [Gio 2.32]
g_dbus_arg_info_get_type [Gobject 2.32]	g_dbus_arg_info_ref [Gio 2.32]
g_dbus_arg_info_unref [Gio 2.32]	g_dbus_auth_observer_authorize_authenticated_peer [Gio 2.32]
g_dbus_auth_observer_get_type [Gobject 2.32]	g_dbus_auth_observer_new [Gio 2.32]
g_dbus_call_flags_get_type [Gobject 2.32]	g_dbus_capability_flags_get_type [Gobject 2.32]
g_dbus_connection_add_filter [Gio 2.32]	g_dbus_connection_call [Gio 2.32]
g_dbus_connection_call_finish [Gio 2.32]	g_dbus_connection_call_sync [Gio 2.32]
g_dbus_connection_call_with_unix_fd_list [Gio 2.32]	g_dbus_connection_call_with_unix_fd_list_finish [Gio 2.32]
g_dbus_connection_call_with_unix_fd_list_sync [Gio 2.32]	g_dbus_connection_close [Gio 2.32]
g_dbus_connection_close_finish [Gio 2.32]	g_dbus_connection_close_sync [Gio 2.32]
g_dbus_connection_emit_signal [Gio 2.32]	g_dbus_connection_export_action_group [Gio 2.32]
g_dbus_connection_export_menu_model [Gio 2.32]	g_dbus_connection_flags_get_type [Gobject 2.32]
g_dbus_connection_flush [Gio 2.32]	g_dbus_connection_flush_finish [Gio 2.32]
g_dbus_connection_flush_sync [Gio 2.32]	g_dbus_connection_get_capabilities [Gio 2.32]
g_dbus_connection_get_exit_on_close [Gio 2.32]	g_dbus_connection_get_guid [Gio 2.32]
g_dbus_connection_get_peer_credentials [Gio 2.32]	g_dbus_connection_get_stream [Gio 2.32]
g_dbus_connection_get_type [Gobject 2.32]	g_dbus_connection_get_unique_name [Gio 2.32]

g_dbus_connection_is_closed [Gio 2.32]	g_dbus_connection_new [Gio 2.32]
g_dbus_connection_new_finish [Gio 2.32]	g_dbus_connection_new_for_addresses [Gio 2.32]
g_dbus_connection_new_for_addresses_finish [Gio 2.32]	g_dbus_connection_new_for_addresses_sync [Gio 2.32]
g_dbus_connection_new_sync [Gio 2.32]	g_dbus_connection_register_object [Gio 2.32]
g_dbus_connection_register_subtree [Gio 2.32]	g_dbus_connection_remove_filter [Gio 2.32]
g_dbus_connection_send_message [Gio 2.32]	g_dbus_connection_send_message_with_reply [Gio 2.32]
g_dbus_connection_send_message_with_reply_finish [Gio 2.32]	g_dbus_connection_send_message_with_reply_sync [Gio 2.32]
g_dbus_connection_set_exit_on_close [Gio 2.32]	g_dbus_connection_signal_subscribe [Gio 2.32]
g_dbus_connection_signal_unsubscribe [Gio 2.32]	g_dbus_connection_start_message_processing [Gio 2.32]
g_dbus_connection_unexport_action_group [Gio 2.32]	g_dbus_connection_unexport_menu_model [Gio 2.32]
g_dbus_connection_unregister_object [Gio 2.32]	g_dbus_connection_unregister_subtree [Gio 2.32]
g_dbus_error_encode_gerror [Gio 2.32]	g_dbus_error_get_remote_error [Gio 2.32]
g_dbus_error_get_type [GObject 2.32]	g_dbus_error_is_remote_error [Gio 2.32]
g_dbus_error_new_for_dbus_error [Gio 2.32]	g_dbus_error_quark [LSB]
g_dbus_error_register_error [Gio 2.32]	g_dbus_error_register_error_domain [Gio 2.32]
g_dbus_error_set_dbus_error [Gio 2.32]	g_dbus_error_set_dbus_error_valist [Gio 2.32]
g_dbus_error_strip_remote_error [Gio 2.32]	g_dbus_error_unregister_error [Gio 2.32]
g_dbus_generate_guid [Gio 2.32]	g_dbus_gvalue_to_gvariant [Gio 2.32]
g_dbus_gvariant_to_gvalue [Gio 2.32]	g_dbus_interface_dup_object [Gio 2.32]
g_dbus_interface_get_info [Gio 2.32]	g_dbus_interface_get_object [Gio 2.32]
g_dbus_interface_get_type [GObject 2.32]	g_dbus_interface_info_cache_build [Gio 2.32]

g_dbus_interface_info_cache_release [Gio 2.32]	g_dbus_interface_info_generate_xml [Gio 2.32]
g_dbus_interface_info_get_type [Gobject 2.32]	g_dbus_interface_info_lookup_method [Gio 2.32]
g_dbus_interface_info_lookup_property [Gio 2.32]	g_dbus_interface_info_lookup_signal [Gio 2.32]
g_dbus_interface_info_ref [Gio 2.32]	g_dbus_interface_info_unref [Gio 2.32]
g_dbus_interface_set_object [Gio 2.32]	g_dbus_interface_skeleton_export [Gio 2.32]
g_dbus_interface_skeleton_flags_get_type [Gobject 2.32]	g_dbus_interface_skeleton_flush [Gio 2.32]
g_dbus_interface_skeleton_get_connection [Gio 2.32]	g_dbus_interface_skeleton_get_connections [Gio 2.32]
g_dbus_interface_skeleton_get_flags [Gio 2.32]	g_dbus_interface_skeleton_get_info [Gio 2.32]
g_dbus_interface_skeleton_get_object_path [Gio 2.32]	g_dbus_interface_skeleton_get_properties [Gio 2.32]
g_dbus_interface_skeleton_get_type [Gobject 2.32]	g_dbus_interface_skeleton_get_vtable [Gio 2.32]
g_dbus_interface_skeleton_has_connection [Gio 2.32]	g_dbus_interface_skeleton_set_flags [Gio 2.32]
g_dbus_interface_skeleton_unexport [Gio 2.32]	g_dbus_interface_skeleton_unexport_from_connection [Gio 2.32]
g_dbus_is_address [Gio 2.32]	g_dbus_is_guid [Gio 2.32]
g_dbus_is_interface_name [Gio 2.32]	g_dbus_is_member_name [Gio 2.32]
g_dbus_is_name [Gio 2.32]	g_dbus_is_supported_address [Gio 2.32]
g_dbus_is_unique_name [Gio 2.32]	g_dbus_menu_model_get [Gio 2.32]
g_dbus_menu_model_get_type [Gobject 2.32]	g_dbus_message_byte_order_get_type [Gobject 2.32]
g_dbus_message_bytes_needed [Gio 2.32]	g_dbus_message_copy [Gio 2.32]
g_dbus_message_flags_get_type [Gobject 2.32]	g_dbus_message_get_arg0 [Gio 2.32]
g_dbus_message_get_body [Gio 2.32]	g_dbus_message_get_byte_order [Gio 2.32]
g_dbus_message_get_destination [Gio 2.32]	g_dbus_message_get_error_name [Gio 2.32]
g_dbus_message_get_flags [Gio 2.32]	g_dbus_message_get_header [Gio 2.32]

g_dbus_message_get_header_fields [Gio 2.32]	g_dbus_message_get_interface [Gio 2.32]
g_dbus_message_get_locked [Gio 2.32]	g_dbus_message_get_member [Gio 2.32]
g_dbus_message_get_message_type [Gio 2.32]	g_dbus_message_get_num_unix_fds [Gio 2.32]
g_dbus_message_get_path [Gio 2.32]	g_dbus_message_get_reply_serial [Gio 2.32]
g_dbus_message_get_sender [Gio 2.32]	g_dbus_message_get_serial [Gio 2.32]
g_dbus_message_get_signature [Gio 2.32]	g_dbus_message_get_type [Gobject 2.32]
g_dbus_message_get_unix_fd_list [Gio 2.32]	g_dbus_message_header_field_get_type [Gobject 2.32]
g_dbus_message_lock [Gio 2.32]	g_dbus_message_new [Gio 2.32]
g_dbus_message_new_from_blob [Gio 2.32]	g_dbus_message_new_method_call [Gio 2.32]
g_dbus_message_new_method_error [Gio 2.32]	g_dbus_message_new_method_error_literal [Gio 2.32]
g_dbus_message_new_method_error_valist [Gio 2.32]	g_dbus_message_new_method_reply [Gio 2.32]
g_dbus_message_new_signal [Gio 2.32]	g_dbus_message_print [Gio 2.32]
g_dbus_message_set_body [Gio 2.32]	g_dbus_message_set_byte_order [Gio 2.32]
g_dbus_message_set_destination [Gio 2.32]	g_dbus_message_set_error_name [Gio 2.32]
g_dbus_message_set_flags [Gio 2.32]	g_dbus_message_set_header [Gio 2.32]
g_dbus_message_set_interface [Gio 2.32]	g_dbus_message_set_member [Gio 2.32]
g_dbus_message_set_message_type [Gio 2.32]	g_dbus_message_set_num_unix_fds [Gio 2.32]
g_dbus_message_set_path [Gio 2.32]	g_dbus_message_set_reply_serial [Gio 2.32]
g_dbus_message_set_sender [Gio 2.32]	g_dbus_message_set_serial [Gio 2.32]
g_dbus_message_set_signature [Gio 2.32]	g_dbus_message_set_unix_fd_list [Gio 2.32]
g_dbus_message_to_blob [Gio 2.32]	g_dbus_message_to_gerror [Gio 2.32]
g_dbus_message_type_get_type [Gobject 2.32]	g_dbus_method_info_get_type [Gobject 2.32]

g_dbus_method_info_ref [Gio 2.32]	g_dbus_method_info_unref [Gio 2.32]
g_dbus_method_invocation_get_connection [Gio 2.32]	g_dbus_method_invocation_get_interface_name [Gio 2.32]
g_dbus_method_invocation_get_message [Gio 2.32]	g_dbus_method_invocation_get_method_info [Gio 2.32]
g_dbus_method_invocation_get_method_name [Gio 2.32]	g_dbus_method_invocation_get_object_path [Gio 2.32]
g_dbus_method_invocation_get_parameters [Gio 2.32]	g_dbus_method_invocation_get_sender [Gio 2.32]
g_dbus_method_invocation_get_type [Gobject 2.32]	g_dbus_method_invocation_get_userdata [Gio 2.32]
g_dbus_method_invocation_return_dbus_error [Gio 2.32]	g_dbus_method_invocation_return_error [Gio 2.32]
g_dbus_method_invocation_return_error_literal [Gio 2.32]	g_dbus_method_invocation_return_error_valist [Gio 2.32]
g_dbus_method_invocation_return_gerror [Gio 2.32]	g_dbus_method_invocation_return_value [Gio 2.32]
g_dbus_method_invocation_return_value_with_unix_fd_list [Gio 2.32]	g_dbus_method_invocation_take_error [Gio 2.32]
g_dbus_node_info_generate_xml [Gio 2.32]	g_dbus_node_info_get_type [Gobject 2.32]
g_dbus_node_info_lookup_interface [Gio 2.32]	g_dbus_node_info_new_for_xml [Gio 2.32]
g_dbus_node_info_ref [Gio 2.32]	g_dbus_node_info_unref [Gio 2.32]
g_dbus_object_get_interface [Gio 2.32]	g_dbus_object_get_interfaces [Gio 2.32]
g_dbus_object_get_object_path [Gio 2.32]	g_dbus_object_get_type [Gobject 2.32]
g_dbus_object_manager_client_flags_get_type [Gobject 2.32]	g_dbus_object_manager_client_get_connection [Gio 2.32]
g_dbus_object_manager_client_get_flags [Gio 2.32]	g_dbus_object_manager_client_get_name [Gio 2.32]
g_dbus_object_manager_client_get_name_owner [Gio 2.32]	g_dbus_object_manager_client_get_type [Gobject 2.32]
g_dbus_object_manager_client_new [Gio 2.32]	g_dbus_object_manager_client_new_finish [Gio 2.32]
g_dbus_object_manager_client_new_for_bus [Gio 2.32]	g_dbus_object_manager_client_new_for_bus_finish [Gio 2.32]
g_dbus_object_manager_client_new_for_bus_sync [Gio 2.32]	g_dbus_object_manager_client_new_sync [Gio 2.32]

g_dbus_object_manager_get_interface [Gio 2.32]	g_dbus_object_manager_get_object [Gio 2.32]
g_dbus_object_manager_get_object_path [Gio 2.32]	g_dbus_object_manager_get_objects [Gio 2.32]
g_dbus_object_manager_get_type [Gobject 2.32]	g_dbus_object_manager_server_export [Gio 2.32]
g_dbus_object_manager_server_export_uniquely [Gio 2.32]	g_dbus_object_manager_server_get_connection [Gio 2.32]
g_dbus_object_manager_server_get_type [Gobject 2.32]	g_dbus_object_manager_server_new [Gio 2.32]
g_dbus_object_manager_server_set_connection [Gio 2.32]	g_dbus_object_manager_server_unexport [Gio 2.32]
g_dbus_object_proxy_get_connection [Gio 2.32]	g_dbus_object_proxy_get_type [Gobject 2.32]
g_dbus_object_proxy_new [Gio 2.32]	g_dbus_object_skeleton_add_interface [Gio 2.32]
g_dbus_object_skeleton_flush [Gio 2.32]	g_dbus_object_skeleton_get_type [Gobject 2.32]
g_dbus_object_skeleton_new [Gio 2.32]	g_dbus_object_skeleton_remove_interface [Gio 2.32]
g_dbus_object_skeleton_remove_interface_by_name [Gio 2.32]	g_dbus_object_skeleton_set_object_path [Gio 2.32]
g_dbus_property_info_flags_get_type [Gobject 2.32]	g_dbus_property_info_get_type [Gobject 2.32]
g_dbus_property_info_ref [Gio 2.32]	g_dbus_property_info_unref [Gio 2.32]
g_dbus_proxy_call [Gio 2.32]	g_dbus_proxy_call_finish [Gio 2.32]
g_dbus_proxy_call_sync [Gio 2.32]	g_dbus_proxy_call_with_unix_fd_list [Gio 2.32]
g_dbus_proxy_call_with_unix_fd_list_finish [Gio 2.32]	g_dbus_proxy_call_with_unix_fd_list_sync [Gio 2.32]
g_dbus_proxy_flags_get_type [Gobject 2.32]	g_dbus_proxy_get_cached_property [Gio 2.32]
g_dbus_proxy_get_cached_property_names [Gio 2.32]	g_dbus_proxy_get_connection [Gio 2.32]
g_dbus_proxy_get_default_timeout [Gio 2.32]	g_dbus_proxy_get_flags [Gio 2.32]
g_dbus_proxy_get_interface_info [Gio 2.32]	g_dbus_proxy_get_interface_name [Gio 2.32]
g_dbus_proxy_get_name [Gio 2.32]	g_dbus_proxy_get_name_owner [Gio 2.32]

g_dbus_proxy_get_object_path [Gio 2.32]	g_dbus_proxy_get_type [Gobject 2.32]
g_dbus_proxy_new [Gio 2.32]	g_dbus_proxy_new_finish [Gio 2.32]
g_dbus_proxy_new_for_bus [Gio 2.32]	g_dbus_proxy_new_for_bus_finish [Gio 2.32]
g_dbus_proxy_new_for_bus_sync [Gio 2.32]	g_dbus_proxy_new_sync [Gio 2.32]
g_dbus_proxy_set_cached_property [Gio 2.32]	g_dbus_proxy_set_default_timeout [Gio 2.32]
g_dbus_proxy_set_interface_info [Gio 2.32]	g_dbus_send_message_flags_get_type [Gobject 2.32]
g_dbus_server_flags_get_type [Gobject 2.32]	g_dbus_server_get_client_address [Gio 2.32]
g_dbus_server_get_flags [Gio 2.32]	g_dbus_server_get_guid [Gio 2.32]
g_dbus_server_get_type [Gobject 2.32]	g_dbus_server_is_active [Gio 2.32]
g_dbus_server_new_sync [Gio 2.32]	g_dbus_server_start [Gio 2.32]
g_dbus_server_stop [Gio 2.32]	g_dbus_signal_flags_get_type [Gobject 2.32]
g_dbus_signal_info_get_type [Gobject 2.32]	g_dbus_signal_info_ref [Gio 2.32]
g_dbus_signal_info_unref [Gio 2.32]	g_dbus_subtree_flags_get_type [Gobject 2.32]
g_desktop_app_info_get_categories [Gio 2.32]	g_desktop_app_info_get_filename [Gio 2.32]
g_desktop_app_info_get_generic_name [Gio 2.32]	g_desktop_app_info_get_is_hidden [Gio 2.32]
g_desktop_app_info_get_keywords [Gio 2.32]	g_desktop_app_info_get_nodisplay [Gio 2.32]
g_desktop_app_info_get_show_in [Gio 2.32]	g_desktop_app_info_get_type [Gobject 2.32]
g_desktop_app_info_launch_uris_as_manager [Gio 2.32]	g_desktop_app_info_lookup_get_default_for_uri_scheme [LSB]
g_desktop_app_info_lookup_get_type [Gobject 2.32]	g_desktop_app_info_new [Gio 2.32]
g_desktop_app_info_new_from_filename [Gio 2.32]	g_desktop_app_info_new_from_keyfile [Gio 2.32]
g_desktop_app_info_set_desktop_env [Gio 2.32]	g_drive_can_eject [Gio 2.32]
g_drive_can_poll_for_media [Gio 2.32]	g_drive_can_start [Gio 2.32]

g_drive_can_start_degraded [Gio 2.32]	g_drive_can_stop [Gio 2.32]
g_drive_eject [Gio 2.32]	g_drive_eject_finish [Gio 2.32]
g_drive_eject_with_operation [Gio 2.32]	g_drive_eject_with_operation_finish [Gio 2.32]
g_drive_enumerate_identifiers [Gio 2.32]	g_drive_get_icon [Gio 2.32]
g_drive_get_identifier [Gio 2.32]	g_drive_get_name [Gio 2.32]
g_drive_get_sort_key [Gio 2.32]	g_drive_get_start_stop_type [Gio 2.32]
g_drive_get_type [Gobject 2.32]	g_drive_get_volumes [Gio 2.32]
g_drive_has_media [Gio 2.32]	g_drive_has_volumes [Gio 2.32]
g_drive_is_media_check_automatic [Gio 2.32]	g_drive_is_media_removable [Gio 2.32]
g_drive_poll_for_media [Gio 2.32]	g_drive_poll_for_media_finish [Gio 2.32]
g_drive_start [Gio 2.32]	g_drive_start_finish [Gio 2.32]
g_drive_start_flags_get_type [Gobject 2.32]	g_drive_start_stop_type_get_type [Gobject 2.32]
g_drive_stop [Gio 2.32]	g_drive_stop_finish [Gio 2.32]
g_emblem_get_icon [Gio 2.32]	g_emblem_get_origin [Gio 2.32]
g_emblem_get_type [Gobject 2.32]	g_emblem_new [Gio 2.32]
g_emblem_new_with_origin [Gio 2.32]	g_emblem_origin_get_type [Gobject 2.32]
g_emblemed_icon_add_emblem [Gio 2.32]	g_emblemed_icon_clear_emblems [Gio 2.32]
g_emblemed_icon_get_emblems [Gio 2.32]	g_emblemed_icon_get_icon [Gio 2.32]
g_emblemed_icon_get_type [Gobject 2.32]	g_emblemed_icon_new [Gio 2.32]
g_file_append_to [Gio 2.32]	g_file_append_to_async [Gio 2.32]
g_file_append_to_finish [Gio 2.32]	g_file_attribute_info_flags_get_type [Gobject 2.32]
g_file_attribute_info_list_add [Gio 2.32]	g_file_attribute_info_list_dup [Gio 2.32]
g_file_attribute_info_list_get_type [Gobject 2.32]	g_file_attribute_info_list_lookup [Gio 2.32]
g_file_attribute_info_list_new [Gio 2.32]	g_file_attribute_info_list_ref [Gio 2.32]

<code>g_file_attribute_info_list_unref</code> [Gio 2.32]	<code>g_file_attribute_matcher_enumerate_namespace</code> [Gio 2.32]
<code>g_file_attribute_matcher_enumerate_next</code> [Gio 2.32]	<code>g_file_attribute_matcher_get_type</code> [Gobject 2.32]
<code>g_file_attribute_matcher_matches</code> [Gio 2.32]	<code>g_file_attribute_matcher_matches_only</code> [Gio 2.32]
<code>g_file_attribute_matcher_new</code> [Gio 2.32]	<code>g_file_attribute_matcher_ref</code> [Gio 2.32]
<code>g_file_attribute_matcher_subtract</code> [Gio 2.32]	<code>g_file_attribute_matcher_to_string</code> [Gio 2.32]
<code>g_file_attribute_matcher_unref</code> [Gio 2.32]	<code>g_file_attribute_status_get_type</code> [Gobject 2.32]
<code>g_file_attribute_type_get_type</code> [Gobject 2.32]	<code>g_file_copy</code> [Gio 2.32]
<code>g_file_copy_async</code> [Gio 2.32]	<code>g_file_copy_attributes</code> [Gio 2.32]
<code>g_file_copy_finish</code> [Gio 2.32]	<code>g_file_copy_flags_get_type</code> [Gobject 2.32]
<code>g_file_create</code> [Gio 2.32]	<code>g_file_create_async</code> [Gio 2.32]
<code>g_file_create_finish</code> [Gio 2.32]	<code>g_file_create_flags_get_type</code> [Gobject 2.32]
<code>g_file_create_readwrite</code> [Gio 2.32]	<code>g_file_create_readwrite_async</code> [Gio 2.32]
<code>g_file_create_readwrite_finish</code> [Gio 2.32]	<code>g_file_delete</code> [Gio 2.32]
<code>g_file_descriptor_based_get_fd</code> [Gio 2.32]	<code>g_file_descriptor_based_get_type</code> [Gobject 2.32]
<code>g_file_dup</code> [Gio 2.32]	<code>g_file_eject_mountable</code> [Gio 2.32]
<code>g_file_eject_mountable_finish</code> [Gio 2.32]	<code>g_file_eject_mountable_with_operation</code> [Gio 2.32]
<code>g_file_eject_mountable_with_operation_finish</code> [Gio 2.32]	<code>g_file_enumerate_children</code> [Gio 2.32]
<code>g_file_enumerate_children_async</code> [Gio 2.32]	<code>g_file_enumerate_children_finish</code> [Gio 2.32]
<code>g_file_enumerator_close</code> [Gio 2.32]	<code>g_file_enumerator_close_async</code> [Gio 2.32]
<code>g_file_enumerator_close_finish</code> [Gio 2.32]	<code>g_file_enumerator_get_container</code> [Gio 2.32]
<code>g_file_enumerator_get_type</code> [Gobject 2.32]	<code>g_file_enumerator_has_pending</code> [Gio 2.32]
<code>g_file_enumerator_is_closed</code> [Gio 2.32]	<code>g_file_enumerator_next_file</code> [Gio 2.32]

<code>g_file_enumerator_next_files_async</code> [Gio 2.32]	<code>g_file_enumerator_next_files_finish</code> [Gio 2.32]
<code>g_file_enumerator_set_pending</code> [Gio 2.32]	<code>g_file_equal</code> [Gio 2.32]
<code>g_file_find_enclosing_mount</code> [Gio 2.32]	<code>g_file_find_enclosing_mount_async</code> [Gio 2.32]
<code>g_file_find_enclosing_mount_finish</code> [Gio 2.32]	<code>g_file_get_basename</code> [Gio 2.32]
<code>g_file_get_child</code> [Gio 2.32]	<code>g_file_get_child_for_display_name</code> [Gio 2.32]
<code>g_file_get_parent</code> [Gio 2.32]	<code>g_file_get_parse_name</code> [Gio 2.32]
<code>g_file_get_path</code> [Gio 2.32]	<code>g_file_get_relative_path</code> [Gio 2.32]
<code>g_file_get_type</code> [Gobject 2.32]	<code>g_file_get_uri</code> [Gio 2.32]
<code>g_file_get_uri_scheme</code> [Gio 2.32]	<code>g_file_has_parent</code> [Gio 2.32]
<code>g_file_has_prefix</code> [Gio 2.32]	<code>g_file_has_uri_scheme</code> [Gio 2.32]
<code>g_file_hash</code> [Gio 2.32]	<code>g_file_icon_get_file</code> [Gio 2.32]
<code>g_file_icon_get_type</code> [Gobject 2.32]	<code>g_file_icon_new</code> [Gio 2.32]
<code>g_file_info_clear_status</code> [Gio 2.32]	<code>g_file_info_copy_into</code> [Gio 2.32]
<code>g_file_info_dup</code> [Gio 2.32]	<code>g_file_info_get_attribute_as_string</code> [Gio 2.32]
<code>g_file_info_get_attribute_boolean</code> [Gio 2.32]	<code>g_file_info_get_attribute_byte_string</code> [Gio 2.32]
<code>g_file_info_get_attribute_data</code> [Gio 2.32]	<code>g_file_info_get_attribute_int32</code> [Gio 2.32]
<code>g_file_info_get_attribute_int64</code> [Gio 2.32]	<code>g_file_info_get_attribute_object</code> [Gio 2.32]
<code>g_file_info_get_attribute_status</code> [Gio 2.32]	<code>g_file_info_get_attribute_string</code> [Gio 2.32]
<code>g_file_info_get_attribute_stringv</code> [Gio 2.32]	<code>g_file_info_get_attribute_type</code> [Gio 2.32]
<code>g_file_info_get_attribute_uint32</code> [Gio 2.32]	<code>g_file_info_get_attribute_uint64</code> [Gio 2.32]
<code>g_file_info_get_content_type</code> [Gio 2.32]	<code>g_file_info_get_display_name</code> [Gio 2.32]
<code>g_file_info_get_edit_name</code> [Gio 2.32]	<code>g_file_info_get_etag</code> [Gio 2.32]
<code>g_file_info_get_file_type</code> [Gio 2.32]	<code>g_file_info_get_icon</code> [Gio 2.32]
<code>g_file_info_get_is_backup</code> [Gio 2.32]	<code>g_file_info_get_is_hidden</code> [Gio 2.32]
<code>g_file_info_get_is_symlink</code> [Gio 2.32]	<code>g_file_info_get_modification_time</code> [Gio 2.32]

<code>g_file_info_get_name</code> [Gio 2.32]	<code>g_file_info_get_size</code> [Gio 2.32]
<code>g_file_info_get_sort_order</code> [Gio 2.32]	<code>g_file_info_get_symlink_target</code> [Gio 2.32]
<code>g_file_info_get_type</code> [Gobject 2.32]	<code>g_file_info_has_attribute</code> [Gio 2.32]
<code>g_file_info_has_namespace</code> [Gio 2.32]	<code>g_file_info_list_attributes</code> [Gio 2.32]
<code>g_file_info_new</code> [Gio 2.32]	<code>g_file_info_remove_attribute</code> [Gio 2.32]
<code>g_file_info_set_attribute</code> [Gio 2.32]	<code>g_file_info_set_attribute_boolean</code> [Gio 2.32]
<code>g_file_info_set_attribute_byte_string</code> [Gio 2.32]	<code>g_file_info_set_attribute_int32</code> [Gio 2.32]
<code>g_file_info_set_attribute_int64</code> [Gio 2.32]	<code>g_file_info_set_attribute_mask</code> [Gio 2.32]
<code>g_file_info_set_attribute_object</code> [Gio 2.32]	<code>g_file_info_set_attribute_status</code> [Gio 2.32]
<code>g_file_info_set_attribute_string</code> [Gio 2.32]	<code>g_file_info_set_attribute_stringv</code> [Gio 2.32]
<code>g_file_info_set_attribute_uint32</code> [Gio 2.32]	<code>g_file_info_set_attribute_uint64</code> [Gio 2.32]
<code>g_file_info_set_content_type</code> [Gio 2.32]	<code>g_file_info_set_display_name</code> [Gio 2.32]
<code>g_file_info_set_edit_name</code> [Gio 2.32]	<code>g_file_info_set_file_type</code> [Gio 2.32]
<code>g_file_info_set_icon</code> [Gio 2.32]	<code>g_file_info_set_is_hidden</code> [Gio 2.32]
<code>g_file_info_set_is_symlink</code> [Gio 2.32]	<code>g_file_info_set_modification_time</code> [Gio 2.32]
<code>g_file_info_set_name</code> [Gio 2.32]	<code>g_file_info_set_size</code> [Gio 2.32]
<code>g_file_info_set_sort_order</code> [Gio 2.32]	<code>g_file_info_set_symlink_target</code> [Gio 2.32]
<code>g_file_info_unset_attribute_mask</code> [Gio 2.32]	<code>g_file_input_stream_get_type</code> [Gobject 2.32]
<code>g_file_input_stream_query_info</code> [Gio 2.32]	<code>g_file_input_stream_query_info_async</code> [Gio 2.32]
<code>g_file_input_stream_query_info_finish</code> [Gio 2.32]	<code>g_file_io_stream_get_etag</code> [Gio 2.32]
<code>g_file_io_stream_get_type</code> [Gobject 2.32]	<code>g_file_io_stream_query_info</code> [Gio 2.32]
<code>g_file_io_stream_query_info_async</code> [Gio 2.32]	<code>g_file_io_stream_query_info_finish</code> [Gio 2.32]
<code>g_file_is_native</code> [Gio 2.32]	<code>g_file_load_contents</code> [Gio 2.32]
<code>g_file_load_contents_async</code> [Gio 2.32]	<code>g_file_load_contents_finish</code> [Gio 2.32]

g_file_load_partial_contents_async [Gio 2.32]	g_file_load_partial_contents_finish [Gio 2.32]
g_file_make_directory [Gio 2.32]	g_file_make_directory_with_parents [Gio 2.32]
g_file_make_symbolic_link [Gio 2.32]	g_file_monitor [Gio 2.32]
g_file_monitor_cancel [Gio 2.32]	g_file_monitor_directory [Gio 2.32]
g_file_monitor_emit_event [Gio 2.32]	g_file_monitor_event_get_type [Gobject 2.32]
g_file_monitor_file [Gio 2.32]	g_file_monitor_flags_get_type [Gobject 2.32]
g_file_monitor_get_type [Gobject 2.32]	g_file_monitor_is_cancelled [Gio 2.32]
g_file_monitor_set_rate_limit [Gio 2.32]	g_file_mount_enclosing_volume [Gio 2.32]
g_file_mount_enclosing_volume_finish [Gio 2.32]	g_file_mount_mountable [Gio 2.32]
g_file_mount_mountable_finish [Gio 2.32]	g_file_move [Gio 2.32]
g_file_new_for_commandline_arg [Gio 2.32]	g_file_new_for_path [Gio 2.32]
g_file_new_for_uri [Gio 2.32]	g_file_new_tmp [Gio 2.32]
g_file_open_readwrite [Gio 2.32]	g_file_open_readwrite_async [Gio 2.32]
g_file_open_readwrite_finish [Gio 2.32]	g_file_output_stream_get_etag [Gio 2.32]
g_file_output_stream_get_type [Gobject 2.32]	g_file_output_stream_query_info [Gio 2.32]
g_file_output_stream_query_info_async [Gio 2.32]	g_file_output_stream_query_info_finish [Gio 2.32]
g_file_parse_name [Gio 2.32]	g_file_poll_mountable [Gio 2.32]
g_file_poll_mountable_finish [Gio 2.32]	g_file_query_default_handler [Gio 2.32]
g_file_query_exists [Gio 2.32]	g_file_query_file_type [Gio 2.32]
g_file_query_filesystem_info [Gio 2.32]	g_file_query_filesystem_info_async [Gio 2.32]
g_file_query_filesystem_info_finish [Gio 2.32]	g_file_query_info [Gio 2.32]
g_file_query_info_async [Gio 2.32]	g_file_query_info_finish [Gio 2.32]
g_file_query_info_flags_get_type [Gobject 2.32]	g_file_query_settable_attributes [Gio 2.32]

g_file_query_writable_namespaces [Gio 2.32]	g_file_read [Gio 2.32]
g_file_read_async [Gio 2.32]	g_file_read_finish [Gio 2.32]
g_file_replace [Gio 2.32]	g_file_replace_async [Gio 2.32]
g_file_replace_contents [Gio 2.32]	g_file_replace_contents_async [Gio 2.32]
g_file_replace_contents_finish [Gio 2.32]	g_file_replace_finish [Gio 2.32]
g_file_replace_readwrite [Gio 2.32]	g_file_replace_readwrite_async [Gio 2.32]
g_file_replace_readwrite_finish [Gio 2.32]	g_file_resolve_relative_path [Gio 2.32]
g_file_set_attribute [Gio 2.32]	g_file_set_attribute_byte_string [Gio 2.32]
g_file_set_attribute_int32 [Gio 2.32]	g_file_set_attribute_int64 [Gio 2.32]
g_file_set_attribute_string [Gio 2.32]	g_file_set_attribute_uint32 [Gio 2.32]
g_file_set_attribute_uint64 [Gio 2.32]	g_file_set_attributes_async [Gio 2.32]
g_file_set_attributes_finish [Gio 2.32]	g_file_set_attributes_from_info [Gio 2.32]
g_file_set_display_name [Gio 2.32]	g_file_set_display_name_async [Gio 2.32]
g_file_set_display_name_finish [Gio 2.32]	g_file_start_mountable [Gio 2.32]
g_file_start_mountable_finish [Gio 2.32]	g_file_stop_mountable [Gio 2.32]
g_file_stop_mountable_finish [Gio 2.32]	g_file_supports_thread_contexts [Gio 2.32]
g_file_trash [Gio 2.32]	g_file_type_get_type [Gobject 2.32]
g_file_unmount_mountable [Gio 2.32]	g_file_unmount_mountable_finish [Gio 2.32]
g_file_unmount_mountable_with_operation [Gio 2.32]	g_file_unmount_mountable_with_operation_finish [Gio 2.32]
g_filename_completer_get_completion_suffix [Gio 2.32]	g_filename_completer_get_completions [Gio 2.32]
g_filename_completer_get_type [Gobject 2.32]	g_filename_completer_new [Gio 2.32]
g_filename_completer_set_dirs_only [Gio 2.32]	g_filesystem_preview_type_get_type [Gobject 2.32]
g_filter_input_stream_get_base_stream [Gio 2.32]	g_filter_input_stream_get_close_base_stream [Gio 2.32]

<code>g_filter_input_stream_get_type</code> [Gobject 2.32]	<code>g_filter_input_stream_set_close_base_stream</code> [Gio 2.32]
<code>g_filter_output_stream_get_base_stream</code> [Gio 2.32]	<code>g_filter_output_stream_get_close_base_stream</code> [Gio 2.32]
<code>g_filter_output_stream_get_type</code> [Gobject 2.32]	<code>g_filter_output_stream_set_close_base_stream</code> [Gio 2.32]
<code>g_icon_equal</code> [Gio 2.32]	<code>g_icon_get_type</code> [Gobject 2.32]
<code>g_icon_hash</code> [Gio 2.32]	<code>g_icon_new_for_string</code> [Gio 2.32]
<code>g_icon_to_string</code> [Gio 2.32]	<code>g_inet_address_equal</code> [Gio 2.32]
<code>g_inet_address_get_family</code> [Gio 2.32]	<code>g_inet_address_get_is_any</code> [Gio 2.32]
<code>g_inet_address_get_is_link_local</code> [Gio 2.32]	<code>g_inet_address_get_is_loopback</code> [Gio 2.32]
<code>g_inet_address_get_is_mc_global</code> [Gio 2.32]	<code>g_inet_address_get_is_mc_link_local</code> [Gio 2.32]
<code>g_inet_address_get_is_mc_node_local</code> [Gio 2.32]	<code>g_inet_address_get_is_mc_org_local</code> [Gio 2.32]
<code>g_inet_address_get_is_mc_site_local</code> [Gio 2.32]	<code>g_inet_address_get_is_multicast</code> [Gio 2.32]
<code>g_inet_address_get_is_site_local</code> [Gio 2.32]	<code>g_inet_address_get_native_size</code> [Gio 2.32]
<code>g_inet_address_get_type</code> [Gobject 2.32]	<code>g_inet_address_mask_equal</code> [Gio 2.32]
<code>g_inet_address_mask_get_address</code> [Gio 2.32]	<code>g_inet_address_mask_get_family</code> [Gio 2.32]
<code>g_inet_address_mask_get_length</code> [Gio 2.32]	<code>g_inet_address_mask_get_type</code> [Gobject 2.32]
<code>g_inet_address_mask_matches</code> [Gio 2.32]	<code>g_inet_address_mask_new</code> [Gio 2.32]
<code>g_inet_address_mask_new_from_string</code> [Gio 2.32]	<code>g_inet_address_mask_to_string</code> [Gio 2.32]
<code>g_inet_address_new_any</code> [Gio 2.32]	<code>g_inet_address_new_from_bytes</code> [Gio 2.32]
<code>g_inet_address_new_from_string</code> [Gio 2.32]	<code>g_inet_address_new_loopback</code> [Gio 2.32]
<code>g_inet_address_to_bytes</code> [Gio 2.32]	<code>g_inet_address_to_string</code> [Gio 2.32]
<code>g_inet_socket_address_get_address</code> [Gio 2.32]	<code>g_inet_socket_address_get_flowinfo</code> [Gio 2.32]
<code>g_inet_socket_address_get_port</code> [Gio 2.32]	<code>g_inet_socket_address_get_scope_id</code> [Gio 2.32]
<code>g_inet_socket_address_get_type</code> [Gobject 2.32]	<code>g_inet_socket_address_new</code> [Gio 2.32]

<code>g_initable_get_type</code> [Gobject 2.32]	<code>g_initable_init</code> [Gio 2.32]
<code>g_initable_new</code> [Gio 2.32]	<code>g_initable_new_valist</code> [Gio 2.32]
<code>g_initable_newv</code> [Gio 2.32]	<code>g_input_stream_clear_pending</code> [Gio 2.32]
<code>g_input_stream_close</code> [Gio 2.32]	<code>g_input_stream_close_async</code> [Gio 2.32]
<code>g_input_stream_close_finish</code> [Gio 2.32]	<code>g_input_stream_get_type</code> [Gobject 2.32]
<code>g_input_stream_has_pending</code> [Gio 2.32]	<code>g_input_stream_is_closed</code> [Gio 2.32]
<code>g_input_stream_read</code> [Gio 2.32]	<code>g_input_stream_read_all</code> [Gio 2.32]
<code>g_input_stream_read_async</code> [Gio 2.32]	<code>g_input_stream_read_finish</code> [Gio 2.32]
<code>g_input_stream_set_pending</code> [Gio 2.32]	<code>g_input_stream_skip</code> [Gio 2.32]
<code>g_input_stream_skip_async</code> [Gio 2.32]	<code>g_input_stream_skip_finish</code> [Gio 2.32]
<code>g_io_error_enum_get_type</code> [Gobject 2.32]	<code>g_io_error_from_errno</code> [Gio 2.32]
<code>g_io_error_quark</code> [LSB]	<code>g_io_extension_get_name</code> [Gio 2.32]
<code>g_io_extension_get_priority</code> [Gio 2.32]	<code>g_io_extension_get_type</code> [Gobject 2.32]
<code>g_io_extension_point_get_extension_by_name</code> [Gio 2.32]	<code>g_io_extension_point_get_extensions</code> [Gio 2.32]
<code>g_io_extension_point_get_required_type</code> [Gio 2.32]	<code>g_io_extension_point_implement</code> [Gio 2.32]
<code>g_io_extension_point_lookup</code> [Gio 2.32]	<code>g_io_extension_point_register</code> [Gio 2.32]
<code>g_io_extension_point_set_required_type</code> [Gio 2.32]	<code>g_io_extension_ref_class</code> [Gio 2.32]
<code>g_io_module_get_type</code> [Gobject 2.32]	<code>g_io_module_new</code> [Gio 2.32]
<code>g_io_module_scope_block</code> [Gio 2.32]	<code>g_io_module_scope_flags_get_type</code> [Gobject 2.32]
<code>g_io_module_scope_free</code> [Gio 2.32]	<code>g_io_module_scope_new</code> [Gio 2.32]
<code>g_io_modules_load_all_in_directory</code> [Gio 2.32]	<code>g_io_modules_load_all_in_directory_with_scope</code> [Gio 2.32]
<code>g_io_modules_scan_all_in_directory</code> [Gio 2.32]	<code>g_io_modules_scan_all_in_directory_with_scope</code> [Gio 2.32]
<code>g_io_scheduler_cancel_all_jobs</code> [Gio 2.32]	<code>g_io_scheduler_job_send_to_mainloop</code> [Gio 2.32]

g_io_scheduler_job_send_to_mainloop_async [Gio 2.32]	g_io_scheduler_push_job [Gio 2.32]
g_io_stream_clear_pending [Gio 2.32]	g_io_stream_close [Gio 2.32]
g_io_stream_close_async [Gio 2.32]	g_io_stream_close_finish [Gio 2.32]
g_io_stream_get_input_stream [Gio 2.32]	g_io_stream_get_output_stream [Gio 2.32]
g_io_stream_get_type [Gobject 2.32]	g_io_stream_has_pending [Gio 2.32]
g_io_stream_is_closed [Gio 2.32]	g_io_stream_set_pending [Gio 2.32]
g_io_stream_splice_async [Gio 2.32]	g_io_stream_splice_finish [Gio 2.32]
g_io_stream_splice_flags_get_type [Gobject 2.32]	g_keyfile_settings_backend_new [Gio 2.32]
g_loadable_icon_get_type [Gobject 2.32]	g_loadable_icon_load [Gio 2.32]
g_loadable_icon_load_async [Gio 2.32]	g_loadable_icon_load_finish [Gio 2.32]
g_memory_input_stream_add_data [Gio 2.32]	g_memory_input_stream_get_type [Gobject 2.32]
g_memory_input_stream_new [Gio 2.32]	g_memory_input_stream_new_from_data [Gio 2.32]
g_memory_output_stream_get_data [Gio 2.32]	g_memory_output_stream_get_data_size [Gio 2.32]
g_memory_output_stream_get_size [Gio 2.32]	g_memory_output_stream_get_type [Gobject 2.32]
g_memory_output_stream_new [Gio 2.32]	g_memory_output_stream_steal_data [Gio 2.32]
g_memory_settings_backend_new [Gio 2.32]	g_menu_append [Gio 2.32]
g_menu_append_item [Gio 2.32]	g_menu_append_section [Gio 2.32]
g_menu_append_submenu [Gio 2.32]	g_menu_attribute_iter_get_name [Gio 2.32]
g_menu_attribute_iter_get_next [Gio 2.32]	g_menu_attribute_iter_get_type [Gobject 2.32]
g_menu_attribute_iter_get_value [Gio 2.32]	g_menu_attribute_iter_next [Gio 2.32]
g_menu_freeze [Gio 2.32]	g_menu_get_type [Gobject 2.32]
g_menu_insert [Gio 2.32]	g_menu_insert_item [Gio 2.32]
g_menu_insert_section [Gio 2.32]	g_menu_insert_submenu [Gio 2.32]
g_menu_item_get_type [Gobject 2.32]	g_menu_item_new [Gio 2.32]

g_menu_item_new_section [Gio 2.32]	g_menu_item_new_submenu [Gio 2.32]
g_menu_item_set_action_and_target [Gio 2.32]	g_menu_item_set_action_and_target_value [Gio 2.32]
g_menu_item_set_attribute [Gio 2.32]	g_menu_item_set_attribute_value [Gio 2.32]
g_menu_item_set_detailed_action [Gio 2.32]	g_menu_item_set_label [Gio 2.32]
g_menu_item_set_link [Gio 2.32]	g_menu_item_set_section [Gio 2.32]
g_menu_item_set_submenu [Gio 2.32]	g_menu_link_iter_get_name [Gio 2.32]
g_menu_link_iter_get_next [Gio 2.32]	g_menu_link_iter_get_type [Gobject 2.32]
g_menu_link_iter_get_value [Gio 2.32]	g_menu_link_iter_next [Gio 2.32]
g_menu_model_get_item_attribute [Gio 2.32]	g_menu_model_get_item_attribute_value [Gio 2.32]
g_menu_model_get_item_link [Gio 2.32]	g_menu_model_get_n_items [Gio 2.32]
g_menu_model_get_type [Gobject 2.32]	g_menu_model_is_mutable [Gio 2.32]
g_menu_model_items_changed [Gio 2.32]	g_menu_model_iterate_item_attributes [Gio 2.32]
g_menu_model_iterate_item_links [Gio 2.32]	g_menu_new [Gio 2.32]
g_menu_prepend [Gio 2.32]	g_menu_prepend_item [Gio 2.32]
g_menu_prepend_section [Gio 2.32]	g_menu_prepend_submenu [Gio 2.32]
g_menu_remove [Gio 2.32]	g_mount_can_eject [Gio 2.32]
g_mount_can_unmount [Gio 2.32]	g_mount_eject [Gio 2.32]
g_mount_eject_finish [Gio 2.32]	g_mount_eject_with_operation [Gio 2.32]
g_mount_eject_with_operation_finish [Gio 2.32]	g_mount_get_default_location [Gio 2.32]
g_mount_get_drive [Gio 2.32]	g_mount_get_icon [Gio 2.32]
g_mount_get_name [Gio 2.32]	g_mount_get_root [Gio 2.32]
g_mount_get_sort_key [Gio 2.32]	g_mount_get_type [Gobject 2.32]
g_mount_get_uuid [Gio 2.32]	g_mount_get_volume [Gio 2.32]
g_mount_guess_content_type [Gio 2.32]	g_mount_guess_content_type_finish [Gio 2.32]

g_mount_guess_content_type_sync [Gio 2.32]	g_mount_is_shadowed [Gio 2.32]
g_mount_mount_flags_get_type [Gobject 2.32]	g_mount_operation_get_anonymous [Gio 2.32]
g_mount_operation_get_choice [Gio 2.32]	g_mount_operation_get_domain [Gio 2.32]
g_mount_operation_get_password [Gio 2.32]	g_mount_operation_get_password_save [Gio 2.32]
g_mount_operation_get_type [Gobject 2.32]	g_mount_operation_get_username [Gio 2.32]
g_mount_operation_new [Gio 2.32]	g_mount_operation_reply [Gio 2.32]
g_mount_operation_result_get_type [Gobject 2.32]	g_mount_operation_set_anonymous [Gio 2.32]
g_mount_operation_set_choice [Gio 2.32]	g_mount_operation_set_domain [Gio 2.32]
g_mount_operation_set_password [Gio 2.32]	g_mount_operation_set_password_save [Gio 2.32]
g_mount_operation_set_username [Gio 2.32]	g_mount_remount [Gio 2.32]
g_mount_remount_finish [Gio 2.32]	g_mount_shadow [Gio 2.32]
g_mount_unmount [Gio 2.32]	g_mount_unmount_finish [Gio 2.32]
g_mount_unmount_flags_get_type [Gobject 2.32]	g_mount_unmount_with_operation [Gio 2.32]
g_mount_unmount_with_operation_finish [Gio 2.32]	g_mount_unshadow [Gio 2.32]
g_native_volume_monitor_get_type [Gobject 2.32]	g_network_address_get_hostname [Gio 2.32]
g_network_address_get_port [Gio 2.32]	g_network_address_get_scheme [Gio 2.32]
g_network_address_get_type [Gobject 2.32]	g_network_address_new [Gio 2.32]
g_network_address_parse [Gio 2.32]	g_network_address_parse_uri [Gio 2.32]
g_network_monitor_can_reach [Gio 2.32]	g_network_monitor_can_reach_async [Gio 2.32]
g_network_monitor_can_reach_finish [Gio 2.32]	g_network_monitor_get_default [Gio 2.32]
g_network_monitor_get_network_available [Gio 2.32]	g_network_monitor_get_type [Gobject 2.32]
g_network_service_get_domain [Gio 2.32]	g_network_service_get_protocol [Gio 2.32]

g_network_service_get_scheme [Gio 2.32]	g_network_service_get_service [Gio 2.32]
g_network_service_get_type [Gobject 2.32]	g_network_service_new [Gio 2.32]
g_network_service_set_scheme [Gio 2.32]	g_null_settings_backend_new [Gio 2.32]
g_output_stream_clear_pending [Gio 2.32]	g_output_stream_close [Gio 2.32]
g_output_stream_close_async [Gio 2.32]	g_output_stream_close_finish [Gio 2.32]
g_output_stream_flush [Gio 2.32]	g_output_stream_flush_async [Gio 2.32]
g_output_stream_flush_finish [Gio 2.32]	g_output_stream_get_type [Gobject 2.32]
g_output_stream_has_pending [Gio 2.32]	g_output_stream_is_closed [Gio 2.32]
g_output_stream_is_closing [Gio 2.32]	g_output_stream_set_pending [Gio 2.32]
g_output_stream_splice [Gio 2.32]	g_output_stream_splice_async [Gio 2.32]
g_output_stream_splice_finish [Gio 2.32]	g_output_stream_splice_flags_get_type [Gobject 2.32]
g_output_stream_write [Gio 2.32]	g_output_stream_write_all [Gio 2.32]
g_output_stream_write_async [Gio 2.32]	g_output_stream_write_finish [Gio 2.32]
g_password_save_get_type [Gobject 2.32]	g_permission_acquire [Gio 2.32]
g_permission_acquire_async [Gio 2.32]	g_permission_acquire_finish [Gio 2.32]
g_permission_get_allowed [Gio 2.32]	g_permission_get_can_acquire [Gio 2.32]
g_permission_get_can_release [Gio 2.32]	g_permission_get_type [Gobject 2.32]
g_permission_impl_update [Gio 2.32]	g_permission_release [Gio 2.32]
g_permission_release_async [Gio 2.32]	g_permission_release_finish [Gio 2.32]
g_pollable_input_stream_can_poll [Gio 2.32]	g_pollable_input_stream_create_source [Gio 2.32]
g_pollable_input_stream_get_type [Gobject 2.32]	g_pollable_input_stream_is_readable [Gio 2.32]

g_pollable_input_stream_read_nonblocking [Gio 2.32]	g_pollable_output_stream_can_poll [Gio 2.32]
g_pollable_output_stream_create_source [Gio 2.32]	g_pollable_output_stream_get_type [Gobject 2.32]
g_pollable_output_stream_is_writable [Gio 2.32]	g_pollable_output_stream_write_nonblocking [Gio 2.32]
g_pollable_source_new [Gio 2.32]	g_proxy_address_enumerator_get_type [Gobject 2.32]
g_proxy_address_get_destination_hostname [Gio 2.32]	g_proxy_address_get_destination_port [Gio 2.32]
g_proxy_address_get_password [Gio 2.32]	g_proxy_address_get_protocol [Gio 2.32]
g_proxy_address_get_type [Gobject 2.32]	g_proxy_address_get_username [Gio 2.32]
g_proxy_address_new [Gio 2.32]	g_proxy_connect [Gio 2.32]
g_proxy_connect_async [Gio 2.32]	g_proxy_connect_finish [Gio 2.32]
g_proxy_get_default_for_protocol [Gio 2.32]	g_proxy_get_type [Gobject 2.32]
g_proxy_resolver_get_default [Gio 2.32]	g_proxy_resolver_get_type [Gobject 2.32]
g_proxy_resolver_is_supported [Gio 2.32]	g_proxy_resolver_lookup [Gio 2.32]
g_proxy_resolver_lookup_async [Gio 2.32]	g_proxy_resolver_lookup_finish [Gio 2.32]
g_proxy_supports_hostname [Gio 2.32]	g_remote_action_group_activate_action_full [Gio 2.32]
g_remote_action_group_change_action_state_full [Gio 2.32]	g_remote_action_group_get_type [Gobject 2.32]
g_resolver_error_get_type [Gobject 2.32]	g_resolver_error_quark [LSB]
g_resolver_free_addresses [Gio 2.32]	g_resolver_free_targets [Gio 2.32]
g_resolver_get_default [Gio 2.32]	g_resolver_get_type [Gobject 2.32]
g_resolver_lookup_by_address [Gio 2.32]	g_resolver_lookup_by_address_async [Gio 2.32]
g_resolver_lookup_by_address_finish [Gio 2.32]	g_resolver_lookup_by_name [Gio 2.32]
g_resolver_lookup_by_name_async [Gio 2.32]	g_resolver_lookup_by_name_finish [Gio 2.32]
g_resolver_lookup_service [Gio 2.32]	g_resolver_lookup_service_async [Gio 2.32]

g_resolver_lookup_service_finish [Gio 2.32]	g_resolver_set_default [Gio 2.32]
g_resource_enumerate_children [Gio 2.32]	g_resource_error_get_type [Gobject 2.32]
g_resource_error_quark [LSB]	g_resource_flags_get_type [Gobject 2.32]
g_resource_get_info [Gio 2.32]	g_resource_get_type [Gobject 2.32]
g_resource_load [Gio 2.32]	g_resource_lookup_data [Gio 2.32]
g_resource_lookup_flags_get_type [Gobject 2.32]	g_resource_new_from_data [Gio 2.32]
g_resource_open_stream [Gio 2.32]	g_resource_ref [Gio 2.32]
g_resource_unref [Gio 2.32]	g_resources_enumerate_children [Gio 2.32]
g_resources_get_info [Gio 2.32]	g_resources_lookup_data [Gio 2.32]
g_resources_open_stream [Gio 2.32]	g_resources_register [Gio 2.32]
g_resources_unregister [Gio 2.32]	g_seekable_can_seek [Gio 2.32]
g_seekable_can_truncate [Gio 2.32]	g_seekable_get_type [Gobject 2.32]
g_seekable_seek [Gio 2.32]	g_seekable_tell [Gio 2.32]
g_seekable_truncate [Gio 2.32]	g_settings_apply [Gio 2.32]
g_settings_backend_changed [Gio 2.32]	g_settings_backend_changed_tree [Gio 2.32]
g_settings_backend_flatten_tree [Gio 2.32]	g_settings_backend_get_default [Gio 2.32]
g_settings_backend_get_type [Gobject 2.32]	g_settings_backend_keys_changed [Gio 2.32]
g_settings_backend_path_changed [Gio 2.32]	g_settings_backend_path_writable_changed [Gio 2.32]
g_settings_backend_writable_changed [Gio 2.32]	g_settings_bind [Gio 2.32]
g_settings_bind_flags_get_type [Gobject 2.32]	g_settings_bind_with_mapping [Gio 2.32]
g_settings_bind_writable [Gio 2.32]	g_settings_create_action [Gio 2.32]
g_settings_delay [Gio 2.32]	g_settings_get [Gio 2.32]
g_settings_get_boolean [Gio 2.32]	g_settings_get_child [Gio 2.32]
g_settings_get_double [Gio 2.32]	g_settings_get_enum [Gio 2.32]
g_settings_get_flags [Gio 2.32]	g_settings_get_has_unapplied [Gio 2.32]
g_settings_get_int [Gio 2.32]	g_settings_get_mapped [Gio 2.32]
g_settings_get_range [Gio 2.32]	g_settings_get_string [Gio 2.32]

g_settings_get_strv [Gio 2.32]	g_settings_get_type [Gobject 2.32]
g_settings_get_uint [Gio 2.32]	g_settings_get_value [Gio 2.32]
g_settings_is_writable [Gio 2.32]	g_settings_list_children [Gio 2.32]
g_settings_list_keys [Gio 2.32]	g_settings_list_relocatable_schemas [Gio 2.32]
g_settings_list_schemas [Gio 2.32]	g_settings_new [Gio 2.32]
g_settings_new_full [Gio 2.32]	g_settings_new_with_backend [Gio 2.32]
g_settings_new_with_backend_and_path [Gio 2.32]	g_settings_new_with_path [Gio 2.32]
g_settings_range_check [Gio 2.32]	g_settings_reset [Gio 2.32]
g_settings_revert [Gio 2.32]	g_settings_schema_get_id [Gio 2.32]
g_settings_schema_get_path [Gio 2.32]	g_settings_schema_get_type [Gobject 2.32]
g_settings_schema_ref [Gio 2.32]	g_settings_schema_source_get_default [Gio 2.32]
g_settings_schema_source_get_type [Gobject 2.32]	g_settings_schema_source_lookup [Gio 2.32]
g_settings_schema_source_new_from_directory [Gio 2.32]	g_settings_schema_source_ref [Gio 2.32]
g_settings_schema_source_unref [Gio 2.32]	g_settings_schema_unref [Gio 2.32]
g_settings_set [Gio 2.32]	g_settings_set_boolean [Gio 2.32]
g_settings_set_double [Gio 2.32]	g_settings_set_enum [Gio 2.32]
g_settings_set_flags [Gio 2.32]	g_settings_set_int [Gio 2.32]
g_settings_set_string [Gio 2.32]	g_settings_set_strv [Gio 2.32]
g_settings_set_uint [Gio 2.32]	g_settings_set_value [Gio 2.32]
g_settings_sync [Gio 2.32]	g_settings_unbind [Gio 2.32]
g_simple_action_get_type [Gobject 2.32]	g_simple_action_group_add_entries [Gio 2.32]
g_simple_action_group_get_type [Gobject 2.32]	g_simple_action_group_insert [Gio 2.32]
g_simple_action_group_lookup [Gio 2.32]	g_simple_action_group_new [Gio 2.32]
g_simple_action_group_remove [Gio 2.32]	g_simple_action_new [Gio 2.32]
g_simple_action_new_stateful [Gio 2.32]	g_simple_action_set_enabled [Gio 2.32]

g_simple_action_set_state [Gio 2.32]	g_simple_async_report_error_in_idle [Gio 2.32]
g_simple_async_report_gerror_in_idle [Gio 2.32]	g_simple_async_report_take_gerror_in_idle [Gio 2.32]
g_simple_async_result_complete [Gio 2.32]	g_simple_async_result_complete_in_idle [Gio 2.32]
g_simple_async_result_get_op_res_gboolean [Gio 2.32]	g_simple_async_result_get_op_res_gpointer [Gio 2.32]
g_simple_async_result_get_op_res_gssize [Gio 2.32]	g_simple_async_result_get_source_tag [Gio 2.32]
g_simple_async_result_get_type [Gobject 2.32]	g_simple_async_result_is_valid [Gio 2.32]
g_simple_async_result_new [Gio 2.32]	g_simple_async_result_new_error [Gio 2.32]
g_simple_async_result_new_from_error [Gio 2.32]	g_simple_async_result_new_take_error [Gio 2.32]
g_simple_async_result_propagate_error [Gio 2.32]	g_simple_async_result_run_in_thread [Gio 2.32]
g_simple_async_result_set_check_cancellable [Gio 2.32]	g_simple_async_result_set_error [Gio 2.32]
g_simple_async_result_set_error_value [Gio 2.32]	g_simple_async_result_set_from_error [Gio 2.32]
g_simple_async_result_set_handle_cancellation [Gio 2.32]	g_simple_async_result_set_op_res_gboolean [Gio 2.32]
g_simple_async_result_set_op_res_gpointer [Gio 2.32]	g_simple_async_result_set_op_res_gssize [Gio 2.32]
g_simple_async_result_take_error [Gio 2.32]	g_simple_permission_get_type [Gobject 2.32]
g_simple_permission_new [Gio 2.32]	g_socket_accept [Gio 2.32]
g_socket_address_enumerator_get_type [Gobject 2.32]	g_socket_address_enumerator_next [Gio 2.32]
g_socket_address_enumerator_next_async [Gio 2.32]	g_socket_address_enumerator_next_finish [Gio 2.32]
g_socket_address_get_family [Gio 2.32]	g_socket_address_get_native_size [Gio 2.32]
g_socket_address_get_type [Gobject 2.32]	g_socket_address_new_from_native [Gio 2.32]
g_socket_address_to_native [Gio 2.32]	g_socket_bind [Gio 2.32]
g_socket_check_connect_result [Gio 2.32]	g_socket_client_add_application_proxy [Gio 2.32]

<code>g_socket_client_connect</code> [Gio 2.32]	<code>g_socket_client_connect_async</code> [Gio 2.32]
<code>g_socket_client_connect_finish</code> [Gio 2.32]	<code>g_socket_client_connect_to_host</code> [Gio 2.32]
<code>g_socket_client_connect_to_host_async</code> [Gio 2.32]	<code>g_socket_client_connect_to_host_finish</code> [Gio 2.32]
<code>g_socket_client_connect_to_service</code> [Gio 2.32]	<code>g_socket_client_connect_to_service_async</code> [Gio 2.32]
<code>g_socket_client_connect_to_service_finish</code> [Gio 2.32]	<code>g_socket_client_connect_to_uri</code> [Gio 2.32]
<code>g_socket_client_connect_to_uri_async</code> [Gio 2.32]	<code>g_socket_client_connect_to_uri_finish</code> [Gio 2.32]
<code>g_socket_client_event_get_type</code> [Gobject 2.32]	<code>g_socket_client_get_enable_proxy</code> [Gio 2.32]
<code>g_socket_client_get_family</code> [Gio 2.32]	<code>g_socket_client_get_local_address</code> [Gio 2.32]
<code>g_socket_client_get_protocol</code> [Gio 2.32]	<code>g_socket_client_get_socket_type</code> [Gio 2.32]
<code>g_socket_client_get_timeout</code> [Gio 2.32]	<code>g_socket_client_get_tls</code> [Gio 2.32]
<code>g_socket_client_get_tls_validation_flags</code> [Gio 2.32]	<code>g_socket_client_get_type</code> [Gobject 2.32]
<code>g_socket_client_new</code> [Gio 2.32]	<code>g_socket_client_set_enable_proxy</code> [Gio 2.32]
<code>g_socket_client_set_family</code> [Gio 2.32]	<code>g_socket_client_set_local_address</code> [Gio 2.32]
<code>g_socket_client_set_protocol</code> [Gio 2.32]	<code>g_socket_client_set_socket_type</code> [Gio 2.32]
<code>g_socket_client_set_timeout</code> [Gio 2.32]	<code>g_socket_client_set_tls</code> [Gio 2.32]
<code>g_socket_client_set_tls_validation_flags</code> [Gio 2.32]	<code>g_socket_close</code> [Gio 2.32]
<code>g_socket_condition_check</code> [Gio 2.32]	<code>g_socket_condition_timed_wait</code> [Gio 2.32]
<code>g_socket_condition_wait</code> [Gio 2.32]	<code>g_socket_connect</code> [Gio 2.32]
<code>g_socket_connectable_enumerate</code> [Gio 2.32]	<code>g_socket_connectable_get_type</code> [Gobject 2.32]
<code>g_socket_connectable_proxy_enumerate</code> [Gio 2.32]	<code>g_socket_connection_connect</code> [Gio 2.32]
<code>g_socket_connection_connect_async</code> [Gio 2.32]	<code>g_socket_connection_connect_finish</code> [Gio 2.32]

g_socket_connection_factory_create_connection [Gio 2.32]	g_socket_connection_factory_lookup_type [Gio 2.32]
g_socket_connection_factory_register_type [Gio 2.32]	g_socket_connection_get_local_address [Gio 2.32]
g_socket_connection_get_remote_address [Gio 2.32]	g_socket_connection_get_socket [Gio 2.32]
g_socket_connection_get_type [Gobject 2.32]	g_socket_connection_is_connected [Gio 2.32]
g_socket_control_message_deserialize [Gio 2.32]	g_socket_control_message_get_level [Gio 2.32]
g_socket_control_message_get_msg_type [Gio 2.32]	g_socket_control_message_get_size [Gio 2.32]
g_socket_control_message_get_type [Gobject 2.32]	g_socket_control_message_serialize [Gio 2.32]
g_socket_create_source [Gio 2.32]	g_socket_family_get_type [Gobject 2.32]
g_socket_get_available_bytes [Gio 2.32]	g_socket_get_blocking [Gio 2.32]
g_socket_get_broadcast [Gio 2.32]	g_socket_get_credentials [Gio 2.32]
g_socket_get_family [Gio 2.32]	g_socket_get_fd [Gio 2.32]
g_socket_get_keepalive [Gio 2.32]	g_socket_get_listen_backlog [Gio 2.32]
g_socket_get_local_address [Gio 2.32]	g_socket_get_multicast_loopback [Gio 2.32]
g_socket_get_multicast_ttl [Gio 2.32]	g_socket_get_protocol [Gio 2.32]
g_socket_get_remote_address [Gio 2.32]	g_socket_get_socket_type [Gio 2.32]
g_socket_get_timeout [Gio 2.32]	g_socket_get_ttl [Gio 2.32]
g_socket_get_type [Gobject 2.32]	g_socket_is_closed [Gio 2.32]
g_socket_is_connected [Gio 2.32]	g_socket_join_multicast_group [Gio 2.32]
g_socket_leave_multicast_group [Gio 2.32]	g_socket_listen [Gio 2.32]
g_socket_listener_accept [Gio 2.32]	g_socket_listener_accept_async [Gio 2.32]
g_socket_listener_accept_finish [Gio 2.32]	g_socket_listener_accept_socket [Gio 2.32]
g_socket_listener_accept_socket_async [Gio 2.32]	g_socket_listener_accept_socket_finish [Gio 2.32]
g_socket_listener_add_address [Gio 2.32]	g_socket_listener_add_any_inet_port [Gio 2.32]

g_socket_listener_add_inet_port [Gio 2.32]	g_socket_listener_add_socket [Gio 2.32]
g_socket_listener_close [Gio 2.32]	g_socket_listener_get_type [Gobject 2.32]
g_socket_listener_new [Gio 2.32]	g_socket_listener_set_backlog [Gio 2.32]
g_socket_msg_flags_get_type [Gobject 2.32]	g_socket_new [Gio 2.32]
g_socket_new_from_fd [Gio 2.32]	g_socket_protocol_get_type [Gobject 2.32]
g_socket_receive [Gio 2.32]	g_socket_receive_from [Gio 2.32]
g_socket_receive_message [Gio 2.32]	g_socket_receive_with_blocking [Gio 2.32]
g_socket_send [Gio 2.32]	g_socket_send_message [Gio 2.32]
g_socket_send_to [Gio 2.32]	g_socket_send_with_blocking [Gio 2.32]
g_socket_service_get_type [Gobject 2.32]	g_socket_service_is_active [Gio 2.32]
g_socket_service_new [Gio 2.32]	g_socket_service_start [Gio 2.32]
g_socket_service_stop [Gio 2.32]	g_socket_set_blocking [Gio 2.32]
g_socket_set_broadcast [Gio 2.32]	g_socket_set_keepalive [Gio 2.32]
g_socket_set_listen_backlog [Gio 2.32]	g_socket_set_multicast_loopback [Gio 2.32]
g_socket_set_multicast_ttl [Gio 2.32]	g_socket_set_timeout [Gio 2.32]
g_socket_set_ttl [Gio 2.32]	g_socket_shutdown [Gio 2.32]
g_socket_speaks_ipv4 [Gio 2.32]	g_socket_type_get_type [Gobject 2.32]
g_srv_target_copy [Gio 2.32]	g_srv_target_free [Gio 2.32]
g_srv_target_get_hostname [Gio 2.32]	g_srv_target_get_port [Gio 2.32]
g_srv_target_get_priority [Gio 2.32]	g_srv_target_get_type [Gobject 2.32]
g_srv_target_get_weight [Gio 2.32]	g_srv_target_list_sort [Gio 2.32]
g_srv_target_new [Gio 2.32]	g_static_resource_fini [LSB]
g_static_resource_get_resource [LSB]	g_static_resource_init [LSB]
g_tcp_connection_get_graceful_disconnect [Gio 2.32]	g_tcp_connection_get_type [Gobject 2.32]
g_tcp_connection_set_graceful_disconnect [Gio 2.32]	g_tcp_wrapper_connection_get_base_io_stream [Gio 2.32]

g_tcp_wrapper_connection_get_type [Gobject 2.32]	g_tcp_wrapper_connection_new [Gio 2.32]
g_themed_icon_append_name [Gio 2.32]	g_themed_icon_get_names [Gio 2.32]
g_themed_icon_get_type [Gobject 2.32]	g_themed_icon_new [Gio 2.32]
g_themed_icon_new_from_names [Gio 2.32]	g_themed_icon_new_with_default_fallbacks [Gio 2.32]
g_themed_icon_prepend_name [Gio 2.32]	g_threaded_socket_service_get_type [Gobject 2.32]
g_threaded_socket_service_new [Gio 2.32]	g_tls_authentication_mode_get_type [Gobject 2.32]
g_tls_backend_get_certificate_type [Gio 2.32]	g_tls_backend_get_client_connection_type [Gio 2.32]
g_tls_backend_get_default [Gio 2.32]	g_tls_backend_get_default_database [Gio 2.32]
g_tls_backend_get_file_database_type [Gio 2.32]	g_tls_backend_get_server_connection_type [Gio 2.32]
g_tls_backend_get_type [Gobject 2.32]	g_tls_backend_supports_tls [Gio 2.32]
g_tls_certificate_flags_get_type [Gobject 2.32]	g_tls_certificate_get_issuer [Gio 2.32]
g_tls_certificate_get_type [Gobject 2.32]	g_tls_certificate_list_new_from_file [Gio 2.32]
g_tls_certificate_new_from_file [Gio 2.32]	g_tls_certificate_new_from_files [Gio 2.32]
g_tls_certificate_new_from_pem [Gio 2.32]	g_tls_certificate_verify [Gio 2.32]
g_tls_client_connection_get_accepted_cas [Gio 2.32]	g_tls_client_connection_get_server_identity [Gio 2.32]
g_tls_client_connection_get_type [Gobject 2.32]	g_tls_client_connection_get_use_ssl3 [Gio 2.32]
g_tls_client_connection_get_validation_flags [Gio 2.32]	g_tls_client_connection_new [Gio 2.32]
g_tls_client_connection_set_server_identity [Gio 2.32]	g_tls_client_connection_set_use_ssl3 [Gio 2.32]
g_tls_client_connection_set_validation_flags [Gio 2.32]	g_tls_connection_emit_accept_certificate [Gio 2.32]
g_tls_connection_get_certificate [Gio 2.32]	g_tls_connection_get_database [Gio 2.32]
g_tls_connection_get_interaction [Gio 2.32]	g_tls_connection_get_peer_certificate [Gio 2.32]

g_tls_connection_get_peer_certificate_errors [Gio 2.32]	g_tls_connection_get_rehandshake_mode [Gio 2.32]
g_tls_connection_get_require_close_notify [Gio 2.32]	g_tls_connection_get_type [Gobject 2.32]
g_tls_connection_get_use_system_certificate [Gio 2.32]	g_tls_connection_handshake [Gio 2.32]
g_tls_connection_handshake_async [Gio 2.32]	g_tls_connection_handshake_finish [Gio 2.32]
g_tls_connection_set_certificate [Gio 2.32]	g_tls_connection_set_database [Gio 2.32]
g_tls_connection_set_interaction [Gio 2.32]	g_tls_connection_set_rehandshake_mode [Gio 2.32]
g_tls_connection_set_require_close_notify [Gio 2.32]	g_tls_connection_set_use_system_certificate [Gio 2.32]
g_tls_database_create_certificate_handle [Gio 2.32]	g_tls_database_get_type [Gobject 2.32]
g_tls_database_lookup_certificate_for_handle [Gio 2.32]	g_tls_database_lookup_certificate_for_handle_async [Gio 2.32]
g_tls_database_lookup_certificate_for_handle_finish [Gio 2.32]	g_tls_database_lookup_certificate_issuer [Gio 2.32]
g_tls_database_lookup_certificate_issuer_async [Gio 2.32]	g_tls_database_lookup_certificate_issuer_finish [Gio 2.32]
g_tls_database_lookup_certificates_issued_by [Gio 2.32]	g_tls_database_lookup_certificates_issued_by_async [Gio 2.32]
g_tls_database_lookup_certificates_issued_by_finish [Gio 2.32]	g_tls_database_lookup_flags_get_type [Gobject 2.32]
g_tls_database_verify_chain [Gio 2.32]	g_tls_database_verify_chain_async [Gio 2.32]
g_tls_database_verify_chain_finish [Gio 2.32]	g_tls_database_verify_flags_get_type [Gobject 2.32]
g_tls_error_get_type [Gobject 2.32]	g_tls_error_quark [LSB]
g_tls_file_database_get_type [Gobject 2.32]	g_tls_file_database_new [Gio 2.32]
g_tls_interaction_ask_password [Gio 2.32]	g_tls_interaction_ask_password_async [Gio 2.32]
g_tls_interaction_ask_password_finish [Gio 2.32]	g_tls_interaction_get_type [Gobject 2.32]
g_tls_interaction_invoke_ask_password [Gio 2.32]	g_tls_interaction_result_get_type [Gobject 2.32]
g_tls_password_flags_get_type [Gobject 2.32]	g_tls_password_get_description [Gio 2.32]

g_tls_password_get_flags [Gio 2.32]	g_tls_password_get_type [Gobject 2.32]
g_tls_password_get_value [Gio 2.32]	g_tls_password_get_warning [Gio 2.32]
g_tls_password_new [Gio 2.32]	g_tls_password_set_description [Gio 2.32]
g_tls_password_set_flags [Gio 2.32]	g_tls_password_set_value [Gio 2.32]
g_tls_password_set_value_full [Gio 2.32]	g_tls_password_set_warning [Gio 2.32]
g_tls_rehandshake_mode_get_type [Gobject 2.32]	g_tls_server_connection_get_type [Gobject 2.32]
g_tls_server_connection_new [Gio 2.32]	g_unix_connection_get_type [Gobject 2.32]
g_unix_connection_receive_credentials [Gio 2.32]	g_unix_connection_receive_credentials_async [Gio 2.32]
g_unix_connection_receive_credentials_finish [Gio 2.32]	g_unix_connection_receive_fd [Gio 2.32]
g_unix_connection_send_credentials [Gio 2.32]	g_unix_connection_send_credentials_async [Gio 2.32]
g_unix_connection_send_credentials_finish [Gio 2.32]	g_unix_connection_send_fd [Gio 2.32]
g_unix_credentials_message_get_credentials [Gio 2.32]	g_unix_credentials_message_get_type [Gobject 2.32]
g_unix_credentials_message_is_supported [Gio 2.32]	g_unix_credentials_message_new [Gio 2.32]
g_unix_credentials_message_new_with_credentials [Gio 2.32]	g_unix_fd_list_append [Gio 2.32]
g_unix_fd_list_get [Gio 2.32]	g_unix_fd_list_get_length [Gio 2.32]
g_unix_fd_list_get_type [Gobject 2.32]	g_unix_fd_list_new [Gio 2.32]
g_unix_fd_list_new_from_array [Gio 2.32]	g_unix_fd_list_peek_fds [Gio 2.32]
g_unix_fd_list_steal_fds [Gio 2.32]	g_unix_fd_message_append_fd [Gio 2.32]
g_unix_fd_message_get_fd_list [Gio 2.32]	g_unix_fd_message_get_type [Gobject 2.32]
g_unix_fd_message_new [Gio 2.32]	g_unix_fd_message_new_with_fd_list [Gio 2.32]
g_unix_fd_message_steal_fds [Gio 2.32]	g_unix_input_stream_get_close_fd [Gio 2.32]
g_unix_input_stream_get_fd [Gio 2.32]	g_unix_input_stream_get_type [Gobject 2.32]

g_unix_input_stream_new [Gio 2.32]	g_unix_input_stream_set_close_fd [Gio 2.32]
g_unix_is_mount_path_system_internal [Gio 2.32]	g_unix_mount_at [Gio 2.32]
g_unix_mount_compare [Gio 2.32]	g_unix_mount_free [Gio 2.32]
g_unix_mount_get_device_path [Gio 2.32]	g_unix_mount_get_fs_type [Gio 2.32]
g_unix_mount_get_mount_path [Gio 2.32]	g_unix_mount_guess_can_eject [Gio 2.32]
g_unix_mount_guess_icon [Gio 2.32]	g_unix_mount_guess_name [Gio 2.32]
g_unix_mount_guess_should_display [Gio 2.32]	g_unix_mount_is_readonly [Gio 2.32]
g_unix_mount_is_system_internal [Gio 2.32]	g_unix_mount_monitor_get_type [Gobject 2.32]
g_unix_mount_monitor_new [Gio 2.32]	g_unix_mount_monitor_set_rate_limit [Gio 2.32]
g_unix_mount_point_compare [Gio 2.32]	g_unix_mount_point_free [Gio 2.32]
g_unix_mount_point_get_device_path [Gio 2.32]	g_unix_mount_point_get_fs_type [Gio 2.32]
g_unix_mount_point_get_mount_path [Gio 2.32]	g_unix_mount_point_get_options [Gio 2.32]
g_unix_mount_point_guess_can_eject [Gio 2.32]	g_unix_mount_point_guess_icon [Gio 2.32]
g_unix_mount_point_guess_name [Gio 2.32]	g_unix_mount_point_is_loopback [Gio 2.32]
g_unix_mount_point_is_readonly [Gio 2.32]	g_unix_mount_point_is_user_mountable [Gio 2.32]
g_unix_mount_points_changed_since [Gio 2.32]	g_unix_mount_points_get [Gio 2.32]
g_unix_mounts_changed_since [Gio 2.32]	g_unix_mounts_get [Gio 2.32]
g_unix_output_stream_get_close_fd [Gio 2.32]	g_unix_output_stream_get_fd [Gio 2.32]
g_unix_output_stream_get_type [Gobject 2.32]	g_unix_output_stream_new [Gio 2.32]
g_unix_output_stream_set_close_fd [Gio 2.32]	g_unix_socket_address_abstract_names_supported [Gio 2.32]
g_unix_socket_address_get_address_type [Gio 2.32]	g_unix_socket_address_get_is_abstract [Gio 2.32]

g_unix_socket_address_get_path [Gio 2.32]	g_unix_socket_address_get_path_len [Gio 2.32]
g_unix_socket_address_get_type [Gobject 2.32]	g_unix_socket_address_new [Gio 2.32]
g_unix_socket_address_new_abstract [Gio 2.32]	g_unix_socket_address_new_with_type [Gio 2.32]
g_unix_socket_address_type_get_type [Gobject 2.32]	g_vfs_get_default [Gio 2.32]
g_vfs_get_file_for_path [Gio 2.32]	g_vfs_get_file_for_uri [Gio 2.32]
g_vfs_get_local [Gio 2.32]	g_vfs_get_supported_uri_schemes [Gio 2.32]
g_vfs_get_type [Gobject 2.32]	g_vfs_is_active [Gio 2.32]
g_vfs_parse_name [Gio 2.32]	g_volume_can_eject [Gio 2.32]
g_volume_can_mount [Gio 2.32]	g_volume_eject [Gio 2.32]
g_volume_eject_finish [Gio 2.32]	g_volume_eject_with_operation [Gio 2.32]
g_volume_eject_with_operation_finish [Gio 2.32]	g_volume_enumerate_identifiers [Gio 2.32]
g_volume_get_activation_root [Gio 2.32]	g_volume_get_drive [Gio 2.32]
g_volume_get_icon [Gio 2.32]	g_volume_get_identifier [Gio 2.32]
g_volume_get_mount [Gio 2.32]	g_volume_get_name [Gio 2.32]
g_volume_get_sort_key [Gio 2.32]	g_volume_get_type [Gobject 2.32]
g_volume_get_uuid [Gio 2.32]	g_volume_monitor_adopt_orphan_mount [Gio 2.32]
g_volume_monitor_get [Gio 2.32]	g_volume_monitor_get_connected_drives [Gio 2.32]
g_volume_monitor_get_mount_for_uuid [Gio 2.32]	g_volume_monitor_get_mounts [Gio 2.32]
g_volume_monitor_get_type [Gobject 2.32]	g_volume_monitor_get_volume_for_uuid [Gio 2.32]
g_volume_monitor_get_volumes [Gio 2.32]	g_volume_mount [Gio 2.32]
g_volume_mount_finish [Gio 2.32]	g_volume_should_automount [Gio 2.32]
g_zlib_compressor_format_get_type [Gobject 2.32]	g_zlib_compressor_get_file_info [Gio 2.32]
g_zlib_compressor_get_type [Gobject 2.32]	g_zlib_compressor_new [Gio 2.32]
g_zlib_compressor_set_file_info [Gio 2.32]	g_zlib_decompressor_get_file_info [Gio 2.32]

<code>g_zlib_decompressor_get_type</code> [Gobject 2.32]	<code>g_zlib_decompressor_new</code> [Gio 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for libgio-2.0 interfaces specified in Table 17-100, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-100 libgio-2.0 - libgio-2.0 interfaces Deprecated Function Interfaces

<code>g_application_set_action_group</code> [Gio 2.32]	<code>g_desktop_app_info_lookup_get_default_for_uri_scheme</code> [LSB]
<code>g_desktop_app_info_lookup_get_type</code> [Gobject 2.32]	<code>g_drive_eject</code> [Gio 2.32]
<code>g_drive_eject_finish</code> [Gio 2.32]	<code>g_file_eject_mountable</code> [Gio 2.32]
<code>g_file_eject_mountable_finish</code> [Gio 2.32]	<code>g_file_unmount_mountable</code> [Gio 2.32]
<code>g_file_unmount_mountable_finish</code> [Gio 2.32]	<code>g_mount_eject</code> [Gio 2.32]
<code>g_mount_eject_finish</code> [Gio 2.32]	<code>g_mount_unmount</code> [Gio 2.32]
<code>g_mount_unmount_finish</code> [Gio 2.32]	<code>g_tls_connection_get_use_system_certificate</code> [Gio 2.32]
<code>g_tls_connection_set_use_system_certificate</code> [Gio 2.32]	<code>g_unix_socket_address_get_is_abstract</code> [Gio 2.32]
<code>g_unix_socket_address_new_abstract</code> [Gio 2.32]	<code>g_volume_eject</code> [Gio 2.32]
<code>g_volume_eject_finish</code> [Gio 2.32]	<code>g_volume_monitor_adopt_orphan_mount</code> [Gio 2.32]

17.12 Data Definitions for libgio-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libgio-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.12.1 gio-unix-2.0/gio/gdesktopappinfo.h

```

#define G_TYPE_DESKTOP_APP_INFO (g_desktop_app_info_get_type ())
#define G_TYPE_DESKTOP_APP_INFO_LOOKUP (G_TYPE_DESKTOP_APP_INFO_LOOKUP)
(g_desktop_app_info_lookup_get_type ()) /* DEPRECATED */
#define G_DESKTOP_APP_INFO_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DESKTOP_APP_INFO, GDesktopAppInfoClass))
#define G_IS_DESKTOP_APP_INFO_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_DESKTOP_APP_INFO))
#define G_DESKTOP_APP_INFO(o) (G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_DESKTOP_APP_INFO, GDesktopAppInfo))
#define G_DESKTOP_APP_INFO_LOOKUP(obj) (G_TYPE_CHECK_INSTANCE_CAST((obj), G_TYPE_DESKTOP_APP_INFO_LOOKUP, GDesktopAppInfoLookup)) /* DEPRECATED */
#define G_IS_DESKTOP_APP_INFO(o) (G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_DESKTOP_APP_INFO))
#define G_IS_DESKTOP_APP_INFO_LOOKUP(obj) (G_TYPE_CHECK_INSTANCE_TYPE((obj), G_TYPE_DESKTOP_APP_INFO_LOOKUP)) /* DEPRECATED */
#define G_DESKTOP_APP_INFO_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_DESKTOP_APP_INFO, GDesktopAppInfoClass))
#define G_DESKTOP_APP_INFO_LOOKUP_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE((obj), G_TYPE_DESKTOP_APP_INFO_LOOKUP, GDesktopAppInfoLookupIface)) /* DEPRECATED */
#define G_DESKTOP_APP_INFO_LOOKUP_EXTENSION_POINT_NAME "gio-desktop-app-info-lookup" /* DEPRECATED */

typedef struct _GDesktopAppInfo GDesktopAppInfo;
typedef struct _GDesktopAppInfoClass {
    GObjectClass parent_class;
} GDesktopAppInfoClass;
typedef struct _GDesktopAppInfoLookup GDesktopAppInfoLookup;
typedef struct _GDesktopAppInfoLookupIface {
    GTypeInterface g_iface;
    GAppInfo *(*get_default_for_uri_scheme) (GDesktopAppInfoLookup *
                                              lookup,
                                              const char *uri_scheme);
} GDesktopAppInfoLookupIface;
typedef void (*GDesktopAppLaunchCallback) (GDesktopAppInfo *
                                             appinfo,
                                             GPid pid, gpointer user_data);
extern const char *g_desktop_app_info_get_categories(GDesktopAppInfo *
                                                    info);
extern const char *g_desktop_app_info_get_filename(GDesktopAppInfo *
                                                    info);
extern const char *g_desktop_app_info_get_generic_name(GDesktopAppInfo *
                                                       info);
extern gboolean g_desktop_app_info_get_is_hidden(GDesktopAppInfo *
                                                  info);
extern const char *g_desktop_app_info_get_keywords(GDesktopAppInfo *
                                                    info);
extern gboolean g_desktop_app_info_get_nodisplay(GDesktopAppInfo *
                                                  info);
extern gboolean g_desktop_app_info_get_show_in(GDesktopAppInfo *
                                                info,
                                                const char *desktop_env);
extern GType g_desktop_app_info_get_type(void);
extern gboolean g_desktop_app_info_launch_uris_as_manager(GDesktopAppInfo *

```

```

        appinfo,
        GList * uris,
        GAppLaunchContext
        * launch_context,
        GSpawnFlags
        spawn_flags,

GSpawnChildSetupFunc

        user_setup,
        gpointer
        user_setup_data,

GDesktopAppLaunchCallback

        pid_callback,
        gpointer
        pid_callback_data,
        GError * *error);

extern GAppInfo
    *g_desktop_app_info_lookup_get_default_for_uri_scheme
    (GDesktopAppInfoLookup * lookup, const char *uri_scheme);
extern GType g_desktop_app_info_lookup_get_type(void);
extern GDesktopAppInfo *g_desktop_app_info_new(const char
*desktop_id);
extern GDesktopAppInfo *g_desktop_app_info_new_from_filename(const
char
        *filename);
extern
        GDesktopAppInfo
*g_desktop_app_info_new_from_keyfile(GKeyFile *
        key_file);
extern void g_desktop_app_info_set_desktop_env(const char
*desktop_env);

```

17.12.2 gio-unix-2.0/gio/gfiledescriptorbased.h

```

#define G_TYPE_FILE_DESCRIPTOR_BASED
(g_file_descriptor_based_get_type ())
#define G_FILE_DESCRIPTOR_BASED(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), G_TYPE_FILE_DESCRIPTOR_BASED,
GFileDescriptorBased))
#define G_IS_FILE_DESCRIPTOR_BASED(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), G_TYPE_FILE_DESCRIPTOR_BASED))
#define G_FILE_DESCRIPTOR_BASED_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj),
G_TYPE_FILE_DESCRIPTOR_BASED, GFileDescriptorBasedIface))

typedef struct _GFileDescriptorBasedIface {
    GTypeInterface g_iface;
    int (*get_fd) (GFileDescriptorBased * fd_based);
} GFileDescriptorBasedIface;
extern int g_file_descriptor_based_get_fd(GFileDescriptorBased *
fd_based);
extern GType g_file_descriptor_based_get_type(void);

```

17.12.3 gio-unix-2.0/gio/gunixconnection.h

```

#define G_UNIX_CONNECTION_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_UNIX_CONNECTION, GUnixConnectionClass))
#define G_IS_UNIX_CONNECTION_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_UNIX_CONNECTION))
#define G_UNIX_CONNECTION(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_UNIX_CONNECTION, GUnixConnection))
#define G_IS_UNIX_CONNECTION(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_UNIX_CONNECTION))

```

```

#define G_UNIX_CONNECTION_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_UNIX_CONNECTION,
GUnixConnectionClass))
#define G_TYPE_UNIX_CONNECTION (g_unix_connection_get_type ())

typedef struct _GUnixConnection {
    GSocketConnection parent_instance;
    GUnixConnectionPrivate *priv;
} GUnixConnection;
typedef struct _GUnixConnectionPrivate GUnixConnectionPrivate;
typedef struct _GUnixConnectionClass {
    GSocketConnectionClass parent_class;
} GUnixConnectionClass;
extern GType g_unix_connection_get_type(void);
extern GCredentials
*g_unix_connection_receive_credentials(GUnixConnection
    * connection,
    GCancellable *
    cancellable,
    GError *
    *error);

extern void
g_unix_connection_receive_credentials_async(GUnixConnection *
    connection,
    GCancellable *
    cancellable,
    GAsyncReadyCallback
    callback,
    gpointer
    user_data);

extern GCredentials
*g_unix_connection_receive_credentials_finish(GUnixConnection
    *
    connection,
    GAsyncResult * result,
    GError * *error);

extern gint g_unix_connection_receive_fd(GUnixConnection *
    connection,
    GCancellable * cancellable,
    GError * *error);

extern gboolean g_unix_connection_send_credentials(GUnixConnection
    *
    connection,
    GCancellable *
    cancellable,
    GError * *error);

extern void
g_unix_connection_send_credentials_async(GUnixConnection *
    connection,
    GCancellable *
    cancellable,
    GAsyncReadyCallback
    callback,
    gpointer user_data);

extern gboolean
g_unix_connection_send_credentials_finish(GUnixConnection *
    connection,
    GAsyncResult *
    result,
    GError * *error);

extern gboolean g_unix_connection_send_fd(GUnixConnection *
    connection,
    gint fd,
    GCancellable * cancellable,
    GError * *error);

```

17.12.4 gio-unix-2.0/gio/gunixcredentialsmessage.h

```

#define G_UNIX_CREDENTIALS_MESSAGE_CLASS(c) \
(G_TYPE_CHECK_CLASS_CAST ((c), G_TYPE_UNIX_CREDENTIALS_MESSAGE, \
GUnixCredentialsMessageClass))
#define G_IS_UNIX_CREDENTIALS_MESSAGE_CLASS(c) \
(G_TYPE_CHECK_CLASS_TYPE ((c), G_TYPE_UNIX_CREDENTIALS_MESSAGE))
#define G_UNIX_CREDENTIALS_MESSAGE(o) \
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_UNIX_CREDENTIALS_MESSAGE, \
GUnixCredentialsMessage))
#define G_IS_UNIX_CREDENTIALS_MESSAGE(o) \
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_UNIX_CREDENTIALS_MESSAGE))
#define G_UNIX_CREDENTIALS_MESSAGE_GET_CLASS(o) \
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_UNIX_CREDENTIALS_MESSAGE, \
GUnixCredentialsMessageClass))
#define G_TYPE_UNIX_CREDENTIALS_MESSAGE \
(g_unix_credentials_message_get_type ())

typedef struct _GUnixCredentialsMessagePrivate
GUnixCredentialsMessagePrivate;
typedef struct _GUnixCredentialsMessageClass {
GSocketControlMessageClass parent_class;
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
} GUnixCredentialsMessageClass;
struct _GUnixCredentialsMessage {
GSocketControlMessage parent_instance;
GUnixCredentialsMessagePrivate *priv;
};
extern GCredentials

*g_unix_credentials_message_get_credentials(GUnixCredentialsMessage *
message);
extern GType g_unix_credentials_message_get_type(void);
extern gboolean g_unix_credentials_message_is_supported(void);
extern GSocketControlMessage *g_unix_credentials_message_new(void);
extern GSocketControlMessage
*g_unix_credentials_message_new_with_credentials(GCredentials
credentials);

```

17.12.5 gio-unix-2.0/gio/gunixfdlist.h

```

#define G_UNIX_FD_LIST_CLASS(class) \
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_UNIX_FD_LIST, GUnixFDListClass))
#define G_IS_UNIX_FD_LIST_CLASS(class) \
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_UNIX_FD_LIST))
#define G_UNIX_FD_LIST(inst) \
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_UNIX_FD_LIST, GUnixFDList))
#define G_IS_UNIX_FD_LIST(inst) \
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_UNIX_FD_LIST))
#define G_UNIX_FD_LIST_GET_CLASS(inst) \
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_UNIX_FD_LIST, GUnixFDListClass))
#define G_TYPE_UNIX_FD_LIST \
(g_unix_fd_list_get_type ())

typedef struct _GUnixFDListPrivate GUnixFDListPrivate;
typedef struct _GUnixFDListClass {
GObjectClass parent_class;
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
}

```

```

    void (*_g_reserved5) (void);
} GUnixFDListClass;
struct _GUnixFDList {
    GObject parent_instance;
    GUnixFDListPrivate *priv;
};
extern gint g_unix_fd_list_append(GUnixFDList * list, gint fd,
                                  GError * *error);
extern gint g_unix_fd_list_get(GUnixFDList * list, gint index_,
                                GError * *error);
extern gint g_unix_fd_list_get_length(GUnixFDList * list);
extern GType g_unix_fd_list_get_type(void);
extern GUnixFDList *g_unix_fd_list_new(void);
extern GUnixFDList *g_unix_fd_list_new_from_array(const int *fds,
                                                    gint n_fds);
extern const int *g_unix_fd_list_peek_fds(GUnixFDList * list,
                                           gint * length);
extern gint *g_unix_fd_list_steal_fds(GUnixFDList * list, gint *
length);

```

17.12.6 gio-unix-2.0/gio/gunixfdmessage.h

```

#define G_UNIX_FD_MESSAGE_CLASS(class) (G_TYPE_CHECK_CLASS_CAST(
((class), G_TYPE_UNIX_FD_MESSAGE, GUnixFDMessageClass))
#define G_IS_UNIX_FD_MESSAGE_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_UNIX_FD_MESSAGE))
#define G_UNIX_FD_MESSAGE(inst) (G_TYPE_CHECK_INSTANCE_CAST(
((inst), G_TYPE_UNIX_FD_MESSAGE, GUnixFDMessage))
#define G_IS_UNIX_FD_MESSAGE(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_UNIX_FD_MESSAGE))
#define G_UNIX_FD_MESSAGE_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_UNIX_FD_MESSAGE,
GUnixFDMessageClass))
#define G_TYPE_UNIX_FD_MESSAGE (g_unix_fd_message_get_type ())

typedef struct _GUnixFDMessagePrivate GUnixFDMessagePrivate;
typedef struct _GUnixFDMessageClass {
    GSocketControlMessageClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
} GUnixFDMessageClass;
typedef struct _GUnixFDMessage {
    GSocketControlMessage parent_instance;
    GUnixFDMessagePrivate *priv;
} GUnixFDMessage;
extern gboolean g_unix_fd_message_append_fd(GUnixFDMessage *
message,
                                           gint fd, GError * *error);
extern GUnixFDList *g_unix_fd_message_get_fd_list(GUnixFDMessage *
message);
extern GType g_unix_fd_message_get_type(void);
extern GSocketControlMessage *g_unix_fd_message_new(void);
extern GSocketControlMessage
*g_unix_fd_message_new_with_fd_list(GUnixFDList * fd_list);
extern gint *g_unix_fd_message_steal_fds(GUnixFDMessage * message,
gint * length);

```

17.12.7 gio-unix-2.0/gio/gunixinputstream.h

```

#define G_UNIX_INPUT_STREAM_CLASS(k) G_TYPE_UNIX_INPUT_STREAM,
(G_TYPE_CHECK_CLASS_CAST((k), G_UNIX_INPUT_STREAM_CLASS(k),
GUnixInputStreamClass))

```



```

#define G_IS_UNIX_INPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_UNIX_INPUT_STREAM))
#define G_UNIX_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_UNIX_INPUT_STREAM, GUnixInputStream))
#define G_IS_UNIX_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_UNIX_INPUT_STREAM))
#define G_UNIX_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_UNIX_INPUT_STREAM,
GUnixInputStreamClass))
#define G_TYPE_UNIX_INPUT_STREAM
(g_unix_input_stream_get_type ())

typedef struct _GUnixInputStream {
    GInputStream parent_instance;
    GUnixInputStreamPrivate *priv;
} GUnixInputStream;
typedef struct _GUnixInputStreamClass {
    GInputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GUnixInputStreamClass;
typedef struct _GUnixInputStreamPrivate GUnixInputStreamPrivate;
extern gboolean g_unix_input_stream_get_close_fd(GUnixInputStream
*
stream);
extern gint g_unix_input_stream_get_fd(GUnixInputStream * stream);
extern GType g_unix_input_stream_get_type(void);
extern GInputStream *g_unix_input_stream_new(gint fd, gboolean
close_fd);
extern void g_unix_input_stream_set_close_fd(GUnixInputStream *
stream,
gboolean close_fd);

```

17.12.8 gio-unix-2.0/gio/gunixmounts.h

```

#define G_UNIX_MOUNT_MONITOR_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_UNIX_MOUNT_MONITOR,
GUnixMountMonitorClass))
#define G_IS_UNIX_MOUNT_MONITOR_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_UNIX_MOUNT_MONITOR))
#define G_UNIX_MOUNT_MONITOR(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_UNIX_MOUNT_MONITOR, GUnixMountMonitor))
#define G_IS_UNIX_MOUNT_MONITOR(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_UNIX_MOUNT_MONITOR))
#define G_TYPE_UNIX_MOUNT_MONITOR
(g_unix_mount_monitor_get_type ())

typedef struct _GUnixMountEntry GUnixMountEntry;
typedef struct _GUnixMountPoint GUnixMountPoint;
typedef struct _GUnixMountMonitor GUnixMountMonitor;
typedef struct _GUnixMountMonitorClass GUnixMountMonitorClass;
extern gboolean g_unix_is_mount_path_system_internal(const char
*mount_path);
extern GUnixMountEntry *g_unix_mount_at(const char *mount_path,
guint64 * time_read);
extern gint g_unix_mount_compare(GUnixMountEntry * mount1,
GUnixMountEntry * mount2);
extern void g_unix_mount_free(GUnixMountEntry * mount_entry);
extern const char *g_unix_mount_get_device_path(GUnixMountEntry *
mount_entry);
extern const char *g_unix_mount_get_fs_type(GUnixMountEntry *
mount_entry);

```

```

extern const char *g_unix_mount_get_mount_path(GUnixMountEntry *
                                                mount_entry);
extern gboolean g_unix_mount_guess_can_eject(GUnixMountEntry *
                                              mount_entry);
extern GIcon      *g_unix_mount_guess_icon(GUnixMountEntry *
      mount_entry);
extern char      *g_unix_mount_guess_name(GUnixMountEntry *
      mount_entry);
extern gboolean g_unix_mount_guess_should_display(GUnixMountEntry
*
                                                mount_entry);
extern gboolean g_unix_mount_is_readonly(GUnixMountEntry *
      mount_entry);
extern gboolean g_unix_mount_is_system_internal(GUnixMountEntry *
      mount_entry);
extern GType g_unix_mount_monitor_get_type(void);
extern GUnixMountMonitor *g_unix_mount_monitor_new(void);
extern void g_unix_mount_monitor_set_rate_limit(GUnixMountMonitor
*
                                                mount_monitor,
                                                int limit_msec);
extern gint g_unix_mount_point_compare(GUnixMountPoint * mount1,
      GUnixMountPoint * mount2);
extern void g_unix_mount_point_free(GUnixMountPoint * mount_point);
extern const char *g_unix_mount_point_get_device_path(GUnixMountPoint *
      mount_point);
extern const char *g_unix_mount_point_get_fs_type(GUnixMountPoint
*
      mount_point);
extern const char *g_unix_mount_point_get_mount_path(GUnixMountPoint *
      mount_point);
extern const char *g_unix_mount_point_get_options(GUnixMountPoint
*
      mount_point);
extern gboolean g_unix_mount_point_guess_can_eject(GUnixMountPoint
*
      mount_point);
extern GIcon *g_unix_mount_point_guess_icon(GUnixMountPoint *
      mount_point);
extern char *g_unix_mount_point_guess_name(GUnixMountPoint *
      mount_point);
extern gboolean g_unix_mount_point_is_loopback(GUnixMountPoint *
      mount_point);
extern gboolean g_unix_mount_point_is_readonly(GUnixMountPoint *
      mount_point);
extern gboolean g_unix_mount_point_is_user_montable(GUnixMountPoint *
      mount_point);
extern gboolean g_unix_mount_points_changed_since(guint64 time);
extern GList *g_unix_mount_points_get(guint64 * time_read);
extern gboolean g_unix_mounts_changed_since(guint64 time);
extern GList *g_unix_mounts_get(guint64 * time_read);

```

17.12.9 gio-unix-2.0/gio/gunixoutputstream.h

```

#define G_UNIX_OUTPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
      GUnixOutputStreamClass))
#define G_IS_UNIX_OUTPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k),
      GUnixOutputStreamClass))
#define G_UNIX_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST((o),
      GUnixOutputStream))

```

```

#define G_IS_UNIX_OUTPUT_STREAM(o)          (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_UNIX_OUTPUT_STREAM))
#define G_UNIX_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_UNIX_OUTPUT_STREAM,
GUnixOutputStreamClass))
#define G_TYPE_UNIX_OUTPUT_STREAM
(g_unix_output_stream_get_type ())

typedef struct _GUnixOutputStream {
    GOutputStream parent_instance;
    GUnixOutputStreamPrivate *priv;
} GUnixOutputStream;
typedef struct _GUnixOutputStreamClass {
    GOutputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GUnixOutputStreamClass;
typedef struct _GUnixOutputStreamPrivate GUnixOutputStreamPrivate;
extern gboolean
g_unix_output_stream_get_close_fd(GUnixOutputStream *
stream);
extern gint g_unix_output_stream_get_fd(GUnixOutputStream *
stream);
extern GType g_unix_output_stream_get_type(void);
extern GOutputStream *g_unix_output_stream_new(gint fd, gboolean
close_fd);
extern void g_unix_output_stream_set_close_fd(GUnixOutputStream *
stream,
gboolean close_fd);

```

17.12.10 gio-unix-2.0/gio/gunixsocketaddress.h

```

#define G_UNIX_SOCKET_ADDRESS_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_UNIX_SOCKET_ADDRESS,
GUnixSocketAddressClass))
#define G_IS_UNIX_SOCKET_ADDRESS_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_UNIX_SOCKET_ADDRESS))
#define G_UNIX_SOCKET_ADDRESS(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_UNIX_SOCKET_ADDRESS, GUnixSocketAddress))
#define G_IS_UNIX_SOCKET_ADDRESS(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_UNIX_SOCKET_ADDRESS))
#define G_UNIX_SOCKET_ADDRESS_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_UNIX_SOCKET_ADDRESS,
GUnixSocketAddressClass))
#define G_TYPE_UNIX_SOCKET_ADDRESS
(g_unix_socket_address_get_type ())

typedef struct _GUnixSocketAddress {
    GSocketAddress parent_instance;
    GUnixSocketAddressPrivate *priv;
} GUnixSocketAddress;
typedef struct _GUnixSocketAddressClass {
    GSocketAddressClass parent_class;
} GUnixSocketAddressClass;
typedef struct _GUnixSocketAddressPrivate
GUnixSocketAddressPrivate;
extern gboolean
g_unix_socket_address_abstract_names_supported(void);
extern GUnixSocketAddressType
g_unix_socket_address_get_address_type(GUnixSocketAddress *
address);

```

```

extern gboolean
g_unix_socket_address_get_is_abstract(GUnixSocketAddress *
                                     address);

extern const char
*g_unix_socket_address_get_path(GUnixSocketAddress *
                               address);

extern gsize g_unix_socket_address_get_path_len(GUnixSocketAddress
*
                                             address);

extern GType g_unix_socket_address_get_type(void);
extern GSocketAddress *g_unix_socket_address_new(const char *path);
extern GSocketAddress *g_unix_socket_address_new_abstract(const
char *path,
                                                           gint path_len);
extern GSocketAddress *g_unix_socket_address_new_with_type(const
char
                                                           *path,
                                                           gint path_len,
                                                           GUnixSocketAddressType
                                                           type);

```

17.12.11 glib-2.0/gio/gaction.h

```

#define G_TYPE_ACTION (g_action_get_type ())
#define G_ACTION(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_ACTION, GAction))
#define G_IS_ACTION(inst) (G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_ACTION))
#define G_ACTION_GET_IFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst), G_TYPE_ACTION,
GActionInterface))
#define SA_STACK SA_ONSTACK

typedef struct _GActionInterface {
    GTypeInterface g_iface;
    const gchar *get_name;
    const GVariantType(*get_parameter_type) (GAction * action);
    const GVariantType(*get_state_type) (GAction * action);
    GVariant *(*get_state_hint) (GAction * action);
    gboolean(*get_enabled) (GAction * action);
    GVariant *(*get_state) (GAction * action);
    void (*change_state) (GAction * action, GVariant * value);
    void (*activate) (GAction * action, GVariant * parameter);
} GActionInterface;

extern void g_action_activate(GAction * action, GVariant *
parameter);
extern void g_action_change_state(GAction * action, GVariant *
value);
extern gboolean g_action_get_enabled(GAction * action);
extern const char *g_action_get_name(GAction * action);
extern GVariantType *g_action_get_parameter_type(GAction * action);
extern GVariant *g_action_get_state(GAction * action);
extern GVariant *g_action_get_state_hint(GAction * action);
extern GVariantType *g_action_get_state_type(GAction * action);
extern GType g_action_get_type(void);

```

17.12.12 glib-2.0/gio/gactiongroup.h

```

#define G_TYPE_ACTION_GROUP (g_action_group_get_type ())
#define G_ACTION_GROUP(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_ACTION_GROUP, GActionGroup))

```

```

#define G_IS_ACTION_GROUP(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_ACTION_GROUP))
#define G_ACTION_GROUP_GET_IFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst), G_TYPE_ACTION_GROUP,
GActionGroupInterface))

typedef struct _GActionGroupInterface {
    GTypeInterface g_iface;
    gboolean(*has_action) (GActionGroup * action_group,
                           const gchar * action_name);
    gchar **(*list_actions) (GActionGroup * action_group);
    gboolean(*get_action_enabled) (GActionGroup * action_group,
                                   const gchar * action_name);
    const GVariantType *(*get_action_parameter_type) (GActionGroup
*
                                                         action_group,
                                                         const gchar *
                                                         action_name);
    const GVariantType *(*get_action_state_type) (GActionGroup *
                                                         action_group,
                                                         const gchar *
                                                         action_name);
    GVariant *(*get_action_state_hint) (GActionGroup * action_group,
                                         const gchar * action_name);
    GVariant *(*get_action_state) (GActionGroup * action_group,
                                   const gchar * action_name);
    void (*change_action_state) (GActionGroup * action_group,
                                 const gchar * action_name,
                                 GVariant * value);
    void (*activate_action) (GActionGroup * action_group,
                             const gchar * action_name,
                             GVariant * parameter);
    void (*action_added) (GActionGroup * action_group,
                          const gchar * action_name);
    void (*action_removed) (GActionGroup * action_group,
                            const gchar * action_name);
    void (*action_enabled_changed) (GActionGroup * action_group,
                                    const gchar * action_name,
                                    gboolean enabled);
    void (*action_state_changed) (GActionGroup * action_group,
                                  const gchar * action_name,
                                  GVariant * state);
    gboolean(*query_action) (GActionGroup * action_group,
                             const gchar * action_name,
                             gboolean * enabled,
                             const GVariantType * parameter_type,
                             const GVariantType * state_type,
                             GVariant * *state_hint, GVariant * *state);
} GActionGroupInterface;
extern void g_action_group_action_added(GActionGroup *
action_group,
                                         const char *action_name);
extern void g_action_group_action_enabled_changed(GActionGroup *
action_group,
                                                  const char *action_name,
                                                  gboolean enabled);
extern void g_action_group_action_removed(GActionGroup *
action_group,
                                         const char *action_name);
extern void g_action_group_action_state_changed(GActionGroup *
action_group,
                                                const char *action_name,
                                                GVariant * state);
extern void g_action_group_activate_action(GActionGroup *
action_group,
                                           const char *action_name,

```

```

                                GVariant * parameter);
extern void g_action_group_change_action_state(GActionGroup *
action_group,
                                const char *action_name,
                                GVariant * value);
extern gboolean g_action_group_get_action_enabled(GActionGroup *
action_group,
                                const char *action_name);
extern
                                GVariantType
*g_action_group_get_action_parameter_type(GActionGroup
*
                                action_group,
                                const char
                                *action_name);
extern GVariant *g_action_group_get_action_state(GActionGroup *
action_group,
                                const char *action_name);
extern GVariant *g_action_group_get_action_state_hint(GActionGroup
*
                                action_group,
                                const char
                                *action_name);
extern
                                GVariantType
*g_action_group_get_action_state_type(GActionGroup *
action_group,
                                const char
                                *action_name);
extern GType g_action_group_get_type(void);
extern gboolean g_action_group_has_action(GActionGroup *
action_group,
                                const char *action_name);
extern gchar **g_action_group_list_actions(GActionGroup *
action_group);
extern gboolean g_action_group_query_action(GActionGroup *
action_group,
                                const char *action_name,
                                gboolean * enabled,
                                const GVariantType *
                                parameter_type,
                                const GVariantType *
                                state_type,
                                GVariant * *state_hint,
                                GVariant * *state);

```

17.12.13 glib-2.0/gio/gactiongroupexporter.h

```

extern guint g_dbus_connection_export_action_group(GDBusConnection
*
                                connection,
                                const gchar *
                                object_path,
                                GActionGroup *
                                action_group,
                                GError * *error);
extern
                                void
g_dbus_connection_unexport_action_group(GDBusConnection *
                                connection,
                                guint export_id);

```

17.12.14 glib-2.0/gio/gactionmap.h

```

#define G_TYPE_ACTION_MAP (g_action_map_get_type ())

```

```

#define G_ACTION_MAP(inst)      (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_ACTION_MAP, GActionMap))
#define G_IS_ACTION_MAP(inst)   (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_ACTION_MAP))
#define G_ACTION_MAP_GET_IFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst), G_TYPE_ACTION_MAP,
GActionMapInterface))

typedef struct _GActionMapInterface {
    GTypeInterface g_iface;
    GAction *(*lookup_action) (GActionMap * action_map,
                                const gchar * action_name);
    void (*add_action) (GActionMap * action_map, GAction * action);
    void (*remove_action) (GActionMap * action_map,
                            const gchar * action_name);
} GActionMapInterface;
typedef struct _GActionEntry {
    const gchar *name;
    void (*activate) (GSimpleAction * action, GVariant * parameter,
                      gpointer user_data);
    const gchar *parameter_type;
    const gchar *state;
    void (*change_state) (GSimpleAction * action, GVariant * value,
                          gpointer user_data);
    gsize padding[3];
} GActionEntry;
extern void g_action_map_add_action(GActionMap * action_map,
                                    GAction * action);
extern void g_action_map_add_action_entries(GActionMap *
action_map,
                                             const GActionEntry * entries,
                                             gint n_entries,
                                             gpointer user_data);
extern GType g_action_map_get_type(void);
extern GAction *g_action_map_lookup_action(GActionMap * action_map,
                                             const char *action_name);
extern void g_action_map_remove_action(GActionMap * action_map,
                                         const char *action_name);

```

17.12.15 glib-2.0/gio/gappinfo.h

```

#define G_TYPE_APP_INFO (g_app_info_get_type ())
#define G_TYPE_APP_LAUNCH_CONTEXT
(g_app_launch_context_get_type ())
#define G_APP_LAUNCH_CONTEXT_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_APP_LAUNCH_CONTEXT,
GAppLaunchContextClass))
#define G_IS_APP_LAUNCH_CONTEXT_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_APP_LAUNCH_CONTEXT))
#define G_APP_LAUNCH_CONTEXT(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_APP_LAUNCH_CONTEXT, GAppLaunchContext))
#define G_APP_INFO(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_APP_INFO, GAppInfo))
#define G_IS_APP_LAUNCH_CONTEXT(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_APP_LAUNCH_CONTEXT))
#define G_IS_APP_INFO(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_APP_INFO))
#define G_APP_LAUNCH_CONTEXT_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_APP_LAUNCH_CONTEXT,
GAppLaunchContextClass))
#define G_APP_INFO_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_APP_INFO,
GAppInfoInterface))

typedef struct _GAppLaunchContextClass {

```

```

    GObjectClass parent_class;
    char *(*get_display) (GAppLaunchContext * context, GAppInfo *
info,
                        GList * files);
    char *(*get_startup_notify_id) (GAppLaunchContext * context,
                                   GAppInfo * info, GList * files);
    void (*launch_failed) (GAppLaunchContext * context,
                           const char *startup_notify_id);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GAppLaunchContextClass;
typedef struct _GAppLaunchContextPrivate GAppLaunchContextPrivate;
typedef struct _GAppInfoIface {
    GTypeInterface g_iface;
    GAppInfo *(*dup) (GAppInfo * appinfo);
    gboolean(*equal) (GAppInfo * appinfo1, GAppInfo * appinfo2);
    const char *(*get_id) (GAppInfo * appinfo);
    const char *(*get_name) (GAppInfo * appinfo);
    const char *(*get_description) (GAppInfo * appinfo);
    const char *(*get_executable) (GAppInfo * appinfo);
    GIcon *(*get_icon) (GAppInfo * appinfo);
    gboolean(*launch) (GAppInfo * appinfo, GList * files,
                      GAppLaunchContext * launch_context,
                      GError * *error);
    gboolean(*supports_uris) (GAppInfo * appinfo);
    gboolean(*supports_files) (GAppInfo * appinfo);
    gboolean(*launch_uris) (GAppInfo * appinfo, GList * files,
                           GAppLaunchContext * launch_context,
                           GError * *error);
    gboolean(*should_show) (GAppInfo * appinfo);
    gboolean(*set_as_default_for_type) (GAppInfo * appinfo,
                                       const char *content_type,
                                       GError * *error);
    gboolean(*set_as_default_for_extension) (GAppInfo * appinfo,
                                             const char *content_type,
                                             GError * *error);
    gboolean(*add_supports_type) (GAppInfo * appinfo,
                                 const char *content_type,
                                 GError * *error);
    gboolean(*can_remove_supports_type) (GAppInfo * appinfo);
    gboolean(*remove_supports_type) (GAppInfo * appinfo,
                                    const char *content_type,
                                    GError * *error);
    gboolean(*can_delete) (GAppInfo * appinfo);
    gboolean(*do_delete) (GAppInfo * appinfo);
    const char *(*get_commandline) (GAppInfo * appinfo);
    const char *(*get_display_name) (GAppInfo * appinfo);
    gboolean(*set_as_last_used_for_type) (GAppInfo * appinfo,
                                         const char *content_type,
                                         GError * *error);
} GAppInfoIface;
struct _GAppLaunchContext {
    GObject parent_instance;
    GAppLaunchContextPrivate *priv;
};
extern gboolean g_app_info_add_supports_type(GAppInfo * appinfo,
                                             const char *content_type,
                                             GError * *error);
extern gboolean g_app_info_can_delete(GAppInfo * appinfo);
extern gboolean g_app_info_can_remove_supports_type(GAppInfo *
appinfo);
extern GAppInfo *g_app_info_create_from_commandline(const char
*commandline,

```



```

const char
    *application_name,
    GAppInfoCreateFlags
    flags,
    GError * *error);

extern gboolean g_app_info_delete(GAppInfo * appinfo);
extern GAppInfo *g_app_info_dup(GAppInfo * appinfo);
extern gboolean g_app_info_equal(GAppInfo * appinfo1, GAppInfo *
    appinfo2);
extern GList *g_app_info_get_all(void);
extern GList *g_app_info_get_all_for_type(const char
    *content_type);
extern const char *g_app_info_get_commandline(GAppInfo * appinfo);
extern GAppInfo *g_app_info_get_default_for_type(const char
    *content_type,

    gboolean
    must_support_uris);
extern GAppInfo *g_app_info_get_default_for_uri_scheme(const char
    *uri_scheme);
extern const char *g_app_info_get_description(GAppInfo * appinfo);
extern const char *g_app_info_get_display_name(GAppInfo * appinfo);
extern const char *g_app_info_get_executable(GAppInfo * appinfo);
extern GList *g_app_info_get_fallback_for_type(const char
    *content_type);
extern GIcon *g_app_info_get_icon(GAppInfo * appinfo);
extern const char *g_app_info_get_id(GAppInfo * appinfo);
extern const char *g_app_info_get_name(GAppInfo * appinfo);
extern GList *g_app_info_get_recommended_for_type(const char
    *content_type);

extern GType g_app_info_get_type(void);
extern gboolean g_app_info_launch(GAppInfo * appinfo, GList * files,
    GAppLaunchContext * launch_context,
    GError * *error);
extern gboolean g_app_info_launch_default_for_uri(const char *uri,
    GAppLaunchContext *
    launch_context,
    GError * *error);
extern gboolean g_app_info_launch_uris(GAppInfo * appinfo, GList *
    uris,

    GAppLaunchContext
    *
    launch_context,

    GError * *error);
extern gboolean g_app_info_remove_supports_type(GAppInfo * appinfo,
    const char *content_type,
    GError * *error);
extern void g_app_info_reset_type_associations(const char
    *content_type);
extern gboolean g_app_info_set_as_default_for_extension(GAppInfo *
    appinfo,

    const char
    *extension,
    GError * *error);
extern gboolean g_app_info_set_as_default_for_type(GAppInfo *
    appinfo,

    const char
    *content_type,
    GError * *error);
extern gboolean g_app_info_set_as_last_used_for_type(GAppInfo *
    appinfo,

    const char
    *content_type,
    GError * *error);
extern gboolean g_app_info_should_show(GAppInfo * appinfo);
extern gboolean g_app_info_supports_files(GAppInfo * appinfo);
extern gboolean g_app_info_supports_uris(GAppInfo * appinfo);

```

```

extern char *g_app_launch_context_get_display(GAppLaunchContext *
context,
                                             GAppInfo * info,
                                             GList * files);
extern void g_app_launch_context_get_environment(GAppLaunchContext
*
                                             context);
extern
char
*g_app_launch_context_get_startup_notify_id(GAppLaunchContext *
context,
                                             GAppInfo * info,
                                             GList * files);
extern GType g_app_launch_context_get_type(void);
extern void g_app_launch_context_launch_failed(GAppLaunchContext *
context,
                                             const char
                                             *startup_notify_id);
extern GAppLaunchContext *g_app_launch_context_new(void);
extern void g_app_launch_context_setenv(GAppLaunchContext *
context,
                                             const char *variable,
                                             const char *value);
extern void g_app_launch_context_unsetenv(GAppLaunchContext *
context,
                                             const char *variable);

```

17.12.16 glib-2.0/gio/gapplication.h

```

#define G_TYPE_APPLICATION (g_application_get_type ())
#define G_APPLICATION_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_APPLICATION, GApplicationClass))
#define G_IS_APPLICATION_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_APPLICATION))
#define G_APPLICATION(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_APPLICATION, GApplication))
#define G_IS_APPLICATION(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_APPLICATION))
#define G_APPLICATION_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_APPLICATION, GApplicationClass))

typedef struct _GApplicationPrivate GApplicationPrivate;
typedef struct _GApplicationClass {
    GObjectClass parent_class;
    void (*startup) (GApplication * application);
    void (*activate) (GApplication * application);
    void (*open) (GApplication * application, GFile * *files, gint
n_files,
                const gchar * hint);
    int (*command_line) (GApplication * application,
                        GApplicationCommandLine * command_line);
    gboolean(*local_command_line) (GApplication * application,
                                   gchar * **arguments, int
*exit_status);
    void (*before_emit) (GApplication * application,
                        GVariant * platform_data);
    void (*after_emit) (GApplication * application,
                       GVariant * platform_data);
    void (*add_platform_data) (GApplication * application,
                              GVariantBuilder * builder);
    void (*quit_mainloop) (GApplication * application);
    void (*run_mainloop) (GApplication * application);
    void (*shutdown) (GApplication * application);
    gpointer padding[11];
} GApplicationClass;
struct _GApplication {

```

```

    GObject parent_instance;
    GApplicationPrivate *priv;
};
extern void g_application_activate(GApplication * application);
extern const char *g_application_get_application_id(GApplication *
    application);
extern GApplication *g_application_get_default(void);
extern GApplicationFlags g_application_get_flags(GApplication *
    application);
extern guint g_application_get_inactivity_timeout(GApplication *
    application);
extern gboolean g_application_get_is_registered(GApplication *
    application);
extern gboolean g_application_get_is_remote(GApplication *
    application);
extern GType g_application_get_type(void);
extern void g_application_hold(GApplication * application);
extern gboolean g_application_id_is_valid(const char
    *application_id);
extern GApplication *g_application_new(const char *application_id,
    GApplicationFlags flags);
extern void g_application_open(GApplication * application, GFile *
    *files,
    gint n_files, const char *hint);
extern void g_application_quit(GApplication * application);
extern gboolean g_application_register(GApplication * application,
    GCancelable * cancellable,
    GError **error);
extern void g_application_release(GApplication * application);
extern int g_application_run(GApplication * application, int argc,
    char **argv);
extern void g_application_set_action_group(GApplication *
    application,
    GActionGroup * action_group);
extern void g_application_set_application_id(GApplication *
    application,
    const char *application_id);
extern void g_application_set_default(GApplication * application);
extern void g_application_set_flags(GApplication * application,
    GApplicationFlags flags);
extern void g_application_set_inactivity_timeout(GApplication *
    application,
    guint inactivity_timeout);

```

17.12.17 glib-2.0/gio/gapplicationcommandline.h

```

#define G_TYPE_APPLICATION_COMMAND_LINE
(g_application_command_line_get_type ())
#define G_APPLICATION_COMMAND_LINE_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_APPLICATION_COMMAND_LINE,
    GApplicationCommandLineClass))
#define G_IS_APPLICATION_COMMAND_LINE_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class),
    G_TYPE_APPLICATION_COMMAND_LINE))
#define G_APPLICATION_COMMAND_LINE(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst),
    G_TYPE_APPLICATION_COMMAND_LINE, GApplicationCommandLine))
#define G_IS_APPLICATION_COMMAND_LINE(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst),
    G_TYPE_APPLICATION_COMMAND_LINE))
#define G_APPLICATION_COMMAND_LINE_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst),
    G_TYPE_APPLICATION_COMMAND_LINE, GApplicationCommandLineClass))

typedef struct _GApplicationCommandLinePrivate

```

```

    GApplicationCommandLinePrivate;
typedef struct _GApplicationCommandLineClass {
    GObjectClass parent_class;
    void (*print_literal) (GApplicationCommandLine * cmdline,
                           const gchar * message);
    void (*printerr_literal) (GApplicationCommandLine * cmdline,
                              const gchar * message);

    gpointer padding[12];
} GApplicationCommandLineClass;
struct _GApplicationCommandLine {
    GObject parent_instance;
    GApplicationCommandLinePrivate *priv;
};
extern gchar

**g_application_command_line_get_arguments(GApplicationCommandLin
e *
                                         cmdline, int *argc);

extern const char
    *g_application_command_line_get_cwd(GApplicationCommandLine *
cmdline);
extern const char *const

*g_application_command_line_get_envIRON(GApplicationCommandLine *
                                         cmdline);

extern int
g_application_command_line_get_exit_status(GApplicationCommandLin
e *
                                         cmdline);

extern gboolean
g_application_command_line_get_is_remote(GApplicationCommandLine *
                                         cmdline);

extern GVariant

*g_application_command_line_get_platform_data(GApplicationCommand
Line *
                                         cmdline);

extern GType g_application_command_line_get_type(void);
extern const char
    *g_application_command_line_getenv(GApplicationCommandLine *
cmdline,
                                         const char *name);

extern
                                         void
g_application_command_line_print(GApplicationCommandLine *
                                         cmdline, const char *format,
                                         ...);

extern
                                         void
g_application_command_line_printerr(GApplicationCommandLine *
                                     cmdline,
                                     const char *format, ...);

extern void
g_application_command_line_set_exit_status(GApplicationCommandLin
e *
                                         cmdline, int exit_status);

```

17.12.18 glib-2.0/gio/gasyncinitable.h

```

#define G_TYPE_ASYNC_INITABLE (g_async_initable_get_type ())
#define G_ASYNC_INITABLE(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_ASYNC_INITABLE, GAsyncInitable))
#define G_IS_ASYNC_INITABLE(obj) (G_TYPE_CHECK_INSTANCE_TYPE
((obj), G_TYPE_ASYNC_INITABLE))
#define G_ASYNC_INITABLE_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_ASYNC_INITABLE,
GAsyncInitableIface))

```

```

#define G_TYPE_IS_ASYNC_INITABLE(type) (g_type_is_a ((type),
G_TYPE_ASYNC_INITABLE))

typedef struct _GAsyncInitableIface {
    GTypeInterface g_iface;
    void (*init_async) (GAsyncInitable * initable, int io_priority,
                        GCancelable * cancellable,
                        GAsyncReadyCallback callback, gpointer
user_data);
    gboolean (*init_finish) (GAsyncInitable * initable,
GAsyncResult * res,
                        GError * *error);
} GAsyncInitableIface;
extern GType g_async_initable_get_type(void);
extern void g_async_initable_init_async(GAsyncInitable * initable,
int io_priority,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
extern gboolean g_async_initable_init_finish(GAsyncInitable *
initable,
GAsyncResult * res,
GError * *error);
extern void g_async_initable_new_async(GType object_type, int
io_priority,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data,
const gchar * first_property_name,
...);
extern GObject *g_async_initable_new_finish(GAsyncInitable *
initable,
GAsyncResult * res,
GError * *error);
extern void g_async_initable_new_valist_async(GType object_type,
const gchar *
first_property_name,
va_list var_args,
int io_priority,
GCancelable * cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);
extern void g_async_initable_newv_async(GType object_type,
guint n_parameters,
GParameter * parameters,
int io_priority,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);

```

17.12.19 glib-2.0/gio/gasyncresult.h

```

#define G_TYPE_ASYNC_RESULT (g_async_result_get_type ())
#define G_ASYNC_RESULT(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_ASYNC_RESULT, GAsyncResult))
#define G_IS_ASYNC_RESULT(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_ASYNC_RESULT))
#define G_ASYNC_RESULT_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_ASYNC_RESULT,
GAsyncResultIface))

typedef struct _GAsyncResultIface {
    GTypeInterface g_iface;
    gpointer (*get_user_data) (GAsyncResult * res);

```

```

    GObject *(*get_source_object) (GAsyncResult * res);
} GAsyncResultInterface;
extern GObject *g_async_result_get_source_object(GAsyncResult *
res);
extern GType g_async_result_get_type(void);
extern gpointer g_async_result_get_user_data(GAsyncResult * res);

```

17.12.20 glib-2.0/gio/gbufferedinputstream.h

```

#define G_TYPE_BUFFERED_INPUT_STREAM
(g_buffered_input_stream_get_type ())
#define G_BUFFERED_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_BUFFERED_INPUT_STREAM,
GBufferedInputStreamClass))
#define G_IS_BUFFERED_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_BUFFERED_INPUT_STREAM))
#define G_BUFFERED_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_BUFFERED_INPUT_STREAM, GBufferedInputStream))
#define G_IS_BUFFERED_INPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_BUFFERED_INPUT_STREAM))
#define G_BUFFERED_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_BUFFERED_INPUT_STREAM,
GBufferedInputStreamClass))

typedef struct _GBufferedInputStreamClass {
    GFilterInputStreamClass parent_class;
    gssize(*fill) (GBufferedInputStream * stream, gssize count,
        GCancelable * cancellable, GError * *error);
    void (*fill_async) (GBufferedInputStream * stream, gssize count,
        int io_priority, GCancelable * cancellable,
        GAsyncReadyCallback callback, gpointer
user_data);
    gssize(*fill_finish) (GBufferedInputStream * stream,
        GAsyncResult * result, GError * *error);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GBufferedInputStreamClass;
typedef struct _GBufferedInputStreamPrivate
GBufferedInputStreamPrivate;
struct _GBufferedInputStream {
    GFilterInputStream parent_instance;
    GBufferedInputStreamPrivate *priv;
};
extern gssize g_buffered_input_stream_fill(GBufferedInputStream *
stream,
    gssize count,
    GCancelable * cancellable,
    GError * *error);
extern void
g_buffered_input_stream_fill_async(GBufferedInputStream *
    stream, gssize count,
    int io_priority,
    GCancelable * cancellable,
    GAsyncReadyCallback
    callback,
    gpointer user_data);
extern gssize
g_buffered_input_stream_fill_finish(GBufferedInputStream *
    stream,
    GAsyncResult * result,
    GError * *error);

```

```

extern                                     gsize
g_buffered_input_stream_get_available(GBufferedInputStream *
                                     stream);

extern                                     gsize
g_buffered_input_stream_get_buffer_size(GBufferedInputStream *
                                     stream);

extern GType g_buffered_input_stream_get_type(void);
extern GInputStream *g_buffered_input_stream_new(GInputStream *
                                     base_stream);

extern                                     GInputStream
*g_buffered_input_stream_new_sized(GInputStream *
                                     base_stream,
                                     gsize size);

extern gsize g_buffered_input_stream_peek(GBufferedInputStream *
stream,
                                     void *buffer, gsize offset,
                                     gsize count);

extern                                     const void
*g_buffered_input_stream_peek_buffer(GBufferedInputStream
                                     * stream,
                                     gsize * count);

extern int g_buffered_input_stream_read_byte(GBufferedInputStream
* stream,
                                     GCancelable * cancellable,
                                     GError * *error);

extern                                     void
g_buffered_input_stream_set_buffer_size(GBufferedInputStream *
stream, gsize size);

```

17.12.21 glib-2.0/gio/gbufferedoutputstream.h

```

#define G_TYPE_BUFFERED_OUTPUT_STREAM
(g_buffered_output_stream_get_type ())
#define G_BUFFERED_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_BUFFERED_OUTPUT_STREAM,
GBufferedOutputStreamClass))
#define G_IS_BUFFERED_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_BUFFERED_OUTPUT_STREAM))
#define G_BUFFERED_OUTPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_BUFFERED_OUTPUT_STREAM,
GBufferedOutputStream))
#define G_IS_BUFFERED_OUTPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_BUFFERED_OUTPUT_STREAM))
#define G_BUFFERED_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_BUFFERED_OUTPUT_STREAM,
GBufferedOutputStreamClass))

typedef struct _GBufferedOutputStreamClass {
    GFilterOutputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
} GBufferedOutputStreamClass;

typedef struct _GBufferedOutputStreamPrivate
GBufferedOutputStreamPrivate;

struct _GBufferedOutputStream {
    GFilterOutputStream parent_instance;
    GBufferedOutputStreamPrivate *priv;
};

extern gboolean
g_buffered_output_stream_get_auto_grow(GBufferedOutputStream *
stream);

extern                                     gsize
g_buffered_output_stream_get_buffer_size(GBufferedOutputStream
* stream);

extern GType g_buffered_output_stream_get_type(void);

```

```

extern GOutputStream *g_buffered_output_stream_new(GOutputStream *
                                                    base_stream);
extern
                                                    GOutputStream
*g_buffered_output_stream_new_sized(GOutputStream *
                                    base_stream,
                                    gsize size);
extern
                                                    void
g_buffered_output_stream_set_auto_grow(GBufferedOutputStream *
                                       stream,
                                       gboolean auto_grow);
extern
                                                    void
g_buffered_output_stream_set_buffer_size(GBufferedOutputStream
                                         * stream, gsize size);

```

17.12.22 glib-2.0/gio/gcancellable.h

```

#define G_TYPE_CANCELLABLE (g_cancellable_get_type ())
#define G_CANCELLABLE_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_CANCELLABLE, GCancellableClass))
#define G_IS_CANCELLABLE_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_CANCELLABLE))
#define G_CANCELLABLE(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_CANCELLABLE, GCancellable))
#define G_IS_CANCELLABLE(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_CANCELLABLE))
#define G_CANCELLABLE_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_CANCELLABLE, GCancellableClass))

typedef struct _GCancellableClass {
    GObjectClass parent_class;
    void (*cancelled) (GCancellable * cancellable);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GCancellableClass;
typedef struct _GCancellablePrivate GCancellablePrivate;
struct _GCancellable {
    GObject parent_instance;
    GCancellablePrivate *priv;
};
extern void g_cancellable_cancel(GCancellable * cancellable);
extern gulong g_cancellable_connect(GCancellable * cancellable,
                                   GCallback callback, gpointer data,
                                   GDestroyNotify data_destroy_func);
extern void g_cancellable_disconnect(GCancellable * cancellable,
                                    gulong handler_id);
extern GCancellable *g_cancellable_get_current(void);
extern int g_cancellable_get_fd(GCancellable * cancellable);
extern GType g_cancellable_get_type(void);
extern gboolean g_cancellable_is_cancelled(GCancellable *
cancellable);
extern gboolean g_cancellable_make_pollfd(GCancellable *
cancellable,
                                         GPollFD * pollfd);
extern GCancellable *g_cancellable_new(void);
extern void g_cancellable_pop_current(GCancellable * cancellable);
extern void g_cancellable_push_current(GCancellable * cancellable);
extern void g_cancellable_release_fd(GCancellable * cancellable);
extern void g_cancellable_reset(GCancellable * cancellable);
extern gboolean g_cancellable_set_error_if_cancelled(GCancellable *
cancellable,
                                                    GError * *error);

```



```
extern GSource *g_cancellable_source_new(GCancellable *
cancellable);
```

17.12.23 glib-2.0/gio/gcharsetconverter.h

```
#define G_TYPE_CHARSET_CONVERTER
(g_charset_converter_get_type ())
#define G_CHARSET_CONVERTER_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_CHARSET_CONVERTER,
GCharsetConverterClass))
#define G_IS_CHARSET_CONVERTER_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_CHARSET_CONVERTER))
#define G_CHARSET_CONVERTER(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_CHARSET_CONVERTER, GCharsetConverter))
#define G_IS_CHARSET_CONVERTER(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_CHARSET_CONVERTER))
#define G_CHARSET_CONVERTER_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_CHARSET_CONVERTER,
GCharsetConverterClass))

typedef struct _GCharsetConverterClass {
    GObjectClass parent_class;
} GCharsetConverterClass;

extern guint
g_charset_converter_get_num_fallbacks(GCharsetConverter *
converter);

extern GType g_charset_converter_get_type(void);

extern gboolean
g_charset_converter_get_use_fallback(GCharsetConverter *
converter);

extern GCharsetConverter *g_charset_converter_new(const char
*to_charset,
const char *from_charset,
GError **error);

extern void g_charset_converter_set_use_fallback(GCharsetConverter
*
converter,
gboolean use_fallback);
```

17.12.24 glib-2.0/gio/gcontenttype.h

```
extern gboolean g_content_type_can_be_executable(const gchar *
type);
extern gboolean g_content_type_equals(const gchar * type1,
const gchar * type2);
extern gchar *g_content_type_from_mime_type(const gchar *
mime_type);
extern gchar *g_content_type_get_description(const gchar * type);
extern GIcon *g_content_type_get_icon(const gchar * type);
extern gchar *g_content_type_get_mime_type(const gchar * type);
extern gchar *g_content_type_guess(const gchar * filename,
const gchar * data, gsize data_size,
gboolean * result_uncertain);
extern gchar **g_content_type_guess_for_tree(GFile * root);
extern gboolean g_content_type_is_a(const gchar * type,
const gchar * supertype);
extern gboolean g_content_type_is_unknown(const gchar * type);
extern GList *g_content_types_get_registered(void);
```

17.12.25 glib-2.0/gio/gconverter.h

```
#define G_TYPE_CONVERTER (g_converter_get_type ())
```

```

#define G_CONVERTER(obj)          (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_CONVERTER, GConverter))
#define G_IS_CONVERTER(obj)      (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_CONVERTER))
#define G_CONVERTER_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_CONVERTER,
GConverterIface))

typedef struct _GConverterIface {
    GTypeInterface g_iface;
    GConverterResult(*convert) (GConverter * converter, const void
*inbuf,
                                gsize inbuf_size, void *outbuf,
                                gsize outbuf_size, GConverterFlags
flags,
                                gsize * bytes_read, gsize *
bytes_written,
                                GError * *error);
    void (*reset) (GConverter * converter);
} GConverterIface;
extern GConverterResult g_converter_convert(GConverter * converter,
const void *inbuf,
gsize inbuf_size, void
*outbuf,
gsize outbuf_size,
GConverterFlags flags,
gsize * bytes_read,
gsize * bytes_written,
GError * *error);
extern GType g_converter_get_type(void);
extern void g_converter_reset(GConverter * converter);

```

17.12.26 glib-2.0/gio/gconverterinputstream.h

```

#define G_TYPE_CONVERTER_INPUT_STREAM
(g_converter_input_stream_get_type ())
#define G_CONVERTER_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_CONVERTER_INPUT_STREAM,
GConverterInputStreamClass))
#define G_IS_CONVERTER_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_CONVERTER_INPUT_STREAM))
#define G_CONVERTER_INPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_CONVERTER_INPUT_STREAM,
GConverterInputStream))
#define G_IS_CONVERTER_INPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_CONVERTER_INPUT_STREAM))
#define G_CONVERTER_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_CONVERTER_INPUT_STREAM,
GConverterInputStreamClass))

typedef struct _GConverterInputStreamClass {
    GFilterInputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GConverterInputStreamClass;
typedef struct _GConverterInputStreamPrivate
GConverterInputStreamPrivate;
struct _GConverterInputStream {
    GFilterInputStream parent_instance;
    GConverterInputStreamPrivate *priv;
};
extern GConverter

```

```

    *g_converter_input_stream_get_converter(GConverterInputStream
*
    converter_stream);
extern GType g_converter_input_stream_get_type(void);
extern GInputStream *g_converter_input_stream_new(GInputStream *
    base_stream,
    GConverter * converter);

```

17.12.27 glib-2.0/gio/gconverteroutputstream.h

```

#define G_TYPE_CONVERTER_OUTPUT_STREAM
(g_converter_output_stream_get_type ())
#define G_CONVERTER_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_CONVERTER_OUTPUT_STREAM,
GConverterOutputStreamClass))
#define G_IS_CONVERTER_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_CONVERTER_OUTPUT_STREAM))
#define G_CONVERTER_OUTPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_CONVERTER_OUTPUT_STREAM,
GConverterOutputStream))
#define G_IS_CONVERTER_OUTPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_CONVERTER_OUTPUT_STREAM))
#define G_CONVERTER_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_CONVERTER_OUTPUT_STREAM,
GConverterOutputStreamClass))

typedef struct _GConverterOutputStreamClass {
    GFilterOutputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GConverterOutputStreamClass;
typedef struct _GConverterOutputStreamPrivate
    GConverterOutputStreamPrivate;
struct _GConverterOutputStream {
    GFilterOutputStream parent_instance;
    GConverterOutputStreamPrivate *priv;
};
extern GConverter

*g_converter_output_stream_get_converter(GConverterOutputStream *
    converter_stream);
extern GType g_converter_output_stream_get_type(void);
extern GOutputStream *g_converter_output_stream_new(GOutputStream
*
    base_stream,
    GConverter *
    converter);

```

17.12.28 glib-2.0/gio/gcredentials.h

```

#define G_TYPE_CREDENTIALS (g_credentials_get_type ())
#define G_CREDENTIALS_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_CREDENTIALS, GCredentialsClass))
#define G_IS_CREDENTIALS_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_CREDENTIALS))
#define G_CREDENTIALS(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_CREDENTIALS, GCredentials))
#define G_IS_CREDENTIALS(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_CREDENTIALS))

```

```

#define G_CREDENTIALS_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_CREDENTIALS, GCredentialsClass))

typedef struct _GCredentialsClass GCredentialsClass;
extern gpointer g_credentials_get_native(GCredentials *
credentials,
GCredentialsType native_type);
extern GType g_credentials_get_type(void);
extern uid_t g_credentials_get_unix_user(GCredentials *
credentials,
GError * *error);
extern gboolean g_credentials_is_same_user(GCredentials *
credentials,
GCredentials *
other_credentials,
GError * *error);
extern GCredentials *g_credentials_new(void);
extern void g_credentials_set_native(GCredentials * credentials,
GCredentialsType native_type,
gpointer native);
extern gboolean g_credentials_set_unix_user(GCredentials *
credentials,
uid_t uid, GError * *error);
extern gchar *g_credentials_to_string(GCredentials * credentials);

```

17.12.29 glib-2.0/gio/gdatainputstream.h

```

#define G_TYPE_DATA_INPUT_STREAM
(g_data_input_stream_get_type ())
#define G_DATA_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DATA_INPUT_STREAM,
GDataInputStreamClass))
#define G_IS_DATA_INPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_DATA_INPUT_STREAM))
#define G_DATA_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DATA_INPUT_STREAM, GDataInputStream))
#define G_IS_DATA_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_DATA_INPUT_STREAM))
#define G_DATA_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_DATA_INPUT_STREAM,
GDataInputStreamClass))

typedef struct _GDataInputStreamClass {
GBufferedInputStreamClass parent_class;
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
void (*_g_reserved5) (void);
} GDataInputStreamClass;
typedef struct _GDataInputStreamPrivate GDataInputStreamPrivate;
struct _GDataInputStream {
GBufferedInputStream parent_instance;
GDataInputStreamPrivate *priv;
};
extern GDataStreamByteOrder
g_data_input_stream_get_byte_order(GDataInputStream * stream);
extern GDataStreamNewlineType
g_data_input_stream_get_newline_type(GDataInputStream * stream);
extern GType g_data_input_stream_get_type(void);
extern GDataInputStream *g_data_input_stream_new(GInputStream *
base_stream);
extern gchar g_data_input_stream_read_byte(GDataInputStream *
stream,
GCancelable * cancellable,

```

```

                                GError * *error);
extern gint16 g_data_input_stream_read_int16(GDataInputStream *
stream,
                                GCancellable * cancellable,
                                GError * *error);
extern gint32 g_data_input_stream_read_int32(GDataInputStream *
stream,
                                GCancellable * cancellable,
                                GError * *error);
extern gint64 g_data_input_stream_read_int64(GDataInputStream *
stream,
                                GCancellable * cancellable,
                                GError * *error);
extern char *g_data_input_stream_read_line(GDataInputStream *
stream,
                                gsize * length,
                                GCancellable * cancellable,
                                GError * *error);
extern void g_data_input_stream_read_line_async(GDataInputStream *
stream,
                                gint io_priority,
                                GCancellable * cancellable,
                                GAsyncReadyCallback
                                callback,
                                gpointer user_data);
extern char *g_data_input_stream_read_line_finish(GDataInputStream
*
stream,
                                GAsyncResult * result,
                                gsize * length,
                                GError * *error);
extern
                                char
*g_data_input_stream_read_line_finish_utf8(GDataInputStream *
stream,
                                GAsyncResult *
                                result,
                                gsize * length,
                                GError * *error);
extern char *g_data_input_stream_read_line_utf8(GDataInputStream *
stream,
                                gsize * length,
                                GCancellable * cancellable,
                                GError * *error);
extern guint16 g_data_input_stream_read_uint16(GDataInputStream *
stream,
                                GCancellable * cancellable,
                                GError * *error);
extern guint32 g_data_input_stream_read_uint32(GDataInputStream *
stream,
                                GCancellable * cancellable,
                                GError * *error);
extern guint64 g_data_input_stream_read_uint64(GDataInputStream *
stream,
                                GCancellable * cancellable,
                                GError * *error);
extern char *g_data_input_stream_read_until(GDataInputStream *
stream,
                                const char *stop_chars,
                                gsize * length,
                                GCancellable * cancellable,
                                GError * *error);
extern void g_data_input_stream_read_until_async(GDataInputStream
* stream,
                                const char *stop_chars,
                                gint io_priority,
                                GCancellable *

```

```

        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern                                     char
*g_data_input_stream_read_until_finish(GDataInputStream *
        stream,
        GAsyncResult * result,
        gsize * length,
        GError * *error);
extern char *g_data_input_stream_read_upto(GDataInputStream *
stream,
        const char *stop_chars,
        gssize stop_chars_len,
        gsize * length,
        Gancellable * cancellable,
        GError * *error);
extern void g_data_input_stream_read_upto_async(GDataInputStream *
stream,
        const char *stop_chars,
        gssize stop_chars_len,
        gint io_priority,
        Gancellable * cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern char *g_data_input_stream_read_upto_finish(GDataInputStream
*
        stream,
        GAsyncResult * result,
        gsize * length,
        GError * *error);
extern void g_data_input_stream_set_byte_order(GDataInputStream *
stream,
        GDataStreamByteOrder
        order);
extern void g_data_input_stream_set_newline_type(GDataInputStream
* stream,
        GDataStreamNewlineType
        type);

```

17.12.30 glib-2.0/gio/gdataoutputstream.h

```

#define                                     G_TYPE_DATA_OUTPUT_STREAM
(g_data_output_stream_get_type ())
#define                                     G_DATA_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k),                                     G_TYPE_DATA_OUTPUT_STREAM,
GDataOutputStreamClass))
#define                                     G_IS_DATA_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_DATA_OUTPUT_STREAM))
#define G_DATA_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DATA_OUTPUT_STREAM, GDataOutputStream))
#define G_IS_DATA_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE
(o), G_TYPE_DATA_OUTPUT_STREAM))
#define                                     G_DATA_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o),                                     G_TYPE_DATA_OUTPUT_STREAM,
GDataOutputStreamClass))

typedef struct _GDataOutputStream {
    GFilterOutputStream parent_instance;
    GDataOutputStreamPrivate *priv;
} GDataOutputStream;
typedef struct _GDataOutputStreamClass {
    GFilterOutputStreamClass parent_class;
    void (*_g_reserved1) (void);

```

```

    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GDataOutputStreamClass;
typedef struct _GDataOutputStreamPrivate GDataOutputStreamPrivate;
extern GDataStreamByteOrder
g_data_output_stream_get_byte_order(GDataOutputStream * stream);
extern GType g_data_output_stream_get_type(void);
extern GDataOutputStream *g_data_output_stream_new(GOutputStream *
                                                    base_stream);
extern gboolean g_data_output_stream_put_byte(GDataOutputStream *
stream,
                                              guchar data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_int16(GDataOutputStream *
stream,
                                              gint16 data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_int32(GDataOutputStream *
stream,
                                              gint32 data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_int64(GDataOutputStream *
stream,
                                              gint64 data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_string(GDataOutputStream
* stream,
                                              const char *str,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_uint16(GDataOutputStream
* stream,
                                              guint16 data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_uint32(GDataOutputStream
* stream,
                                              guint32 data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_data_output_stream_put_uint64(GDataOutputStream
* stream,
                                              guint64 data,
                                              GCancellable * cancellable,
                                              GError * *error);
extern void g_data_output_stream_set_byte_order(GDataOutputStream
* stream,
                                              GDataStreamByteOrder
order);

```

17.12.31 glib-2.0/gio/gdbusactiongroup.h

```

#define G_TYPE_DBUS_ACTION_GROUP
(g_dbus_action_group_get_type ())
#define G_DBUS_ACTION_GROUP_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_DBUS_ACTION_GROUP,
GDBusActionGroupClass))
#define G_IS_DBUS_ACTION_GROUP_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_DBUS_ACTION_GROUP))

```

```

#define G_DBUS_ACTION_GROUP(inst)          (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_DBUS_ACTION_GROUP, GDBusActionGroup))
#define G_IS_DBUS_ACTION_GROUP(inst)      (G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_DBUS_ACTION_GROUP))
#define G_DBUS_ACTION_GROUP_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_DBUS_ACTION_GROUP,
GDBusActionGroupClass))

extern GDBusActionGroup *g_dbus_action_group_get(GDBusConnection *
connection,
const gchar * bus_name,
const gchar *
object_path);
extern GType g_dbus_action_group_get_type(void);

```

17.12.32 glib-2.0/gio/gdbusaddress.h

```

extern gchar *g_dbus_address_get_for_bus_sync(GBusType bus_type,
GCancelable * cancellable,
GError * *error);
extern void g_dbus_address_get_stream(const gchar * address,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
extern GIOStream *g_dbus_address_get_stream_finish(GAsyncResult *
res,
gchar * *out_guid,
GError * *error);
extern GIOStream *g_dbus_address_get_stream_sync(const gchar *
address,
gchar * *out_guid,
GCancelable *
cancellable,
GError * *error);
extern gboolean g_dbus_is_address(const gchar * string);
extern gboolean g_dbus_is_supported_address(const gchar * string,
GError * *error);

```

17.12.33 glib-2.0/gio/gdbusauthobserver.h

```

#define G_TYPE_DBUS_AUTH_OBSERVER          (g_dbus_auth_observer_get_type ())
#define G_DBUS_AUTH_OBSERVER(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_AUTH_OBSERVER, GDBusAuthObserver))
#define G_IS_DBUS_AUTH_OBSERVER(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_DBUS_AUTH_OBSERVER))

extern gboolean
g_dbus_auth_observer_authorize_authenticated_peer(GDBusAuthObserv
er *
observer,
GIOStream * stream,
GCredentials *
credentials);
extern GType g_dbus_auth_observer_get_type(void);
extern GDBusAuthObserver *g_dbus_auth_observer_new(void);

```

17.12.34 glib-2.0/gio/gdbusconnection.h

```

#define G_TYPE_DBUS_CONNECTION (g_dbus_connection_get_type ())
#define G_DBUS_CONNECTION(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_CONNECTION, GDBusConnection))

```



```

#define G_IS_DBUS_CONNECTION(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_CONNECTION))

typedef void (*GDBusInterfaceMethodCallFunc) (GDBusConnection *
connection,
const gchar * sender,
const gchar * object_path,
const gchar * interface_name,
const gchar * method_name,
GVariant * parameters,
GDBusMethodInvocation *
invocation,
gpointer user_data);

typedef GVariant (*GDBusInterfaceGetPropertyFunc)
(GDBusConnection *
connection,
const gchar * sender,
const gchar *
object_path,
const gchar *
interface_name,
const gchar *
property_name,
GError **error,
gpointer user_data);

typedef gboolean (*GDBusInterfaceSetPropertyFunc) (GDBusConnection
*
connection,
const gchar * sender,
const gchar *
object_path,
const gchar *
interface_name,
const gchar *
property_name,
GVariant * value,
GError **error,
gpointer user_data);

struct _GDBusInterfaceVTable {
    GDBusInterfaceMethodCallFunc method_call;
    GDBusInterfaceGetPropertyFunc get_property;
    GDBusInterfaceSetPropertyFunc set_property;
    gpointer padding[8];
};

typedef gchar ** (*GDBusSubtreeEnumerateFunc) (GDBusConnection *
connection,
const gchar * sender,
const gchar * object_path,
gpointer user_data);

typedef GDBusInterfaceInfo ** (*GDBusSubtreeIntrospectFunc)
(GDBusConnection
* connection,
const gchar *
sender,
const gchar *
object_path,
const gchar *
node,
gpointer
user_data);

typedef const GDBusInterfaceVTable
* (*GDBusSubtreeDispatchFunc) (GDBusConnection * connection,
const gchar * sender,
const gchar * object_path,
const gchar * interface_name,
const gchar * node,

```

```

        gpointer * out_user_data,
        gpointer user_data);

struct _GDBusSubtreeVTable {
    GDBusSubtreeEnumerateFunc enumerate;
    GDBusSubtreeIntrospectFunc introspect;
    GDBusSubtreeDispatchFunc dispatch;
    gpointer padding[8];
};

typedef void (*GDBusSignalCallback) (GDBusConnection * connection,
                                     const gchar * sender,
                                     const gchar * object_path,
                                     const gchar * interface_name,
                                     const gchar * signal_name,
                                     GVariant * parameters,
                                     gpointer user_data);

typedef      GDBusMessage
(GDBusConnection *
                                     (*GDBusMessageFilterFunction)
                                     connection,
                                     GDBusMessage *
                                     message,
                                     gboolean incoming,
                                     gpointer user_data);

extern void g_bus_get(GBusType bus_type, GCancellable * cancellable,
                     GAsyncReadyCallback callback, gpointer
                     user_data);
extern GDBusConnection *g_bus_get_finish(GAsyncResult * res,
                                         GError * *error);
extern GDBusConnection *g_bus_get_sync(GBusType bus_type,
                                       GCancellable * cancellable,
                                       GError * *error);
extern guint g_dbus_connection_add_filter(GDBusConnection *
connection,
                                         GDBusMessageFilterFunction
                                         filter_function, void
                                         *user_data,
                                         GDestroyNotify
                                         user_data_free_func);
extern void g_dbus_connection_call(GDBusConnection * connection,
                                   const gchar * bus_name,
                                   const gchar * object_path,
                                   const gchar * interface_name,
                                   const gchar * method_name,
                                   GVariant * parameters,
                                   const GVariantType * reply_type,
                                   GDBusCallFlags flags, gint
                                   timeout_msec,
                                   GCancellable * cancellable,
                                   GAsyncReadyCallback callback,
                                   gpointer user_data);
extern GVariant *g_dbus_connection_call_finish(GDBusConnection *
connection,
                                               GAsyncResult * res,
                                               GError * *error);
extern GVariant *g_dbus_connection_call_sync(GDBusConnection *
connection,
                                              const gchar * bus_name,
                                              const gchar * object_path,
                                              const gchar * interface_name,
                                              const gchar * method_name,
                                              GVariant * parameters,
                                              const GVariantType *
                                              reply_type,
                                              GDBusCallFlags flags,
                                              gint timeout_msec,
                                              GCancellable * cancellable,
                                              GError * *error);

```

```

extern
g_dbus_connection_call_with_unix_fd_list(GDBusConnection *
                                         connection,
                                         const gchar *
                                         bus_name,
                                         const gchar *
                                         object_path,
                                         const gchar *
                                         interface_name,
                                         const gchar *
                                         method_name,
                                         GVariant * parameters,
                                         const GVariantType *
                                         reply_type,
                                         GDBusCallFlags flags,
                                         gint timeout_msec,
                                         GUnixFDList * fd_list,
                                         GCancellable *
                                         cancellable,
                                         GAsyncReadyCallback
                                         callback,
                                         gpointer user_data);

extern GVariant

*g_dbus_connection_call_with_unix_fd_list_finish(GDBusConnection *
                                                  connection,
                                                  GUnixFDList *
                                                  *out_fd_list,
                                                  GAsyncResult * res,
                                                  GError * *error);

extern GVariant
*g_dbus_connection_call_with_unix_fd_list_sync(GDBusConnection
*
                                                  connection,
                                                  const gchar * bus_name,
                                                  const gchar *
                                                  object_path,
                                                  const gchar *
                                                  interface_name,
                                                  const gchar *
                                                  method_name,
                                                  GVariant * parameters,
                                                  const GVariantType *
                                                  reply_type,
                                                  GDBusCallFlags flags,
                                                  gint timeout_msec,
                                                  GUnixFDList * fd_list,
                                                  GUnixFDList *
                                                  *out_fd_list,
                                                  GCancellable *
                                                  cancellable,
                                                  GError * *error);

extern void g_dbus_connection_close(GDBusConnection * connection,
                                    GCancellable * cancellable,
                                    GAsyncReadyCallback callback,
                                    gpointer user_data);

extern gboolean g_dbus_connection_close_finish(GDBusConnection *
                                                connection,
                                                GAsyncResult * res,
                                                GError * *error);

extern gboolean g_dbus_connection_close_sync(GDBusConnection *
connection,
                                             GCancellable * cancellable,
                                             GError * *error);

extern gboolean g_dbus_connection_emit_signal(GDBusConnection *
connection,

```

```

        const gchar *
        destination_bus_name,
        const gchar * object_path,
        const gchar * interface_name,
        const gchar * signal_name,
        GVariant * parameters,
        GError * *error);
extern void g_dbus_connection_flush(GDBusConnection * connection,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_dbus_connection_flush_finish(GDBusConnection *
        connection,
        GAsyncResult * res,
        GError * *error);
extern gboolean g_dbus_connection_flush_sync(GDBusConnection *
        connection,
        GCancellable * cancellable,
        GError * *error);
extern GDBusCapabilityFlags
g_dbus_connection_get_capabilities(GDBusConnection * connection);
extern gboolean
g_dbus_connection_get_exit_on_close(GDBusConnection *
        connection);
extern const char *g_dbus_connection_get_guid(GDBusConnection *
        connection);
extern GCredentials
*g_dbus_connection_get_peer_credentials(GDBusConnection
        * connection);
extern GIOStream *g_dbus_connection_get_stream(GDBusConnection *
        connection);
extern GType g_dbus_connection_get_type(void);
extern const char
*g_dbus_connection_get_unique_name(GDBusConnection *
        connection);
extern gboolean g_dbus_connection_is_closed(GDBusConnection *
        connection);
extern void g_dbus_connection_new(GIOStream * stream, const gchar
        * guid,
        GDBusConnectionFlags flags,
        GDBusAuthObserver * observer,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern GDBusConnection *g_dbus_connection_new_finish(GAsyncResult
        * res,
        GError * *error);
extern void g_dbus_connection_new_for_address(const gchar * address,
        GDBusConnectionFlags flags,
        GDBusAuthObserver *
        observer,
        GCancellable * cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern GDBusConnection
*g_dbus_connection_new_for_address_finish(GAsyncResult * res,
        GError * *error);
extern GDBusConnection
*g_dbus_connection_new_for_address_sync(const gchar
        * address,
        GDBusConnectionFlags
        flags,
        GDBusAuthObserver

```

```

        * observer,
        Gancellable
        *
        cancellable,
        GError *
        *error);

extern GDBusConnection *g_dbus_connection_new_sync(GIOStream *
stream,

        const gchar * guid,
        GDBusConnectionFlags
        flags,
        GDBusAuthObserver *
        observer,
        Gancellable *
        cancellable,
        GError * *error);

extern guint g_dbus_connection_register_object(GDBusConnection *
connection,
const gchar * object_path,
GDBusInterfaceInfo *
interface_info,
const GDBusInterfaceVTable

*

vtable, gpointer user_data,
GDestroyNotify
user_data_free_func,
GError * *error);

extern guint g_dbus_connection_register_subtree(GDBusConnection *
connection,
const gchar * object_path,
const GDBusSubtreeVTable *
vtable,
GDBusSubtreeFlags flags,
void *user_data,
GDestroyNotify
user_data_free_func,
GError * *error);

extern void g_dbus_connection_remove_filter(GDBusConnection *
connection,

        guint filter_id);

extern gboolean g_dbus_connection_send_message(GDBusConnection *
connection,
GDBusMessage * message,
GDBusSendMessageFlags

flags,

volatile unsigned int
*out_serial,
GError * *error);

extern void g_dbus_connection_send_message_with_reply(GDBusConnection *
connection,
GDBusMessage *
message,

GDBusSendMessageFlags

flags,
gint timeout_msec,
volatile unsigned int
*out_serial,
Gancellable *
cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);

extern GDBusMessage

```

```

*g_dbus_connection_send_message_with_reply_finish(GDBusConnection
*
                                                    connection,
                                                    GAsyncResult * res,
                                                    GError * *error);

extern GDBusMessage

*g_dbus_connection_send_message_with_reply_sync(GDBusConnection *
                                                    connection,
                                                    GDBusMessage * message,
                                                    GDBusSendMessageFlags
                                                    flags,
                                                    gint timeout_msec,
                                                    volatile unsigned int
                                                    *out_serial,
                                                    GCancellable *
                                                    cancellable,
                                                    GError * *error);

extern void g_dbus_connection_set_exit_on_close(GDBusConnection *
                                                    connection,
                                                    gboolean exit_on_close);

extern guint g_dbus_connection_signal_subscribe(GDBusConnection *
                                                    connection,
                                                    const gchar * sender,
                                                    const gchar *
                                                    interface_name,
                                                    const gchar * member,
                                                    const gchar * object_path,
                                                    const gchar * arg0,
                                                    GDBusSignalFlags flags,
                                                    GDBusSignalCallback
                                                    callback, void *user_data,
                                                    GDestroyNotify
                                                    user_data_free_func);

extern void g_dbus_connection_signal_unsubscribe(GDBusConnection *
                                                    connection,
                                                    guint subscription_id);

extern
                                                    void
g_dbus_connection_start_message_processing(GDBusConnection *
                                                    connection);

extern
                                                    gboolean
g_dbus_connection_unregister_object(GDBusConnection *
                                                    connection,
                                                    guint registration_id);

extern
                                                    gboolean
g_dbus_connection_unregister_subtree(GDBusConnection *
                                                    connection,
                                                    guint
                                                    registration_id);

```

17.12.35 glib-2.0/gio/gdbuserror.h

```

#define G_DBUS_ERROR    g_dbus_error_quark()

struct _GDBusErrorEntry {
    gint error_code;
    const gchar *dbus_error_name;
};

extern gchar *g_dbus_error_encode_gerror(const GError * error);
extern gchar *g_dbus_error_get_remote_error(const GError * error);
extern gboolean g_dbus_error_is_remote_error(const GError * error);
extern GError *g_dbus_error_new_for_dbus_error(const gchar *
                                                    dbus_error_name,
                                                    const gchar *

```

```

        dbus_error_message);
extern GQuark g_dbus_error_quark(void);
extern gboolean g_dbus_error_register_error(GQuark error_domain,
        gint error_code,
        const gchar *
        dbus_error_name);
extern void g_dbus_error_register_error_domain(const gchar *
        error_domain_quark_name,
        volatile gsize *
        quark_volatile,
        const GDBusErrorEntry *
        entries, guint num_entries);
extern void g_dbus_error_set_dbus_error(GError * *error,
        const gchar * dbus_error_name,
        const gchar * dbus_error_message,
        const gchar * format, ...);
extern void g_dbus_error_set_dbus_error_valist(GError * *error,
        const gchar *
        dbus_error_name,
        const gchar *
        dbus_error_message,
        const gchar * format,
        va_list var_args);
extern gboolean g_dbus_error_strip_remote_error(GError * error);
extern gboolean g_dbus_error_unregister_error(GQuark error_domain,
        gint error_code,
        const gchar *
        dbus_error_name);

```

17.12.36 glib-2.0/gio/gdbusinterface.h

```

#define G_TYPE_DBUS_INTERFACE (g_dbus_interface_get_type())
#define G_DBUS_INTERFACE(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_INTERFACE, GDBusInterface))
#define G_IS_DBUS_INTERFACE(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_INTERFACE))
#define G_DBUS_INTERFACE_GET_IFACE(o)
(G_TYPE_INSTANCE_GET_INTERFACE((o), G_TYPE_DBUS_INTERFACE,
GDBusInterfaceIface))

typedef struct GDBusInterfaceIface {
    GTypeInterface parent_iface;
    GDBusInterfaceInfo *(*get_info) (GDBusInterface * interface_);
    GDBusObject *(*get_object) (GDBusInterface * interface_);
    void (*set_object) (GDBusInterface * interface_, GDBusObject *
object);
    GDBusObject *(*dup_object) (GDBusInterface * interface_);
} GDBusInterfaceIface;
extern GDBusObject *g_dbus_interface_dup_object(GDBusInterface *
interface_);
extern GDBusInterfaceInfo
g_dbus_interface_get_info(GDBusInterface *
interface_);
extern GDBusObject *g_dbus_interface_get_object(GDBusInterface *
interface_);
extern GType g_dbus_interface_get_type(void);
extern void g_dbus_interface_set_object(GDBusInterface *
interface_,
GDBusObject * object);

```

17.12.37 glib-2.0/gio/gdbusinterfaceskeleton.h

```

#define G_TYPE_DBUS_INTERFACE_SKELETON
(g_dbus_interface_skeleton_get_type ())
#define G_DBUS_INTERFACE_SKELETON_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DBUS_INTERFACE_SKELETON,
GDBusInterfaceSkeletonClass))
#define G_IS_DBUS_INTERFACE_SKELETON_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_DBUS_INTERFACE_SKELETON))
#define G_DBUS_INTERFACE_SKELETON(o)
(G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_DBUS_INTERFACE_SKELETON,
GDBusInterfaceSkeleton))
#define G_IS_DBUS_INTERFACE_SKELETON(o)
(G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_DBUS_INTERFACE_SKELETON))
#define G_DBUS_INTERFACE_SKELETON_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_DBUS_INTERFACE_SKELETON,
GDBusInterfaceSkeletonClass))

typedef struct _GDBusInterfaceSkeletonClass {
    GObjectClass parent_class;
    GDBusInterfaceInfo *(*get_info) (GDBusInterfaceSkeleton *
interface_);
    GDBusInterfaceVTable *(*get_vtable) (GDBusInterfaceSkeleton *
interface_);
    GVariant *(*get_properties) (GDBusInterfaceSkeleton *
interface_);
    void (*flush) (GDBusInterfaceSkeleton * interface_);
    gpointer vfunc_padding[8];
    gboolean(*g_authorize_method) (GDBusInterfaceSkeleton *
interface_,
GDBusMethodInvocation * invocation);
    gpointer signal_padding[8];
} GDBusInterfaceSkeletonClass;
typedef struct _GDBusInterfaceSkeletonPrivate
GDBusInterfaceSkeletonPrivate;
struct _GDBusInterfaceSkeleton {
    GObject parent_instance;
    GDBusInterfaceSkeletonPrivate *priv;
};
extern gboolean
g_dbus_interface_skeleton_export(GDBusInterfaceSkeleton *
interface_,
GDBusConnection *
connection,
const gchar * object_path,
GError * *error);
extern void g_dbus_interface_skeleton_flush(GDBusInterfaceSkeleton *
interface_);
extern GDBusConnection
*g_dbus_interface_skeleton_get_connection(GDBusInterfaceSkeleton *
interface_);
extern GList
*g_dbus_interface_skeleton_get_connections(GDBusInterfaceSkeleton *
interface_);
extern GDBusInterfaceSkeletonFlags
g_dbus_interface_skeleton_get_flags(GDBusInterfaceSkeleton *
interface_);
extern GDBusInterfaceInfo
*g_dbus_interface_skeleton_get_info(GDBusInterfaceSkeleton *
interface_);
extern const char
*g_dbus_interface_skeleton_get_object_path(GDBusInterfaceSkeleton *

```



```

        interface_);

extern GVariant

g_dbus_interface_skeleton_get_properties(GDBusInterfaceSkeleton *
        interface_);
extern GType g_dbus_interface_skeleton_get_type(void);
extern GDBusInterfaceVTable
        *g_dbus_interface_skeleton_get_vtable(GDBusInterfaceSkeleton *
        interface_);

extern gboolean
g_dbus_interface_skeleton_has_connection(GDBusInterfaceSkeleton *
        interface_,
        GDBusConnection * connection);

extern void
g_dbus_interface_skeleton_set_flags(GDBusInterfaceSkeleton *
        interface_,

GDBusInterfaceSkeletonFlags

        flags);

extern void
g_dbus_interface_skeleton_unexport(GDBusInterfaceSkeleton *
        interface_);

extern void
g_dbus_interface_skeleton_unexport_from_connection(GDBusInterface
Skeleton *
        interface_,
        GDBusConnection *
        connection);

```

17.12.38 glib-2.0/gio/gdbusintrospection.h

```

#define G_TYPE_DBUS_ANNOTATION_INFO
(g_dbus_annotation_info_get_type ())
#define G_TYPE_DBUS_ARG_INFO (g_dbus_arg_info_get_type ())
#define G_TYPE_DBUS_INTERFACE_INFO
(g_dbus_interface_info_get_type ())
#define G_TYPE_DBUS_METHOD_INFO (g_dbus_method_info_get_type ())
#define G_TYPE_DBUS_NODE_INFO (g_dbus_node_info_get_type ())
#define G_TYPE_DBUS_PROPERTY_INFO
(g_dbus_property_info_get_type ())
#define G_TYPE_DBUS_SIGNAL_INFO (g_dbus_signal_info_get_type ())

struct _GDBusAnnotationInfo {
    volatile gint ref_count;
    gchar *key;
    gchar *value;
    GDBusAnnotationInfo **annotations;
};
struct _GDBusArgInfo {
    volatile gint ref_count;
    gchar *name;
    gchar *signature;
    GDBusAnnotationInfo **annotations;
};
struct _GDBusMethodInfo {
    volatile gint ref_count;
    gchar *name;
    GDBusArgInfo **in_args;
    GDBusArgInfo **out_args;
    GDBusAnnotationInfo **annotations;
};
struct _GDBusSignalInfo {
    volatile gint ref_count;
    gchar *name;
    GDBusArgInfo **args;
};

```

```

        GDBusAnnotationInfo **annotations;
    };
    struct _GDBusPropertyInfo {
        volatile gint ref_count;
        gchar *name;
        gchar *signature;
        GDBusPropertyInfoFlags flags;
        GDBusAnnotationInfo **annotations;
    };
    struct _GDBusInterfaceInfo {
        volatile gint ref_count;
        gchar *name;
        GDBusMethodInfo **methods;
        GDBusSignalInfo **signals;
        GDBusPropertyInfo **properties;
        GDBusAnnotationInfo **annotations;
    };
    struct _GDBusNodeInfo {
        volatile gint ref_count;
        gchar *path;
        GDBusInterfaceInfo **interfaces;
        GDBusNodeInfo **nodes;
        GDBusAnnotationInfo **annotations;
    };
    extern GType g_dbus_annotation_info_get_type(void);
    extern const char
    *g_dbus_annotation_info_lookup(GDBusAnnotationInfo *
                                   *annotations,
                                   const gchar * name);
    extern GDBusAnnotationInfo
    *g_dbus_annotation_info_ref(GDBusAnnotationInfo
                                * info);
    extern void g_dbus_annotation_info_unref(GDBusAnnotationInfo *
    info);
    extern GType g_dbus_arg_info_get_type(void);
    extern GDBusArgInfo *g_dbus_arg_info_ref(GDBusArgInfo * info);
    extern void g_dbus_arg_info_unref(GDBusArgInfo * info);
    extern void g_dbus_interface_info_cache_build(GDBusInterfaceInfo *
    info);
    extern void g_dbus_interface_info_cache_release(GDBusInterfaceInfo
    * info);
    extern void g_dbus_interface_info_generate_xml(GDBusInterfaceInfo
    * info,
                                                    guint indent,
                                                    GString * string_builder);
    extern GType g_dbus_interface_info_get_type(void);
    extern GDBusMethodInfo
    *g_dbus_interface_info_lookup_method(GDBusInterfaceInfo * info,
                                         const gchar * name);
    extern GDBusPropertyInfo
    *g_dbus_interface_info_lookup_property(GDBusInterfaceInfo *
    info,
                                         const gchar * name);
    extern GDBusSignalInfo
    *g_dbus_interface_info_lookup_signal(GDBusInterfaceInfo * info,
                                         const gchar * name);
    extern GDBusInterfaceInfo
    *g_dbus_interface_info_ref(GDBusInterfaceInfo *
                                info);
    extern void g_dbus_interface_info_unref(GDBusInterfaceInfo * info);
    extern GType g_dbus_method_info_get_type(void);
    extern GDBusMethodInfo *g_dbus_method_info_ref(GDBusMethodInfo *
    info);
    extern void g_dbus_method_info_unref(GDBusMethodInfo * info);
    extern void g_dbus_node_info_generate_xml(GDBusNodeInfo * info,
                                              guint indent,

```

```

        GString * string_builder);
extern GType g_dbus_node_info_get_type(void);
extern GDBusInterfaceInfo
*g_dbus_node_info_lookup_interface(GDBusNodeInfo
        * info,
        const gchar *
        name);
extern GDBusNodeInfo *g_dbus_node_info_new_for_xml(const gchar *
xml_data,
        GError * *error);
extern GDBusNodeInfo *g_dbus_node_info_ref(GDBusNodeInfo * info);
extern void g_dbus_node_info_unref(GDBusNodeInfo * info);
extern GType g_dbus_property_info_get_type(void);
extern GDBusPropertyInfo
*g_dbus_property_info_ref(GDBusPropertyInfo *
        info);
extern void g_dbus_property_info_unref(GDBusPropertyInfo * info);
extern GType g_dbus_signal_info_get_type(void);
extern GDBusSignalInfo *g_dbus_signal_info_ref(GDBusSignalInfo *
info);
extern void g_dbus_signal_info_unref(GDBusSignalInfo * info);

```

17.12.39 glib-2.0/gio/gdbusmenumodel.h

```

#define G_TYPE_DBUS_MENU_MODEL (g_dbus_menu_model_get_type ())
#define G_DBUS_MENU_MODEL(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_DBUS_MENU_MODEL, GDBusMenuModel))
#define G_IS_DBUS_MENU_MODEL(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_DBUS_MENU_MODEL))

typedef struct _GDBusMenuModel GDBusMenuModel;
extern GDBusMenuModel *g_dbus_menu_model_get(GDBusConnection *
connection,
        const gchar * bus_name,
        const gchar * object_path);
extern GType g_dbus_menu_model_get_type(void);

```

17.12.40 glib-2.0/gio/gdbusmessage.h

```

#define G_TYPE_DBUS_MESSAGE (g_dbus_message_get_type ())
#define G_DBUS_MESSAGE(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_MESSAGE, GDBusMessage))
#define G_IS_DBUS_MESSAGE(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_MESSAGE))

extern gssize g_dbus_message_bytes_needed(guchar * blob, gsize
blob_len,
        GError * *error);
extern GDBusMessage *g_dbus_message_copy(GDBusMessage * message,
        GError * *error);
extern const char *g_dbus_message_get_arg0(GDBusMessage * message);
extern GVariant *g_dbus_message_get_body(GDBusMessage * message);
extern GDBusMessageByteOrder
g_dbus_message_get_byte_order(GDBusMessage *
        message);
extern const char *g_dbus_message_get_destination(GDBusMessage *
message);
extern const char *g_dbus_message_get_error_name(GDBusMessage *
message);
extern GDBusMessageFlags g_dbus_message_get_flags(GDBusMessage *
message);
extern GVariant *g_dbus_message_get_header(GDBusMessage * message,
        GDBusMessageHeaderField

```

```

                                header_field);
extern guchar *g_dbus_message_get_header_fields(GDBusMessage *
message);
extern const char *g_dbus_message_get_interface(GDBusMessage *
message);
extern gboolean g_dbus_message_get_locked(GDBusMessage * message);
extern const char *g_dbus_message_get_member(GDBusMessage *
message);
extern
                                GDBusMessageType
g_dbus_message_get_message_type(GDBusMessage *
                                message);
extern guint32 g_dbus_message_get_num_unix_fds(GDBusMessage *
message);
extern const char *g_dbus_message_get_path(GDBusMessage * message);
extern guint32 g_dbus_message_get_reply_serial(GDBusMessage *
message);
extern const char *g_dbus_message_get_sender(GDBusMessage *
message);
extern guint32 g_dbus_message_get_serial(GDBusMessage * message);
extern const char *g_dbus_message_get_signature(GDBusMessage *
message);
extern GType g_dbus_message_get_type(void);
extern GUnixFDList *g_dbus_message_get_unix_fd_list(GDBusMessage *
                                message);
extern void g_dbus_message_lock(GDBusMessage * message);
extern GDBusMessage *g_dbus_message_new(void);
extern GDBusMessage *g_dbus_message_new_from_blob(guchar * blob,
                                gsize blob_len,
                                GDBusCapabilityFlags
                                capabilities,
                                GError * *error);
extern GDBusMessage *g_dbus_message_new_method_call(const gchar *
name,
                                const gchar * path,
                                const gchar *
                                interface_,
                                const gchar * method);
extern GDBusMessage *g_dbus_message_new_method_error(GDBusMessage
*
                                method_call_message,
                                const gchar *
                                error_name,
                                const gchar *
                                error_message_format,
                                ...);
extern
                                GDBusMessage
*g_dbus_message_new_method_error_literal(GDBusMessage *
                                method_call_message,
                                const gchar *
                                error_name,
                                const gchar *
                                error_message);
extern
                                GDBusMessage
*g_dbus_message_new_method_error_valist(GDBusMessage *
                                method_call_message,
                                const gchar *
                                error_name,
                                const gchar *
                                error_message_format,
                                va_list
                                var_args);
extern GDBusMessage *g_dbus_message_new_method_reply(GDBusMessage
*

```

```

        method_call_message);
extern GDBusMessage *g_dbus_message_new_signal(const gchar * path,
        const gchar * interface_,
        const gchar * signal);
extern gchar *g_dbus_message_print(GDBusMessage * message, guint
indent);
extern void g_dbus_message_set_body(GDBusMessage * message,
        GVariant * body);
extern void g_dbus_message_set_byte_order(GDBusMessage * message,
        GDBusMessageByteOrder
        byte_order);
extern void g_dbus_message_set_destination(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_error_name(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_flags(GDBusMessage * message,
        GDBusMessageFlags flags);
extern void g_dbus_message_set_header(GDBusMessage * message,
        GDBusMessageHeaderField
header_field,
        GVariant * value);
extern void g_dbus_message_set_interface(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_member(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_message_type(GDBusMessage * message,
        GDBusMessageType type);
extern void g_dbus_message_set_num_unix_fds(GDBusMessage * message,
        guint32 value);
extern void g_dbus_message_set_path(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_reply_serial(GDBusMessage * message,
        guint32 value);
extern void g_dbus_message_set_sender(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_serial(GDBusMessage * message,
        guint32 serial);
extern void g_dbus_message_set_signature(GDBusMessage * message,
        const gchar * value);
extern void g_dbus_message_set_unix_fd_list(GDBusMessage * message,
        GUnixFDList * fd_list);
extern gchar *g_dbus_message_to_blob(GDBusMessage * message,
        gsize * out_size,
        GDBusCapabilityFlags capabilities,
        GError * *error);
extern gboolean g_dbus_message_to_gerror(GDBusMessage * message,
        GError * *error);

```

17.12.41 glib-2.0/gio/gdbusmethodinvocation.h

```

#define G_TYPE_DBUS_METHOD_INVOCATION
(g_dbus_method_invocation_get_type ())
#define G_DBUS_METHOD_INVOCATION(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_DBUS_METHOD_INVOCATION,
GDBusMethodInvocation))
#define G_IS_DBUS_METHOD_INVOCATION(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_DBUS_METHOD_INVOCATION))

extern GDBusConnection
*g_dbus_method_invocation_get_connection(GDBusMethodInvocation
*
        invocation);
extern const char

```

```

*g_dbus_method_invocation_get_interface_name(GDBusMethodInvocation
n *
                                invocation);

extern GDBusMessage
    *g_dbus_method_invocation_get_message(GDBusMethodInvocation *
                                invocation);

extern const GDBusMethodInfo

*g_dbus_method_invocation_get_method_info(GDBusMethodInvocation *
                                invocation);

extern const char

*g_dbus_method_invocation_get_method_name(GDBusMethodInvocation *
                                invocation);

extern const char

*g_dbus_method_invocation_get_object_path(GDBusMethodInvocation *
                                invocation);

extern GVariant
    *g_dbus_method_invocation_get_parameters(GDBusMethodInvocation
*
                                invocation);

extern const char
    *g_dbus_method_invocation_get_sender(GDBusMethodInvocation *
                                invocation);

extern GType g_dbus_method_invocation_get_type(void);
extern gpointer
g_dbus_method_invocation_get_user_data(GDBusMethodInvocation *
invocation);
extern void
g_dbus_method_invocation_return_dbus_error(GDBusMethodInvocation *
invocation,
const gchar * error_name,
const gchar * error_message);

extern
void
g_dbus_method_invocation_return_error(GDBusMethodInvocation *
invocation,
GQuark domain, gint code,
const gchar * format,
...);

extern void
g_dbus_method_invocation_return_error_literal(GDBusMethodInvocati
on *
                                invocation, GQuark domain,
                                gint code,
                                const gchar * message);

extern void
g_dbus_method_invocation_return_error_valist(GDBusMethodInvocatio
n *
                                invocation, GQuark domain,
                                gint code,
                                const gchar * format,
                                va_list var_args);

extern
void
g_dbus_method_invocation_return_gerror(GDBusMethodInvocation *
invocation,
const GError * error);

extern
void
g_dbus_method_invocation_return_value(GDBusMethodInvocation *
invocation,
GVariant * parameters);

extern void
g_dbus_method_invocation_return_value_with_unix_fd_list
(GDBusMethodInvocation * invocation, GVariant * parameters,
GUnixFDList * fd_list);

```

```

extern void
g_dbus_method_invocation_take_error(GDBusMethodInvocation *
                                     invocation,
                                     GError * error);

```

17.12.42 glib-2.0/gio/gdbusnameowning.h

```

typedef void (*GDBusAcquiredCallback) (GDBusConnection * connection,
                                       const gchar * name,
                                       gpointer user_data);
typedef void (*GDBusNameAcquiredCallback) (GDBusConnection *
connection,
                                       const gchar * name,
                                       gpointer user_data);
typedef void (*GDBusNameLostCallback) (GDBusConnection * connection,
                                       const gchar * name,
                                       gpointer user_data);
extern guint g_bus_own_name(GBusType bus_type, const gchar * name,
                           GBusNameOwnerFlags flags,
                           GDBusAcquiredCallback bus_acquired_handler,
                           GDBusNameAcquiredCallback
name_acquired_handler,
                           GDBusNameLostCallback name_lost_handler,
                           gpointer user_data,
                           GDestroyNotify user_data_free_func);
extern guint g_bus_own_name_on_connection(GDBusConnection *
connection,
                                       const gchar * name,
                                       GBusNameOwnerFlags flags,
                                       GDBusNameAcquiredCallback
name_acquired_handler,
                                       GDBusNameLostCallback
name_lost_handler,
                                       gpointer user_data,
                                       GDestroyNotify
user_data_free_func);
extern
g_bus_own_name_on_connection_with_closures(GDBusConnection *
connection,
                                       const gchar * name,
                                       GBusNameOwnerFlags
flags,
                                       GClosure *
name_acquired_closure,
                                       GClosure *
name_lost_closure);
extern guint g_bus_own_name_with_closures(GBusType bus_type,
                                       const gchar * name,
                                       GBusNameOwnerFlags flags,
                                       GClosure * bus_acquired_closure,
                                       GClosure *
name_acquired_closure,
                                       GClosure * name_lost_closure);
extern void g_bus_unown_name(guint owner_id);

```

17.12.43 glib-2.0/gio/gdbusnamewatching.h

```

typedef void (*GDBusNameAppearedCallback) (GDBusConnection *
connection,
                                       const gchar * name,
                                       const gchar * name_owner,
                                       gpointer user_data);

```

```

typedef void (*GBusNameVanishedCallback) (GDBusConnection *
connection,
                                         const gchar * name,
                                         gpointer user_data);
extern void g_bus_unwatch_name(guint watcher_id);
extern guint g_bus_watch_name(GBusType bus_type, const gchar * name,
                              GBusNameWatcherFlags flags,
                              GBusNameAppearedCallback
                              name_appeared_handler,
                              GBusNameVanishedCallback
                              name_vanished_handler, gpointer user_data,
                              GDestroyNotify user_data_free_func);
extern guint g_bus_watch_name_on_connection(GDBusConnection *
connection,
                                         const gchar * name,
                                         GBusNameWatcherFlags flags,
                                         GBusNameAppearedCallback
                                         name_appeared_handler,
                                         GBusNameVanishedCallback
                                         name_vanished_handler,
                                         gpointer user_data,
                                         GDestroyNotify
                                         user_data_free_func);

extern
g_bus_watch_name_on_connection_with_closures(GDBusConnection *
connection,
const gchar *
name,
GBusNameWatcherFlags
flags,
GClosure *
name_appeared_closure,
GClosure *
name_vanished_closure);
extern guint g_bus_watch_name_with_closures(GBusType bus_type,
const gchar * name,
GBusNameWatcherFlags flags,
GClosure *
name_appeared_closure,
GClosure *
name_vanished_closure);

```

17.12.44 glib-2.0/gio/gdbusobject.h

```

#define G_TYPE_DBUS_OBJECT (g_dbus_object_get_type())
#define G_DBUS_OBJECT(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_OBJECT, GDBusObject))
#define G_IS_DBUS_OBJECT(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_OBJECT))
#define G_DBUS_OBJECT_GET_IFACE(o)
(G_TYPE_INSTANCE_GET_INTERFACE((o), G_TYPE_DBUS_OBJECT,
GDBusObjectIface))

typedef struct _GDBusObjectIface {
  GTypeInterface parent_iface;
  const gchar *(*get_object_path) (GDBusObject * object);
  GList *(*get_interfaces) (GDBusObject * object);
  GDBusInterface *(*get_interface) (GDBusObject * object,
const gchar * interface_name);
  void (*interface_added) (GDBusObject * object,
GDBusInterface * interface_);
  void (*interface_removed) (GDBusObject * object,

```



```

        GDBusInterface * interface_);
} GDBusObjectIface;
extern GDBusInterface *g_dbus_object_get_interface(GDBusObject *
object,
        const gchar *
        interface_name);
extern GList *g_dbus_object_get_interfaces(GDBusObject * object);
extern const char *g_dbus_object_get_object_path(GDBusObject *
object);
extern GType g_dbus_object_get_type(void);

```

17.12.45 glib-2.0/gio/gdbusobjectmanager.h

```

#define G_TYPE_DBUS_OBJECT_MANAGER (G_TYPE_CHECK_INSTANCE_CAST
(g_dbus_object_manager_get_type())
#define G_DBUS_OBJECT_MANAGER(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_DBUS_OBJECT_MANAGER, GDBusObjectManager))
#define G_IS_DBUS_OBJECT_MANAGER(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_DBUS_OBJECT_MANAGER))
#define G_DBUS_OBJECT_MANAGER_GET_IFACE(o) (G_TYPE_INSTANCE_GET_INTERFACE((o), G_TYPE_DBUS_OBJECT_MANAGER,
GDBusObjectManagerInterface))

typedef struct _GDBusObjectManagerInterface {
    GTypeInterface parent_iface;
    const gchar *(*get_object_path) (GDBusObjectManager * manager);
    GList *(*get_objects) (GDBusObjectManager * manager);
    GDBusObject *(*get_object) (GDBusObjectManager * manager,
        const gchar * object_path);
    GDBusInterface *(*get_interface) (GDBusObjectManager * manager,
        const gchar * object_path,
        const gchar * interface_name);
    void (*object_added) (GDBusObjectManager * manager,
        GDBusObject * object);
    void (*object_removed) (GDBusObjectManager * manager,
        GDBusObject * object);
    void (*interface_added) (GDBusObjectManager * manager,
        GDBusObject * object,
        GDBusInterface * interface_);
    void (*interface_removed) (GDBusObjectManager * manager,
        GDBusObject * object,
        GDBusInterface * interface_);
} GDBusObjectManagerInterface;
extern GDBusInterface
    *g_dbus_object_manager_get_interface(GDBusObjectManager *
manager,
        const gchar * object_path,
        const gchar * interface_name);
extern
    GDBusObject
    *g_dbus_object_manager_get_object(GDBusObjectManager *
manager,
        const gchar *
        object_path);
extern
    const
    char
    *g_dbus_object_manager_get_object_path(GDBusObjectManager
        * manager);
extern GList *g_dbus_object_manager_get_objects(GDBusObjectManager
    *
        manager);
extern GType g_dbus_object_manager_get_type(void);

```

17.12.46 glib-2.0/gio/gdbusobjectmanagerclient.h

```

#define G_TYPE_DBUS_OBJECT_MANAGER_CLIENT
(g_dbus_object_manager_client_get_type ())
#define G_DBUS_OBJECT_MANAGER_CLIENT_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DBUS_OBJECT_MANAGER_CLIENT,
GDBusObjectManagerClientClass))
#define G_IS_DBUS_OBJECT_MANAGER_CLIENT_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_DBUS_OBJECT_MANAGER_CLIENT))
#define G_DBUS_OBJECT_MANAGER_CLIENT(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_OBJECT_MANAGER_CLIENT, GDBusObjectManagerClient))
#define G_IS_DBUS_OBJECT_MANAGER_CLIENT(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_OBJECT_MANAGER_CLIENT))
#define G_DBUS_OBJECT_MANAGER_CLIENT_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_DBUS_OBJECT_MANAGER_CLIENT,
GDBusObjectManagerClientClass))

typedef struct _GDBusObjectManagerClientClass {
    GObjectClass parent_class;
    void (*interface_proxy_signal) (GDBusObjectManagerClient *
manager,
                                GDBusObjectProxy * object_proxy,
                                GDBusProxy * interface_proxy,
                                const gchar * sender_name,
                                const gchar * signal_name,
                                GVariant * parameters);
    void (*interface_proxy_properties_changed)
(GDBusObjectManagerClient *
manager,
GDBusObjectProxy *
object_proxy,
GDBusProxy *
interface_proxy,
GVariant *
changed_properties,
const gchar *
const
*invalidated_properties);
    gpointer padding[8];
} GDBusObjectManagerClientClass;
typedef struct _GDBusObjectManagerClientPrivate
GDBusObjectManagerClientPrivate;
struct _GDBusObjectManagerClient {
    GObject parent_instance;
    GDBusObjectManagerClientPrivate *priv;
};
extern GDBusConnection
*g_dbus_object_manager_client_get_connection(GDBusObjectManagerCl
ient *
manager);
extern GDBusObjectManagerClientFlags
g_dbus_object_manager_client_get_flags(GDBusObjectManagerClient *
manager);
extern const char
*g_dbus_object_manager_client_get_name(GDBusObjectManagerClient *
manager);
extern gchar
*g_dbus_object_manager_client_get_name_owner(GDBusObjectManagerCl
ient *
manager);
extern GType g_dbus_object_manager_client_get_type(void);
extern void g_dbus_object_manager_client_new(GDBusConnection *
connection,

```

```

GDBusObjectManagerClientFlags

flags, const gchar * name,
const gchar * object_path,
GDBusProxyTypeFunc
get_proxy_type_func,
gpointer
get_proxy_type_user_data,
GDestroyNotify

get_proxy_type_destroy_notify,

GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);

extern GDBusObjectManager
    *g_dbus_object_manager_client_new_finish(GAsyncResult * res,
                                              GError * *error);
extern void g_dbus_object_manager_client_new_for_bus(GBusType
bus_type,

GDBusObjectManagerClientFlags

flags,
const gchar * name,
const gchar *
object_path,
GDBusProxyTypeFunc
get_proxy_type_func,
gpointer

get_proxy_type_user_data,

GDestroyNotify

get_proxy_type_destroy_notify,

GCancelable *
cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);

extern GDBusObjectManager
    *g_dbus_object_manager_client_new_for_bus_finish(GAsyncResult
* res,

GError * *error);
extern GDBusObjectManager
    *g_dbus_object_manager_client_new_for_bus_sync(GBusType
bus_type,

GDBusObjectManagerClientFlags

flags,
const gchar * name,
const gchar *
object_path,
GDBusProxyTypeFunc
get_proxy_type_func,
gpointer

get_proxy_type_user_data,

GDestroyNotify

get_proxy_type_destroy_notify,

GCancelable *
cancellable,
GError * *error);

extern GDBusObjectManager
    *g_dbus_object_manager_client_new_sync(GDBusConnection *
connection,

GDBusObjectManagerClientFlags

```

```

flags, const gchar * name,
const gchar * object_path,
GDBusProxyTypeFunc
get_proxy_type_func,
gpointer
get_proxy_type_user_data,
GDestroyNotify
get_proxy_type_destroy_notify,
GCancelable * cancellable,
GError * *error);

```

17.12.47 glib-2.0/gio/gdbusobjectmanagerserver.h

```

#define G_TYPE_DBUS_OBJECT_MANAGER_SERVER (g_dbus_object_manager_server_get_type ())
#define G_DBUS_OBJECT_MANAGER_SERVER_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DBUS_OBJECT_MANAGER_SERVER, GDBusObjectManagerServerClass))
#define G_IS_DBUS_OBJECT_MANAGER_SERVER_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_DBUS_OBJECT_MANAGER_SERVER))
#define G_DBUS_OBJECT_MANAGER_SERVER(o) (G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_DBUS_OBJECT_MANAGER_SERVER, GDBusObjectManagerServer))
#define G_IS_DBUS_OBJECT_MANAGER_SERVER(o) (G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_DBUS_OBJECT_MANAGER_SERVER))
#define G_DBUS_OBJECT_MANAGER_SERVER_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_DBUS_OBJECT_MANAGER_SERVER, GDBusObjectManagerServerClass))

typedef struct _GDBusObjectManagerServerClass {
    GObjectClass parent_class;
    gpointer padding[8];
} GDBusObjectManagerServerClass;
typedef struct _GDBusObjectManagerServerPrivate {
    GDBusObjectManagerServerPrivate;
} GDBusObjectManagerServerPrivate;
struct _GDBusObjectManagerServer {
    GObject parent_instance;
    GDBusObjectManagerServerPrivate *priv;
};

extern void g_dbus_object_manager_server_export(GDBusObjectManagerServer *
manager,
GDBusObjectSkeleton *
object);

extern void g_dbus_object_manager_server_export_uniquely(GDBusObjectManagerSe
rver *
manager,
GDBusObjectSkeleton *
object);

extern GDBusConnection *g_dbus_object_manager_server_get_connection(GDBusObjectManagerSe
rver *
manager);

extern GType g_dbus_object_manager_server_get_type(void);
extern GDBusObjectManagerServer *g_dbus_object_manager_server_new(const
gchar *
object_path);
extern void g_dbus_object_manager_server_set_connection(GDBusObjectManagerSer
ver *

```

```

manager,
GDBusConnection * connection);

extern gboolean
g_dbus_object_manager_server_unexport(GDBusObjectManagerServer *
manager,

const gchar * object_path);

```

17.12.48 glib-2.0/gio/gdbusobjectproxy.h

```

#define G_TYPE_DBUS_OBJECT_PROXY
(g_dbus_object_proxy_get_type ())
#define G_DBUS_OBJECT_PROXY_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DBUS_OBJECT_PROXY,
GDBusObjectProxyClass))
#define G_IS_DBUS_OBJECT_PROXY_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_DBUS_OBJECT_PROXY))
#define G_DBUS_OBJECT_PROXY(o) (G_TYPE_CHECK_INSTANCE_CAST((o),
G_TYPE_DBUS_OBJECT_PROXY, GDBusObjectProxy))
#define G_IS_DBUS_OBJECT_PROXY(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_DBUS_OBJECT_PROXY))
#define G_DBUS_OBJECT_PROXY_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_DBUS_OBJECT_PROXY,
GDBusObjectProxyClass))

typedef struct _GDBusObjectProxyClass {
GObjectClass parent_class;
gpointer padding[8];
} GDBusObjectProxyClass;
typedef struct _GDBusObjectProxyPrivate GDBusObjectProxyPrivate;
struct _GDBusObjectProxy {
GObject parent_instance;
GDBusObjectProxyPrivate *priv;
};

extern GDBusConnection
*g_dbus_object_proxy_get_connection(GDBusObjectProxy
* proxy);

extern GType g_dbus_object_proxy_get_type(void);
extern GDBusObjectProxy *g_dbus_object_proxy_new(GDBusConnection *
connection,
const gchar *
object_path);

```

17.12.49 glib-2.0/gio/gdbusobjectskeleton.h

```

#define G_TYPE_DBUS_OBJECT_SKELETON
(g_dbus_object_skeleton_get_type ())
#define G_DBUS_OBJECT_SKELETON_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_DBUS_OBJECT_SKELETON,
GDBusObjectSkeletonClass))
#define G_IS_DBUS_OBJECT_SKELETON_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_DBUS_OBJECT_SKELETON))
#define G_DBUS_OBJECT_SKELETON(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_DBUS_OBJECT_SKELETON, GDBusObjectSkeleton))
#define G_IS_DBUS_OBJECT_SKELETON(o)
(G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_DBUS_OBJECT_SKELETON))
#define G_DBUS_OBJECT_SKELETON_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_DBUS_OBJECT_SKELETON,
GDBusObjectSkeletonClass))

typedef struct _GDBusObjectSkeletonClass {
GObjectClass parent_class;
gboolean(*authorize_method)(GDBusObjectSkeleton * object,
GDBusInterfaceSkeleton * interface_,

```

```

        GDBusMethodInvocation * invocation);

    gpointer padding[8];
} GDBusObjectSkeletonClass;
typedef struct _GDBusObjectSkeletonPrivate
GDBusObjectSkeletonPrivate;
struct _GDBusObjectSkeleton {
    GObject parent_instance;
    GDBusObjectSkeletonPrivate *priv;
};

extern void
g_dbus_object_skeleton_add_interface(GDBusObjectSkeleton *
                                     object,
                                     GDBusInterfaceSkeleton *
                                     interface_);

extern void g_dbus_object_skeleton_flush(GDBusObjectSkeleton *
object);
extern GType g_dbus_object_skeleton_get_type(void);
extern GDBusObjectSkeleton *g_dbus_object_skeleton_new(const gchar *
object_path);

extern void
g_dbus_object_skeleton_remove_interface(GDBusObjectSkeleton *
object,
GDBusInterfaceSkeleton
* interface_);

extern void
g_dbus_object_skeleton_remove_interface_by_name(GDBusObjectSkelet
on *
object,
const gchar *
interface_name);

extern void
g_dbus_object_skeleton_set_object_path(GDBusObjectSkeleton *
object,
const gchar *
object_path);

```

17.12.50 glib-2.0/gio/gdbusproxy.h

```

#define G_TYPE_DBUS_PROXY (g_dbus_proxy_get_type ())
#define G_DBUS_PROXY_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_DBUS_PROXY, GDBusProxyClass))
#define G_IS_DBUS_PROXY_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_DBUS_PROXY))
#define G_DBUS_PROXY(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_PROXY, GDBusProxy))
#define G_IS_DBUS_PROXY(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_PROXY))
#define G_DBUS_PROXY_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_DBUS_PROXY, GDBusProxyClass))

typedef struct _GDBusProxyClass {
    GObjectClass parent_class;
    void (*g_properties_changed) (GDBusProxy * proxy,
GVariant * changed_properties,
const gchar *
const *invalidated_properties);
    void (*g_signal) (GDBusProxy * proxy, const gchar * sender_name,
const gchar * signal_name, GVariant *
parameters);
    gpointer padding[32];
} GDBusProxyClass;
typedef struct _GDBusProxyPrivate GDBusProxyPrivate;
struct _GDBusProxy {
    GObject parent_instance;

```

```

    GDBusProxyPrivate *priv;
};
extern void g_dbus_proxy_call(GDBusProxy * proxy,
                             const gchar * method_name,
                             GVariant * parameters, GDBusCallFlags
flags,
                             gint timeout_msec,
                             Gancellable * cancellable,
                             GAsyncReadyCallback callback,
                             gpointer user_data);
extern GVariant *g_dbus_proxy_call_finish(GDBusProxy * proxy,
                                           GAsyncResult * res,
                                           GError * *error);
extern GVariant *g_dbus_proxy_call_sync(GDBusProxy * proxy,
                                        const gchar * method_name,
                                        GVariant * parameters,
                                        GDBusCallFlags flags,
                                        gint timeout_msec,
                                        Gancellable * cancellable,
                                        GError * *error);
extern void g_dbus_proxy_call_with_unix_fd_list(GDBusProxy * proxy,
                                                const gchar * method_name,
                                                GVariant * parameters,
                                                GDBusCallFlags flags,
                                                gint timeout_msec,
                                                GUnixFDList * fd_list,
                                                Gancellable * cancellable,
                                                GAsyncReadyCallback
callback,
                                                gpointer user_data);
extern GVariant
*g_dbus_proxy_call_with_unix_fd_list_finish(GDBusProxy *
                                           proxy,
                                           GUnixFDList *
*out_fd_list,
                                           GAsyncResult *
res,
                                           GError *
*error);
extern GVariant
*g_dbus_proxy_call_with_unix_fd_list_sync(GDBusProxy *
                                           proxy,
                                           const gchar *
method_name,
                                           GVariant *
parameters,
                                           GDBusCallFlags
flags,
                                           gint
timeout_msec,
                                           GUnixFDList *
fd_list,
                                           GUnixFDList *
*out_fd_list,
                                           Gancellable *
cancellable,
                                           GError * *error);
extern GVariant *g_dbus_proxy_get_cached_property(GDBusProxy *
proxy,
                                                  const gchar *
property_name);
extern gchar **g_dbus_proxy_get_cached_property_names(GDBusProxy *
proxy);
extern GDBusConnection *g_dbus_proxy_get_connection(GDBusProxy *
proxy);
extern gint g_dbus_proxy_get_default_timeout(GDBusProxy * proxy);

```

```

extern GDBusProxyFlags g_dbus_proxy_get_flags(GDBusProxy * proxy);
extern
GDBusInterfaceInfo
*g_dbus_proxy_get_interface_info(GDBusProxy *
                                proxy);
extern const char *g_dbus_proxy_get_interface_name(GDBusProxy *
proxy);
extern const char *g_dbus_proxy_get_name(GDBusProxy * proxy);
extern gchar *g_dbus_proxy_get_name_owner(GDBusProxy * proxy);
extern const char *g_dbus_proxy_get_object_path(GDBusProxy * proxy);
extern GType g_dbus_proxy_get_type(void);
extern void g_dbus_proxy_new(GDBusConnection * connection,
                             GDBusProxyFlags flags,
                             GDBusInterfaceInfo * info, const gchar *
name,
                             const gchar * object_path,
                             const gchar * interface_name,
                             Gancellable * cancellable,
                             GAsyncReadyCallback callback,
                             gpointer user_data);
extern GDBusProxy *g_dbus_proxy_new_finish(GAsyncResult * res,
                                             GError * *error);
extern void g_dbus_proxy_new_for_bus(GBusType bus_type,
                                     GDBusProxyFlags flags,
                                     GDBusInterfaceInfo * info,
                                     const gchar * name,
                                     const gchar * object_path,
                                     const gchar * interface_name,
                                     Gancellable * cancellable,
                                     GAsyncReadyCallback callback,
                                     gpointer user_data);
extern GDBusProxy *g_dbus_proxy_new_for_bus_finish(GAsyncResult *
res,
                                                    GError * *error);
extern GDBusProxy *g_dbus_proxy_new_for_bus_sync(GBusType bus_type,
                                                  GDBusProxyFlags flags,
                                                  GDBusInterfaceInfo * info,
                                                  const gchar * name,
                                                  const gchar * object_path,
                                                  const gchar *
                                                  interface_name,
                                                  Gancellable *
                                                  cancellable,
                                                  GError * *error);
extern GDBusProxy *g_dbus_proxy_new_sync(GDBusConnection *
connection,
                                         GDBusProxyFlags flags,
                                         GDBusInterfaceInfo * info,
                                         const gchar * name,
                                         const gchar * object_path,
                                         const gchar * interface_name,
                                         Gancellable * cancellable,
                                         GError * *error);
extern void g_dbus_proxy_set_cached_property(GDBusProxy * proxy,
                                              const gchar * property_name,
                                              GVariant * value);
extern void g_dbus_proxy_set_default_timeout(GDBusProxy * proxy,
                                              gint timeout_msec);
extern void g_dbus_proxy_set_interface_info(GDBusProxy * proxy,
                                              GDBusInterfaceInfo * info);

```

17.12.51 glib-2.0/gio/gdbusserver.h

```

#define G_TYPE_DBUS_SERVER (g_dbus_server_get_type ())
#define G_DBUS_SERVER(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_DBUS_SERVER, GDBusServer))

```



```

#define G_IS_DBUS_SERVER(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_DBUS_SERVER))

extern const char *g_dbus_server_get_client_address(GDBusServer *
server);
extern GDBusServerFlags g_dbus_server_get_flags(GDBusServer *
server);
extern const char *g_dbus_server_get_guid(GDBusServer * server);
extern GType g_dbus_server_get_type(void);
extern gboolean g_dbus_server_is_active(GDBusServer * server);
extern GDBusServer *g_dbus_server_new_sync(const gchar * address,
GDBusServerFlags flags,
const gchar * guid,
GDBusAuthObserver * observer,
GCancelable * cancellable,
GError * *error);
extern void g_dbus_server_start(GDBusServer * server);
extern void g_dbus_server_stop(GDBusServer * server);

```

17.12.52 glib-2.0/gio/gdbusutils.h

```

extern gchar *g_dbus_generate_guid(void);
extern GVariant *g_dbus_gvalue_to_gvariant(const GValue * gvalue,
const GVariantType * type);
extern void g_dbus_gvariant_to_gvalue(GVariant * value,
GValue * out_gvalue);
extern gboolean g_dbus_is_guid(const gchar * string);
extern gboolean g_dbus_is_interface_name(const gchar * string);
extern gboolean g_dbus_is_member_name(const gchar * string);
extern gboolean g_dbus_is_name(const gchar * string);
extern gboolean g_dbus_is_unique_name(const gchar * string);

```

17.12.53 glib-2.0/gio/gdrive.h

```

#define G_TYPE_DRIVE (g_drive_get_type ())
#define G_DRIVE(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_DRIVE, GDrive))
#define G_IS_DRIVE(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_DRIVE))
#define G_DRIVE_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE
((obj), G_TYPE_DRIVE, GDriveIface))

typedef struct _GDriveIface {
GTypeInterface g_iface;
void (*changed) (GDrive * drive);
void (*disconnected) (GDrive * drive);
void (*eject_button) (GDrive * drive);
char *(*get_name) (GDrive * drive);
GIcon *(*get_icon) (GDrive * drive);
gboolean(*has_volumes) (GDrive * drive);
GList *(*get_volumes) (GDrive * drive);
gboolean(*is_media_removable) (GDrive * drive);
gboolean(*has_media) (GDrive * drive);
gboolean(*is_media_check_automatic) (GDrive * drive);
gboolean(*can_eject) (GDrive * drive);
gboolean(*can_poll_for_media) (GDrive * drive);
void (*eject) (GDrive * drive, GMountUnmountFlags flags,
GCancelable * cancellable,
GAsyncReadyCallback callback, gpointer user_data);
gboolean(*eject_finish) (GDrive * drive, GAsyncResult * result,
GError * *error);
void (*poll_for_media) (GDrive * drive, GCancelable *
cancellable,

```

```

        GAsyncReadyCallback callback,
        gpointer user_data);
gboolean(*poll_for_media_finish) (GDrive * drive,
        GAsyncResult * result,
        GError * *error);
char *(*get_identifier) (GDrive * drive, const char *kind);
char **(*enumerate_identifiers) (GDrive * drive);
GDriveStartStopType(*get_start_stop_type) (GDrive * drive);
gboolean(*can_start) (GDrive * drive);
gboolean(*can_start_degraded) (GDrive * drive);
void (*start) (GDrive * drive, GDriveStartFlags flags,
        GMountOperation mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback, gpointer user_data);
gboolean(*start_finish) (GDrive * drive, GAsyncResult * result,
        GError * *error);
gboolean(*can_stop) (GDrive * drive);
void (*stop) (GDrive * drive, GMountUnmountFlags flags,
        GMountOperation * mount_operation,
        GCancellable * cancellable, GAsyncReadyCallback
callback,
        gpointer user_data);
gboolean(*stop_finish) (GDrive * drive, GAsyncResult * result,
        GError * *error);
void (*stop_button) (GDrive * drive);
void (*eject_with_operation) (GDrive * drive,
        GMountUnmountFlags flags,
        GMountOperation * mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
gboolean(*eject_with_operation_finish) (GDrive * drive,
        GAsyncResult * result,
        GError * *error);
const gchar *(*get_sort_key) (GDrive * drive);
} GDriveIface;
extern gboolean g_drive_can_eject(GDrive * drive);
extern gboolean g_drive_can_poll_for_media(GDrive * drive);
extern gboolean g_drive_can_start(GDrive * drive);
extern gboolean g_drive_can_start_degraded(GDrive * drive);
extern gboolean g_drive_can_stop(GDrive * drive);
extern void g_drive_eject(GDrive * drive, GMountUnmountFlags flags,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_drive_eject_finish(GDrive * drive, GAsyncResult
* result,
        GError * *error);
extern void g_drive_eject_with_operation(GDrive * drive,
        GMountUnmountFlags flags,
        GMountOperation * mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_drive_eject_with_operation_finish(GDrive * drive,
        GAsyncResult * result,
        GError * *error);
extern void g_drive_enumerate_identifiers(GDrive * drive);
extern GIcon *g_drive_get_icon(GDrive * drive);
extern char *g_drive_get_identifier(GDrive * drive, const char
*kind);
extern char *g_drive_get_name(GDrive * drive);
extern const char *g_drive_get_sort_key(GDrive * drive);
extern GDriveStartStopType g_drive_get_start_stop_type(GDrive *
drive);

```

```

extern GType g_drive_get_type(void);
extern GList *g_drive_get_volumes(GDrive * drive);
extern gboolean g_drive_has_media(GDrive * drive);
extern gboolean g_drive_has_volumes(GDrive * drive);
extern gboolean g_drive_is_media_check_automatic(GDrive * drive);
extern gboolean g_drive_is_media_removable(GDrive * drive);
extern void g_drive_poll_for_media(GDrive * drive,
                                   Gancellable * cancellable,
                                   GAsyncReadyCallback callback,
                                   gpointer user_data);
extern gboolean g_drive_poll_for_media_finish(GDrive * drive,
                                              GAsyncResult * result,
                                              GError * *error);
extern void g_drive_start(GDrive * drive, GDriveStartFlags flags,
                          GMountOperation * mount_operation,
                          Gancellable * cancellable,
                          GAsyncReadyCallback callback,
                          gpointer user_data);
extern gboolean g_drive_start_finish(GDrive * drive, GAsyncResult
* result,
                                   GError * *error);
extern void g_drive_stop(GDrive * drive, GMountUnmountFlags flags,
                          GMountOperation * mount_operation,
                          Gancellable * cancellable,
                          GAsyncReadyCallback callback,
                          gpointer
user_data);
extern gboolean g_drive_stop_finish(GDrive * drive, GAsyncResult *
result,
                                   GError * *error);

```

17.12.54 glib-2.0/gio/gemblem.h

```

#define G_TYPE_EMBLEM (g_emblem_get_type ())
#define G_EMBLEM_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_EMBLEM, GEmblemClass))
#define G_IS_EMBLEM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_TYPE_EMBLEM))
#define G_EMBLEM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_EMBLEM, GEmblem))
#define G_IS_EMBLEM(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_EMBLEM))
#define G_EMBLEM_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o),
G_TYPE_EMBLEM, GEmblemClass))

typedef struct _GEmblem GEmblem;
typedef struct _GEmblemClass GEmblemClass;
extern GIcon *g_emblem_get_icon(GEmblem * emblem);
extern GEmblemOrigin g_emblem_get_origin(GEmblem * emblem);
extern GType g_emblem_get_type(void);
extern GEmblem *g_emblem_new(GIcon * icon);
extern GEmblem *g_emblem_new_with_origin(GIcon * icon,
                                         GEmblemOrigin origin);

```

17.12.55 glib-2.0/gio/gemblemedicon.h

```

#define G_TYPE_EMBLEMED_ICON (g_emblemed_icon_get_type ())
#define G_EMBLEMED_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_EMBLEMED_ICON, GEmblemedIconClass))
#define G_IS_EMBLEMED_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_EMBLEMED_ICON))
#define G_EMBLEMED_ICON(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_EMBLEMED_ICON, GEmblemedIcon))

```

```

#define G_IS_EMBLEMED_ICON(o)      (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_EMBLEMED_ICON))
#define G_EMBLEMED_ICON_GET_CLASS(o)  (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_EMBLEMED_ICON, GEmblemedIconClass))

typedef struct _GEmblemedIcon {
    GObject parent_instance;
    GEmblemedIconPrivate *priv;
} GEmblemedIcon;
typedef struct _GEmblemedIconClass {
    GObjectClass parent_class;
} GEmblemedIconClass;
typedef struct _GEmblemedIconPrivate GEmblemedIconPrivate;
extern void g_emblemed_icon_add_emblem(GEmblemedIcon * emblemed,
                                       GEmblem * emblem);
extern void g_emblemed_icon_clear_emblems(GEmblemedIcon *
emblemed);
extern GList *g_emblemed_icon_get_emblems(GEmblemedIcon *
emblemed);
extern GIcon *g_emblemed_icon_get_icon(GEmblemedIcon * emblemed);
extern GType g_emblemed_icon_get_type(void);
extern GIcon *g_emblemed_icon_new(GIcon * icon, GEmblem * emblem);

```

17.12.56 glib-2.0/gio/gfile.h

```

#define G_TYPE_FILE      (g_file_get_type ())
#define G_FILE(obj)      (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_FILE, GFile))
#define G_IS_FILE(obj)   (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_FILE))
#define G_FILE_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE
((obj), G_TYPE_FILE, GFileIface))

struct _GFileIface {
    GTypeInterface g_iface;
    GFile *(*dup) (GFile * file);
    guint(*hash) (GFile * file);
    gboolean(*equal) (GFile * file1, GFile * file2);
    gboolean(*is_native) (GFile * file);
    gboolean(*has_uri_scheme) (GFile * file, const char
*uri_scheme);
    char *(*get_uri_scheme) (GFile * file);
    char *(*get_basename) (GFile * file);
    char *(*get_path) (GFile * file);
    char *(*get_uri) (GFile * file);
    char *(*get_parse_name) (GFile * file);
    GFile *(*get_parent) (GFile * file);
    gboolean(*prefix_matches) (GFile * prefix, GFile * file);
    char *(*get_relative_path) (GFile * parent, GFile * descendant);
    GFile *(*resolve_relative_path) (GFile * file,
const char *relative_path);
    GFile *(*get_child_for_display_name) (GFile * file,
const char *display_name,
GError * *error);
    GFileEnumerator *(*enumerate_children) (GFile * file,
const char *attributes,
GFileQueryInfoFlags flags,
Gancellable * cancellable,
GError * *error);
    void (*enumerate_children_async) (GFile * file, const char
*attributes,
GFileQueryInfoFlags flags,
int io_priority,
Gancellable * cancellable,
GAsyncReadyCallback callback,

```

```

        gpointer user_data);
GFileEnumerator *(*enumerate_children_finish) (GFile * file,
                                              GAsyncResult * res,
                                              GError * *error);
GFileInfo *(*query_info) (GFile * file, const char *attributes,
                          GFileQueryInfoFlags flags,
                          GCancellable * cancellable, GError *
*error);
void (*query_info_async) (GFile * file, const char *attributes,
                          GFileQueryInfoFlags flags, int
io_priority,
                          GCancellable * cancellable,
                          GAsyncReadyCallback callback,
                          gpointer user_data);
GFileInfo *(*query_info_finish) (GFile * file, GAsyncResult *
res,
                              GError * *error);
GFileInfo *(*query_filesystem_info) (GFile * file,
                                     const char *attributes,
                                     GCancellable * cancellable,
                                     GError * *error);
void (*query_filesystem_info_async) (GFile * file,
                                     const char *attributes,
                                     int io_priority,
                                     GCancellable * cancellable,
                                     GAsyncReadyCallback callback,
                                     gpointer user_data);
GFileInfo *(*query_filesystem_info_finish) (GFile * file,
                                           GAsyncResult * res,
                                           GError * *error);
GMount *(*find_enclosing_mount) (GFile * file,
                                 GCancellable * cancellable,
                                 GError * *error);
void (*find_enclosing_mount_async) (GFile * file, int
io_priority,
                                 GCancellable * cancellable,
                                 GAsyncReadyCallback callback,
                                 gpointer user_data);
GMount *(*find_enclosing_mount_finish) (GFile * file,
                                       GAsyncResult * res,
                                       GError * *error);
GFile *(*set_display_name) (GFile * file, const char
*display_name,
                          GCancellable * cancellable,
                          GError * *error);
void (*set_display_name_async) (const char *display_name,
                              int io_priority,
                              GCancellable * cancellable,
                              GAsyncReadyCallback callback,
                              gpointer user_data);
GFile *(*set_display_name_finish) (GFile * file, GAsyncResult *
res,
                              GError * *error);
GFileAttributeInfoList *(*query_settable_attributes) (GFile *
file,
                                                      GCancellable *
cancellable,
                                                      GError * *error);
void (*_query_settable_attributes_async) (void);
void (*_query_settable_attributes_finish) (void);
GFileAttributeInfoList *(*query_writable_namespaces) (GFile *
file,
                                                      GCancellable *
cancellable,
                                                      GError * *error);
void (*_query_writable_namespaces_async) (void);

```

```

void (*_query_writable_namespaces_finish) (void);
gboolean(*set_attribute) (GFile * file, const char *attribute,
                          GFileAttributeType type, gpointer
value_p,
                          GFileQueryInfoFlags flags,
                          GCancellable * cancellable,
                          GError * *error);
gboolean(*set_attributes_from_info) (GFile * file, GFileInfo *
info,
                                     GFileQueryInfoFlags flags,
                                     GCancellable * cancellable,
                                     GError * *error);
void (*set_attributes_async) (GFile * file, GFileInfo * info,
                              GFileQueryInfoFlags flags,
                              int io_priority,
                              GCancellable * cancellable,
                              GAsyncReadyCallback callback,
                              gpointer user_data);
gboolean(*set_attributes_finish) (GFile * file, GAsyncResult *
result,
                                 GFileInfo * *info, GError *
*error);
GFileInputStream *(*read_fn) (GFile * file, GCancellable *
cancellable,
                              GError * *error);
void (*read_async) (GFile * file, int io_priority,
                   GCancellable * cancellable,
                   GAsyncReadyCallback callback, gpointer
user_data);
GFileInputStream *(*read_finish) (GFile * file, GAsyncResult *
res,
                                  GError * *error);
GFileOutputStream *(*append_to) (GFile * file, GFileCreateFlags
flags,
                                 GCancellable * cancellable,
                                 GError * *error);
void (*append_to_async) (GFile * file, GFileCreateFlags flags,
                         int io_priority, GCancellable *
cancellable,
                         GAsyncReadyCallback callback,
                         gpointer user_data);
GFileOutputStream *(*append_to_finish) (GFile * file,
                                         GAsyncResult * res,
                                         GError * *error);
GFileOutputStream *(*create) (GFile * file, GFileCreateFlags
flags,
                              GCancellable * cancellable,
                              GError * *error);
void (*create_async) (GFile * file, GFileCreateFlags flags,
                     int io_priority, GCancellable * cancellable,
                     GAsyncReadyCallback callback,
                     gpointer user_data);
GFileOutputStream *(*create_finish) (GFile * file, GAsyncResult
* res,
                                     GError * *error);
GFileOutputStream *(*replace) (GFile * file, const char *etag,
                               gboolean make_backup,
                               GFileCreateFlags flags,
                               GCancellable * cancellable,
                               GError * *error);
void (*replace_async) (GFile * file, const char *etag,
                      gboolean make_backup, GFileCreateFlags flags,
                      int io_priority, GCancellable * cancellable,
                      GAsyncReadyCallback callback,
                      gpointer user_data);

```

```

GFileOutputStream      (*replace_finish)      (GFile      *      file,
GAsyncResult * res,
                                GError * *error);
gboolean(*delete_file)      (GFile      *      file,      Gancellable *
cancellable,
                                GError * *error);
void (*_delete_file_async) (void);
void (*_delete_file_finish) (void);
gboolean(*trash) (GFile * file, Gancellable * cancellable,
                                GError * *error);
void (*_trash_async) (void);
void (*_trash_finish) (void);
gboolean(*make_directory)      (GFile      *      file,      Gancellable *
cancellable,
                                GError * *error);
void (*_make_directory_async) (void);
void (*_make_directory_finish) (void);
gboolean(*make_symbolic_link) (GFile * file,
                                const char *symlink_value,
                                Gancellable * cancellable,
                                GError * *error);
void (*_make_symbolic_link_async) (void);
void (*_make_symbolic_link_finish) (void);
gboolean(*copy) (GFile * source, GFile * destination,
GFileCopyFlags flags, Gancellable * cancellable,
GFileProgressCallback progress_callback,
gpointer progress_callback_data, GError *
*error);
void (*copy_async) (GFile * source, GFile * destination,
GFileCopyFlags flags, int io_priority,
Gancellable * cancellable,
GFileProgressCallback progress_callback,
gpointer progress_callback_data,
GAsyncReadyCallback callback,      gpointer
user_data);
gboolean(*copy_finish) (GFile * file, GAsyncResult * res,
GError * *error);
gboolean(*move) (GFile * source, GFile * destination,
GFileCopyFlags flags, Gancellable * cancellable,
GFileProgressCallback progress_callback,
gpointer progress_callback_data, GError *
*error);
void (*_move_async) (void);
void (*_move_finish) (void);
void (*mount_mountable) (GFile * file, GMountMountFlags flags,
GMountOperation * mount_operation,
Gancellable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
GFile *(*mount_mountable_finish) (GFile * file, GAsyncResult *
result,
                                GError * *error);
void (*unmount_mountable) (GFile * file, GMountMountFlags flags,
Gancellable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
gboolean(*unmount_mountable_finish) (GFile * file,
GAsyncResult * result,
GError * *error);
void (*eject_mountable) (GFile * file, GMountMountFlags flags,
Gancellable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
gboolean(*eject_mountable_finish) (GFile * file,
GAsyncResult * result,
GError * *error);

```

```

void (*mount_enclosing_volume) (GFile * file, GMountMountFlags
flags,
                                GMountOperation * mount_operation,
                                Gancellable * cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
gboolean(*mount_enclosing_volume_finish) (GFile * file,
                                           GAsyncResult * result,
                                           GError * *error);
GFileMonitor *(*monitor_dir) (GFile * file, GFileMonitorFlags
flags,
                              Gancellable * cancellable,
                              GError * *error);
GFileMonitor *(*monitor_file) (GFile * file, GFileMonitorFlags
flags,
                              Gancellable * cancellable,
                              GError * *error);
GFileIOStream *(*open_readwrite) (GFile * file,
                                  Gancellable * cancellable,
                                  GError * *error);
void (*open_readwrite_async) (GFile * file, int io_priority,
                              Gancellable * cancellable,
                              GAsyncReadyCallback callback,
                              gpointer user_data);
GFileIOStream *(*open_readwrite_finish) (GFile * file,
                                          GAsyncResult * res,
                                          GError * *error);
GFileIOStream *(*create_readwrite) (GFile * file,
                                    GFileCreateFlags flags,
                                    Gancellable * cancellable,
                                    GError * *error);
void (*create_readwrite_async) (GFile * file, GFileCreateFlags
flags,
                              int io_priority,
                              Gancellable * cancellable,
                              GAsyncReadyCallback callback,
                              gpointer user_data);
GFileIOStream *(*create_readwrite_finish) (GFile * file,
                                           GAsyncResult * res,
                                           GError * *error);
GFileIOStream *(*replace_readwrite) (GFile * file, const char
*etag,
                                     gboolean make_backup,
                                     GFileCreateFlags flags,
                                     Gancellable * cancellable,
                                     GError * *error);
void (*replace_readwrite_async) (GFile * file, const char *etag,
                                gboolean make_backup,
                                GFileCreateFlags flags,
                                int io_priority,
                                Gancellable * cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
GFileIOStream *(*replace_readwrite_finish) (GFile * file,
                                           GAsyncResult * res,
                                           GError * *error);
void (*start_mountable) (GFile * file, GDriveStartFlags flags,
                        GMountOperation * start_operation,
                        Gancellable * cancellable,
                        GAsyncReadyCallback callback,
                        gpointer user_data);
gboolean(*start_mountable_finish) (GFile * file,
                                   GAsyncResult * result,
                                   GError * *error);
void (*stop_mountable) (GFile * file, GMountUnmountFlags flags,
                        GMountOperation * mount_operation,

```



```

        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
gboolean(*stop_mountable_finish)(GFile * file, GAsyncResult *
result,
                                GError * *error);
gboolean supports_thread_contexts;
void (*unmount_mountable_with_operation)(GFile * file,
                                          GMountUnmountFlags flags,
                                          GMountOperation *
                                          mount_operation,
                                          GCancelable * cancellable,
                                          GAsyncReadyCallback
callback,
                                          gpointer user_data);
gboolean(*unmount_mountable_with_operation_finish)(GFile *
file,
                                                    GAsyncResult *
result,
                                                    GError * *error);
void (*eject_mountable_with_operation)(GFile * file,
                                        GMountUnmountFlags flags,
                                        GMountOperation *
                                        mount_operation,
                                        GCancelable * cancellable,
                                        GAsyncReadyCallback callback,
                                        gpointer user_data);
gboolean(*eject_mountable_with_operation_finish)(GFile * file,
                                                  GAsyncResult *
                                                  result,
                                                  GError * *error);
void (*poll_mountable)(GFile * file, GCancelable * cancellable,
                        gpointer user_data);
gboolean(*poll_mountable_finish)(GFile * file, GAsyncResult *
result,
                                GError * *error);
};
extern GFileOutputStream *g_file_append_to(GFile * file,
                                           GFileCreateFlags flags,
                                           GCancelable * cancellable,
                                           GError * *error);
extern void g_file_append_to_async(GFile * file, GFileCreateFlags
flags,
                                  int io_priority,
                                  GCancelable * cancellable,
                                  GAsyncReadyCallback callback,
                                  gpointer user_data);
extern GFileOutputStream *g_file_append_to_finish(GFile * file,
                                                  GAsyncResult * res,
                                                  GError * *error);
extern gboolean g_file_copy(GFile * source, GFile * destination,
                            GFileCopyFlags flags,
                            GCancelable * cancellable,
                            GFileProgressCallback progress_callback,
                            gpointer progress_callback_data,
                            GError * *error);
extern void g_file_copy_async(GFile * source, GFile * destination,
                              GFileCopyFlags flags, int io_priority,
                              GCancelable * cancellable,
                              GFileProgressCallback progress_callback,
                              gpointer progress_callback_data,
                              GAsyncReadyCallback callback,
                              gpointer user_data);
extern gboolean g_file_copy_attributes(GFile * source, GFile *
destination,
                                       GFileCopyFlags flags,

```

```

        GCancelable * cancellable,
        GError * *error);
extern gboolean g_file_copy_finish(GFile * file, GAsyncResult * res,
        GError * *error);
extern GFileOutputStream *g_file_create(GFile * file,
        GFileCreateFlags flags,
        GCancelable * cancellable,
        GError * *error);
extern void g_file_create_async(GFile * file, GFileCreateFlags
flags,
        int io_priority,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern GFileOutputStream *g_file_create_finish(GFile * file,
        GAsyncResult * res,
        GError * *error);
extern GFileIOStream *g_file_create_readwrite(GFile * file,
        GFileCreateFlags flags,
        GCancelable * cancellable,
        GError * *error);
extern void g_file_create_readwrite_async(GFile * file,
        GFileCreateFlags flags,
        int io_priority,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern GFileIOStream *g_file_create_readwrite_finish(GFile * file,
        GAsyncResult * res,
        GError * *error);
extern gboolean g_file_delete(GFile * file, GCancelable *
cancellable,
        GError * *error);
extern GFile *g_file_dup(GFile * file);
extern void g_file_eject_mountable(GFile * file, GMountUnmountFlags
flags,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_file_eject_mountable_finish(GFile * file,
        GAsyncResult * result,
        GError * *error);
extern void g_file_eject_mountable_with_operation(GFile * file,
        GMountUnmountFlags flags,
        GMountOperation *
mount_operation,
        GCancelable *
cancellable,
        GAsyncReadyCallback
callback,
        gpointer user_data);
extern gboolean g_file_eject_mountable_with_operation_finish(GFile
* file,
        GAsyncResult *
result,
        GError *
*error);
extern GFileEnumerator *g_file_enumerate_children(GFile * file,
        const char *attributes,
        GFileQueryInfoFlags
flags,
        GCancelable *
cancellable,
        GError * *error);
extern void g_file_enumerate_children_async(GFile * file,
        const char *attributes,

```

```

        GFileQueryInfoFlags flags,
        int io_priority,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern GFileEnumerator *g_file_enumerate_children_finish(GFile *
file,
        GAsyncResult *
        res,
        GError * *error);
extern gboolean g_file_equal(GFile * file1, GFile * file2);
extern GMount *g_file_find_enclosing_mount(GFile * file,
        Gancellable * cancellable,
        GError * *error);
extern void g_file_find_enclosing_mount_async(GFile * file,
        int io_priority,
        Gancellable * cancellable,
        GAsyncReadyCallback
callback,
        gpointer user_data);
extern GMount *g_file_find_enclosing_mount_finish(GFile * file,
        GAsyncResult * res,
        GError * *error);
extern char *g_file_get_basename(GFile * file);
extern GFile *g_file_get_child(GFile * file, const char *name);
extern GFile *g_file_get_child_for_display_name(GFile * file,
        const char *display_name,
        GError * *error);
extern GFile *g_file_get_parent(GFile * file);
extern char *g_file_get_parse_name(GFile * file);
extern char *g_file_get_path(GFile * file);
extern char *g_file_get_relative_path(GFile * parent, GFile *
descendant);
extern GType g_file_get_type(void);
extern char *g_file_get_uri(GFile * file);
extern char *g_file_get_uri_scheme(GFile * file);
extern gboolean g_file_has_parent(GFile * file, GFile * parent);
extern gboolean g_file_has_prefix(GFile * file, GFile * prefix);
extern gboolean g_file_has_uri_scheme(GFile * file,
        const char *uri_scheme);
extern guint g_file_hash(gconstpointer file);
extern gboolean g_file_is_native(GFile * file);
extern gboolean g_file_load_contents(GFile * file,
        Gancellable * cancellable,
        char **contents, gsize * length,
        char **etag_out, GError * *error);
extern void g_file_load_contents_async(GFile * file,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_file_load_contents_finish(GFile * file,
        GAsyncResult * res,
        char **contents,
        gsize * length,
        char **etag_out,
        GError * *error);
extern void g_file_load_partial_contents_async(GFile * file,
        Gancellable * cancellable,
        GFileReadMoreCallback
read_more_callback,
        GAsyncReadyCallback
callback,
        gpointer user_data);
extern gboolean g_file_load_partial_contents_finish(GFile * file,
        GAsyncResult * res,
        char **contents,

```

```

        gsize * length,
        char **etag_out,
        GError * *error);
extern gboolean g_file_make_directory(GFile * file,
        GCancellable * cancellable,
        GError * *error);
extern gboolean g_file_make_directory_with_parents(GFile * file,
        GCancellable *
        cancellable,
        GError * *error);
extern gboolean g_file_make_symbolic_link(GFile * file,
        const char *symlink_value,
        GCancellable * cancellable,
        GError * *error);
extern GFileMonitor *g_file_monitor(GFile * file, GFileMonitorFlags
flags,
        GCancellable * cancellable,
        GError * *error);
extern GFileMonitor *g_file_monitor_directory(GFile * file,
        GFileMonitorFlags flags,
        GCancellable * cancellable,
        GError * *error);
extern GFileMonitor *g_file_monitor_file(GFile * file,
        GFileMonitorFlags flags,
        GCancellable * cancellable,
        GError * *error);
extern void g_file_mount_enclosing_volume(GFile * location,
        GMountMountFlags flags,
        GMountOperation *
        mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_file_mount_enclosing_volume_finish(GFile *
location,
        GAsyncResult * result,
        GError * *error);
extern void g_file_mount_mountable(GFile * file, GMountMountFlags
flags,
        GMountOperation * mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern GFile *g_file_mount_mountable_finish(GFile * file,
        GAsyncResult * result,
        GError * *error);
extern gboolean g_file_move(GFile * source, GFile * destination,
        GFileCopyFlags flags,
        GCancellable * cancellable,
        GFileProgressCallback progress_callback,
        gpointer progress_callback_data,
        GError * *error);
extern GFile *g_file_new_for_commandline_arg(const char *arg);
extern GFile *g_file_new_for_path(const char *path);
extern GFile *g_file_new_for_uri(const char *uri);
extern GFile *g_file_new_tmp(const char *tmpl, GFileIOStream *
*iostream,
        GError * *error);
extern GFileIOStream *g_file_open_readwrite(GFile * file,
        GCancellable * cancellable,
        GError * *error);
extern void g_file_open_readwrite_async(GFile * file, int
io_priority,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);

```

```

extern GFileIOStream *g_file_open_readwrite_finish(GFile * file,
                                                    GAsyncResult * res,
                                                    GError * *error);

extern GFile *g_file_parse_name(const char *parse_name);
extern void g_file_poll_mountable(GFile * file, GCancellable *
cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
extern gboolean g_file_poll_mountable_finish(GFile * file,
                                              GAsyncResult * result,
                                              GError * *error);
extern GAppInfo *g_file_query_default_handler(GFile * file,
                                              GCancellable * cancellable,
                                              GError * *error);
extern gboolean g_file_query_exists(GFile * file,
                                    GCancellable * cancellable);
extern GFileType g_file_query_file_type(GFile * file,
                                         GFileQueryInfoFlags flags,
                                         GCancellable * cancellable);
extern GFileInfo *g_file_query_filesystem_info(GFile * file,
                                              const char *attributes,
                                              GCancellable * cancellable,
                                              GError * *error);
extern void g_file_query_filesystem_info_async(GFile * file,
                                              const char *attributes,
                                              int io_priority,
                                              GCancellable * cancellable,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer user_data);
extern GFileInfo *g_file_query_filesystem_info_finish(GFile * file,
                                                      GAsyncResult * res,
                                                      GError * *error);
extern GFileInfo *g_file_query_info(GFile * file, const char
*attributes,
                                GFileQueryInfoFlags flags,
                                GCancellable * cancellable,
                                GError * *error);
extern void g_file_query_info_async(GFile * file, const char
*attributes,
                                GFileQueryInfoFlags flags,
                                int io_priority,
                                GCancellable * cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
extern GFileInfo *g_file_query_info_finish(GFile * file,
                                           GAsyncResult * res,
                                           GError * *error);
extern
*g_file_query_settable_attributes(GFile *
                                GFileAttributeInfoList
                                file,
                                GCancellable
                                *
                                cancellable,
                                GError *
                                *error);
extern
*g_file_query_writable_namespaces(GFile *
                                GFileAttributeInfoList
                                file,
                                GCancellable
                                *
                                cancellable,
                                GError *
                                *error);

extern GFileInputStream *g_file_read(GFile * file,
                                     GCancellable * cancellable,

```

```

        GError * *error);
extern void g_file_read_async(GFile * file, int io_priority,
                             Gancellable * cancellable,
                             GAsyncReadyCallback callback,
                             gpointer user_data);
extern GFileInputStream *g_file_read_finish(GFile * file,
                                             GAsyncResult * res,
                                             GError * *error);
extern GFileOutputStream *g_file_replace(GFile * file, const char
*etag,
                                         gboolean make_backup,
                                         GFileCreateFlags flags,
                                         Gancellable * cancellable,
                                         GError * *error);
extern void g_file_replace_async(GFile * file, const char *etag,
                                 gboolean make_backup,
                                 GFileCreateFlags flags, int io_priority,
                                 Gancellable * cancellable,
                                 GAsyncReadyCallback callback,
                                 gpointer user_data);
extern gboolean g_file_replace_contents(GFile * file, char
**contents,
                                         gsize length, const char *etag,
                                         gboolean make_backup,
                                         GFileCreateFlags flags,
                                         char **new_etag,
                                         Gancellable * cancellable,
                                         GError * *error);
extern void g_file_replace_contents_async(GFile * file,
                                          const char *contents,
                                          gsize length, const char *etag,
                                          gboolean make_backup,
                                          GFileCreateFlags flags,
                                          Gancellable * cancellable,
                                          GAsyncReadyCallback callback,
                                          gpointer user_data);
extern gboolean g_file_replace_contents_finish(GFile * file,
                                              GAsyncResult * res,
                                              char **new_etag,
                                              GError * *error);
extern GFileOutputStream *g_file_replace_finish(GFile * file,
                                                GAsyncResult * res,
                                                GError * *error);
extern GFileIOStream *g_file_replace_readwrite(GFile * file,
                                              const char *etag,
                                              gboolean make_backup,
                                              GFileCreateFlags flags,
                                              Gancellable * cancellable,
                                              GError * *error);
extern void g_file_replace_readwrite_async(GFile * file, const char
*etag,
                                           gboolean make_backup,
                                           GFileCreateFlags flags,
                                           int io_priority,
                                           Gancellable * cancellable,
                                           GAsyncReadyCallback callback,
                                           gpointer user_data);
extern GFileIOStream *g_file_replace_readwrite_finish(GFile * file,
                                                      GAsyncResult * res,
                                                      GError * *error);
extern GFile *g_file_resolve_relative_path(GFile * file,
                                           const char *relative_path);
extern gboolean g_file_set_attribute(GFile * file, const char
*attribute,
                                     GFileAttributeType type,
                                     gpointer value_p,

```

```

        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern gboolean g_file_set_attribute_byte_string(GFile * file,
        const char *attribute,
        const char *value,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern gboolean g_file_set_attribute_int32(GFile * file,
        const char *attribute,
        gint32 value,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern gboolean g_file_set_attribute_int64(GFile * file,
        const char *attribute,
        gint64 value,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern gboolean g_file_set_attribute_string(GFile * file,
        const char *attribute,
        const char *value,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern gboolean g_file_set_attribute_uint32(GFile * file,
        const char *attribute,
        guint32 value,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern gboolean g_file_set_attribute_uint64(GFile * file,
        const char *attribute,
        guint64 value,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern void g_file_set_attributes_async(GFile * file, GFileInfo *
info,
        GFileQueryInfoFlags flags,
        int io_priority,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_file_set_attributes_finish(GFile * file,
        GAsyncResult * result,
        GFileInfo * *info,
        GError * *error);
extern gboolean g_file_set_attributes_from_info(GFile * file,
        GFileInfo * info,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
extern GFile *g_file_set_display_name(GFile * file,
        const char *display_name,
        Gancellable * cancellable,
        GError * *error);
extern void g_file_set_display_name_async(GFile * file,
        const char *display_name,
        int io_priority,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);

```

```

struct _GFileAttributeInfo {
    char *name;
    GFileAttributeType type;
    GFileAttributeInfoFlags flags;
};
struct _GFileAttributeInfoList {
    GFileAttributeInfo *infos;
    int n_infos;
};
extern void g_file_attribute_info_list_add(GFileAttributeInfoList
* list,
                                         const char *name,

```



```

        GFileAttributeType type,
        GFileAttributeInfoFlags flags);

extern GFileAttributeInfoList
    *g_file_attribute_info_list_dup(GFileAttributeInfoList * list);
extern GType g_file_attribute_info_list_get_type(void);
extern const GFileAttributeInfo
    *g_file_attribute_info_list_lookup(GFileAttributeInfoList *
    list,
                                     const char *name);
extern
    GFileAttributeInfoList
    *g_file_attribute_info_list_new(void);
extern GFileAttributeInfoList
    *g_file_attribute_info_list_ref(GFileAttributeInfoList * list);
extern
    void
    g_file_attribute_info_list_unref(GFileAttributeInfoList *
    list);

```

17.12.58 glib-2.0/gio/gfileenumerator.h

```

#define G_TYPE_FILE_ENUMERATOR (g_file_enumerator_get_type ())
#define G_FILE_ENUMERATOR_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_FILE_ENUMERATOR, GFileEnumeratorClass))
#define G_IS_FILE_ENUMERATOR_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_FILE_ENUMERATOR))
#define G_FILE_ENUMERATOR(o) (G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_FILE_ENUMERATOR, GFileEnumerator))
#define G_IS_FILE_ENUMERATOR(o) (G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_FILE_ENUMERATOR))
#define G_FILE_ENUMERATOR_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_FILE_ENUMERATOR, GFileEnumeratorClass))

typedef struct _GFileEnumeratorClass {
    GObjectClass parent_class;
    GFileInfo *(*next_file)(GFileEnumerator * enumerator,
                           GCancellable * cancellable, GError *
    *error);
    gboolean(*close_fn)(GFileEnumerator * enumerator,
                       GCancellable * cancellable, GError * *error);
    void (*next_files_async)(GFileEnumerator * enumerator, int
    num_files,
                           int io_priority, GCancellable *
    cancellable,
                           GAsyncReadyCallback callback,
                           gpointer user_data);
    GList *(*next_files_finish)(GFileEnumerator * enumerator,
                               GAsyncResult * result, GError * *error);
    void (*close_async)(GFileEnumerator * enumerator, int
    io_priority,
                           GCancellable * cancellable,
                           GAsyncReadyCallback callback, gpointer
    user_data);
    gboolean(*close_finish)(GFileEnumerator * enumerator,
                            GAsyncResult * result, GError * *error);
    void (*_g_reserved1)(void);
    void (*_g_reserved2)(void);
    void (*_g_reserved3)(void);
    void (*_g_reserved4)(void);
    void (*_g_reserved5)(void);
    void (*_g_reserved6)(void);
    void (*_g_reserved7)(void);
} GFileEnumeratorClass;
typedef struct _GFileEnumeratorPrivate GFileEnumeratorPrivate;
struct _GFileEnumerator {
    GObject parent_instance;

```

```

    GFileEnumeratorPrivate *priv;
};
extern gboolean g_file_enumerator_close(GFileEnumerator *
enumerator,
                                     GCancellable * cancellable,
                                     GError * *error);
extern void g_file_enumerator_close_async(GFileEnumerator *
enumerator,
                                     int io_priority,
                                     GCancellable * cancellable,
                                     GAsyncReadyCallback callback,
                                     gpointer user_data);
extern gboolean g_file_enumerator_close_finish(GFileEnumerator *
enumerator,
                                     GAsyncResult * result,
                                     GError * *error);
extern GFile *g_file_enumerator_get_container(GFileEnumerator *
enumerator);
extern GType g_file_enumerator_get_type(void);
extern gboolean g_file_enumerator_has_pending(GFileEnumerator *
enumerator);
extern gboolean g_file_enumerator_is_closed(GFileEnumerator *
enumerator);
extern GFileInfo *g_file_enumerator_next_file(GFileEnumerator *
enumerator,
                                     GCancellable * cancellable,
                                     GError * *error);
extern void g_file_enumerator_next_files_async(GFileEnumerator *
enumerator, int num_files,
int io_priority,
GCancellable * cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);
extern GList *g_file_enumerator_next_files_finish(GFileEnumerator
*
enumerator,
GAsyncResult * result,
GError * *error);
extern void g_file_enumerator_set_pending(GFileEnumerator *
enumerator,
                                     gboolean pending);

```

17.12.59 glib-2.0/gio/gfileicon.h

```

#define G_TYPE_FILE_ICON (g_file_icon_get_type ())
#define G_FILE_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_FILE_ICON, GFileIconClass))
#define G_IS_FILE_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_TYPE_FILE_ICON))
#define G_FILE_ICON(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_FILE_ICON, GFileIcon))
#define G_IS_FILE_ICON(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_FILE_ICON))
#define G_FILE_ICON_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_FILE_ICON, GFileIconClass))

typedef struct _GFileIconClass GFileIconClass;
extern GFile *g_file_icon_get_file(GFileIcon * icon);
extern GType g_file_icon_get_type(void);
extern GIcon *g_file_icon_new(GFile * file);

```

17.12.60 glib-2.0/gio/gfileinfo.h

```

#define G_TYPE_FILE_INFO (g_file_info_get_type ())
#define G_FILE_INFO_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_FILE_INFO, GFileInfoClass))
#define G_IS_FILE_INFO_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_TYPE_FILE_INFO))
#define G_FILE_INFO(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_FILE_INFO, GFileInfo))
#define G_IS_FILE_INFO(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_FILE_INFO))
#define G_FILE_INFO_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_FILE_INFO, GFileInfoClass))
#define G_FILE_ATTRIBUTE_ACCESS_CAN_DELETE "access::can-
delete"
#define G_FILE_ATTRIBUTE_ACCESS_CAN_EXECUTE "access::can-
execute"
#define G_FILE_ATTRIBUTE_ACCESS_CAN_READ "access::can-read"
#define G_FILE_ATTRIBUTE_ACCESS_CAN_RENAME "access::can-
rename"
#define G_FILE_ATTRIBUTE_ACCESS_CAN_TRASH "access::can-trash"
#define G_FILE_ATTRIBUTE_ACCESS_CAN_WRITE "access::can-write"
#define G_FILE_ATTRIBUTE_DOS_IS_ARCHIVE "dos::is-archive"
#define G_FILE_ATTRIBUTE_DOS_IS_SYSTEM "dos::is-system"
#define G_FILE_ATTRIBUTE_ETAG_VALUE "etag::value"
#define G_FILE_ATTRIBUTE_FILESYSTEM_FREE "filesystem::free"
#define G_FILE_ATTRIBUTE_FILESYSTEM_READONLY "filesystem::readonly"
#define G_FILE_ATTRIBUTE_FILESYSTEM_SIZE "filesystem::size"
#define G_FILE_ATTRIBUTE_FILESYSTEM_TYPE "filesystem::type"
#define G_FILE_ATTRIBUTE_FILESYSTEM_USE_PREVIEW "filesystem::use-
preview"
#define G_FILE_ATTRIBUTE_FILESYSTEM_USED "filesystem::used"
#define G_FILE_ATTRIBUTE_GVFS_BACKEND "gvfs::backend"
#define G_FILE_ATTRIBUTE_ID_FILE "id::file"
#define G_FILE_ATTRIBUTE_ID_FILESYSTEM "id::filesystem"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_EJECT "mountable::can-
eject"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_MOUNT "mountable::can-
mount"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_POLL "mountable::can-
poll"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_START "mountable::can-
start"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_START_DEGRADED "mountable::can-start-degraded"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_STOP "mountable::can-
stop"
#define G_FILE_ATTRIBUTE_MOUNTABLE_CAN_UNMOUNT "mountable::can-
unmount"
#define G_FILE_ATTRIBUTE_MOUNTABLE_HAL_UDI "mountable::hal-
udi"
#define G_FILE_ATTRIBUTE_MOUNTABLE_IS_MEDIA_CHECK_AUTOMATIC "mountable::is-media-check-automatic"
#define G_FILE_ATTRIBUTE_MOUNTABLE_START_STOP_TYPE "mountable::start-stop-type"
#define G_FILE_ATTRIBUTE_MOUNTABLE_UNIX_DEVICE "mountable::unix-
device"
#define G_FILE_ATTRIBUTE_MOUNTABLE_UNIX_DEVICE_FILE "mountable::unix-device-file"
#define G_FILE_ATTRIBUTE_OWNER_GROUP "owner::group"
#define G_FILE_ATTRIBUTE_OWNER_USER "owner::user"
#define G_FILE_ATTRIBUTE_OWNER_USER_REAL "owner::user-real"
#define G_FILE_ATTRIBUTE_PREVIEW_ICON "preview::icon"
#define G_FILE_ATTRIBUTE_SELINUX_CONTEXT "selinux::context"

```

```

#define G_FILE_ATTRIBUTE_STANDARD_ALLOCATED_SIZE
"standard::allocated-size"
#define G_FILE_ATTRIBUTE_STANDARD_CONTENT_TYPE
"standard::content-type"
#define G_FILE_ATTRIBUTE_STANDARD_COPY_NAME "standard::copy-
name"
#define G_FILE_ATTRIBUTE_STANDARD_DESCRIPTION
"standard::description"
#define G_FILE_ATTRIBUTE_STANDARD_DISPLAY_NAME
"standard::display-name"
#define G_FILE_ATTRIBUTE_STANDARD_EDIT_NAME "standard::edit-
name"
#define G_FILE_ATTRIBUTE_STANDARD_FAST_CONTENT_TYPE
"standard::fast-content-type"
#define G_FILE_ATTRIBUTE_STANDARD_ICON "standard::icon"
#define G_FILE_ATTRIBUTE_STANDARD_IS_BACKUP "standard::is-
backup"
#define G_FILE_ATTRIBUTE_STANDARD_IS_HIDDEN "standard::is-
hidden"
#define G_FILE_ATTRIBUTE_STANDARD_IS_SYMLINK "standard::is-
symlink"
#define G_FILE_ATTRIBUTE_STANDARD_IS_VIRTUAL "standard::is-
virtual"
#define G_FILE_ATTRIBUTE_STANDARD_NAME "standard::name"
#define G_FILE_ATTRIBUTE_STANDARD_SIZE "standard::size"
#define G_FILE_ATTRIBUTE_STANDARD_SORT_ORDER "standard::sort-
order"
#define G_FILE_ATTRIBUTE_STANDARD_SYMLINK_TARGET
"standard::symlink-target"
#define G_FILE_ATTRIBUTE_STANDARD_TARGET_URI "standard::target-
uri"
#define G_FILE_ATTRIBUTE_STANDARD_TYPE "standard::type"
#define G_FILE_ATTRIBUTE_THUMBAILING_FAILED
"thumbnail::failed"
#define G_FILE_ATTRIBUTE_THUMBAIL_PATH "thumbnail::path"
#define G_FILE_ATTRIBUTE_TIME_ACCESS "time::access"
#define G_FILE_ATTRIBUTE_TIME_ACCESS_USEC "time::access-usec"
#define G_FILE_ATTRIBUTE_TIME_CHANGED "time::changed"
#define G_FILE_ATTRIBUTE_TIME_CHANGED_USEC "time::changed-
usec"
#define G_FILE_ATTRIBUTE_TIME_CREATED "time::created"
#define G_FILE_ATTRIBUTE_TIME_CREATED_USEC "time::created-
usec"
#define G_FILE_ATTRIBUTE_TIME_MODIFIED "time::modified"
#define G_FILE_ATTRIBUTE_TIME_MODIFIED_USEC "time::modified-
usec"
#define G_FILE_ATTRIBUTE_TRASH_DELETION_DATE "trash::deletion-
date"
#define G_FILE_ATTRIBUTE_TRASH_ITEM_COUNT "trash::item-count"
#define G_FILE_ATTRIBUTE_TRASH_ORIG_PATH "trash::orig-path"
#define G_FILE_ATTRIBUTE_UNIX_BLOCK_SIZE "unix::block-size"
#define G_FILE_ATTRIBUTE_UNIX_BLOCKS "unix::blocks"
#define G_FILE_ATTRIBUTE_UNIX_DEVICE "unix::device"
#define G_FILE_ATTRIBUTE_UNIX_GID "unix::gid"
#define G_FILE_ATTRIBUTE_UNIX_INODE "unix::inode"
#define G_FILE_ATTRIBUTE_UNIX_IS_MOUNTPOINT "unix::is-
mountpoint"
#define G_FILE_ATTRIBUTE_UNIX_MODE "unix::mode"
#define G_FILE_ATTRIBUTE_UNIX_NLINK "unix::nlink"
#define G_FILE_ATTRIBUTE_UNIX_RDEV "unix::rdev"
#define G_FILE_ATTRIBUTE_UNIX_UID "unix::uid"

typedef struct _GFileInfoClass GFileInfoClass;
extern gboolean
g_file_attribute_matcher_enumerate_namespace(GFileAttributeMatche
r *

```

```

                                matcher, const char *ns);
extern const char
    *g_file_attribute_matcher_enumerate_next(GFileAttributeMatcher
    *
                                matcher);
extern GType g_file_attribute_matcher_get_type(void);
extern
    g_file_attribute_matcher_matches(GFileAttributeMatcher *
                                matcher,
                                const char *attribute);
extern
    g_file_attribute_matcher_matches_only(GFileAttributeMatcher
                                * matcher,
                                const char
                                *attribute);
extern GFileAttributeMatcher *g_file_attribute_matcher_new(const
    char
                                *attributes);
extern GFileAttributeMatcher
    *g_file_attribute_matcher_ref(GFileAttributeMatcher * matcher);
extern GFileAttributeMatcher
    *g_file_attribute_matcher_subtract(GFileAttributeMatcher *
    matcher,
                                GFileAttributeMatcher * subtract);
extern
    *g_file_attribute_matcher_to_string(GFileAttributeMatcher *
    matcher);
extern void g_file_attribute_matcher_unref(GFileAttributeMatcher *
    matcher);
extern void g_file_info_clear_status(GFileInfo * info);
extern void g_file_info_copy_into(GFileInfo * src_info,
    GFileInfo * dest_info);
extern GFileInfo *g_file_info_dup(GFileInfo * other);
extern char *g_file_info_get_attribute_as_string(GFileInfo * info,
    const char *attribute);
extern gboolean g_file_info_get_attribute_boolean(GFileInfo * info,
    const char *attribute);
extern const char *g_file_info_get_attribute_byte_string(GFileInfo
    * info,
                                const char
                                *attribute);
extern gboolean g_file_info_get_attribute_data(GFileInfo * info,
    const char *attribute,
    GFileAttributeType * type,
    void **value_pp,
    GFileAttributeStatus *
    status);
extern gint32 g_file_info_get_attribute_int32(GFileInfo * info,
    const char *attribute);
extern gint64 g_file_info_get_attribute_int64(GFileInfo * info,
    const char *attribute);
extern GObject *g_file_info_get_attribute_object(GFileInfo * info,
    const char *attribute);
extern
    g_file_info_get_attribute_status(GFileInfo *
                                info,
                                const char
                                *attribute);
extern const char *g_file_info_get_attribute_string(GFileInfo *
    info,
                                const char *attribute);
extern void g_file_info_get_attribute_stringv(GFileInfo * info,
    const char *attribute);
extern GFileAttributeType g_file_info_get_attribute_type(GFileInfo
    * info,
                                const char
                                *attribute);

```

```

        *attribute);
extern guint32 g_file_info_get_attribute_uint32(GFileInfo * info,
        const char *attribute);
extern guint64 g_file_info_get_attribute_uint64(GFileInfo * info,
        const char *attribute);
extern const char *g_file_info_get_content_type(GFileInfo * info);
extern const char *g_file_info_get_display_name(GFileInfo * info);
extern const char *g_file_info_get_edit_name(GFileInfo * info);
extern const char *g_file_info_get_etag(GFileInfo * info);
extern GFileType g_file_info_get_file_type(GFileInfo * info);
extern GIcon *g_file_info_get_icon(GFileInfo * info);
extern gboolean g_file_info_get_is_backup(GFileInfo * info);
extern gboolean g_file_info_get_is_hidden(GFileInfo * info);
extern gboolean g_file_info_get_is_symlink(GFileInfo * info);
extern void g_file_info_get_modification_time(GFileInfo * info,
        GTimeVal * result);
extern const char *g_file_info_get_name(GFileInfo * info);
extern goffset g_file_info_get_size(GFileInfo * info);
extern gint32 g_file_info_get_sort_order(GFileInfo * info);
extern const char *g_file_info_get_symlink_target(GFileInfo * info);
extern GType g_file_info_get_type(void);
extern gboolean g_file_info_has_attribute(GFileInfo * info,
        const char *attribute);
extern gboolean g_file_info_has_namespace(GFileInfo * info,
        const char *name_space);
extern void g_file_info_list_attributes(GFileInfo * info,
        const char *name_space);
extern GFileInfo *g_file_info_new(void);
extern void g_file_info_remove_attribute(GFileInfo * info,
        const char *attribute);
extern void g_file_info_set_attribute(GFileInfo * info,
        const char *attribute,
        GFileAttributeType type,
        void *value_p);
extern void g_file_info_set_attribute_boolean(GFileInfo * info,
        const char *attribute,
        gboolean attr_value);
extern void g_file_info_set_attribute_byte_string(GFileInfo * info,
        const char *attribute,
        const char *attr_value);
extern void g_file_info_set_attribute_int32(GFileInfo * info,
        const char *attribute,
        gint32 attr_value);
extern void g_file_info_set_attribute_int64(GFileInfo * info,
        const char *attribute,
        gint64 attr_value);
extern void g_file_info_set_attribute_mask(GFileInfo * info,
        GFileAttributeMatcher * mask);
extern void g_file_info_set_attribute_object(GFileInfo * info,
        const char *attribute,
        GObject * attr_value);
extern gboolean g_file_info_set_attribute_status(GFileInfo * info,
        const char *attribute,
        GFileAttributeStatus
        status);
extern void g_file_info_set_attribute_string(GFileInfo * info,
        const char *attribute,
        const char *attr_value);
extern void g_file_info_set_attribute_stringv(GFileInfo * info,
        const char *attribute,
        char **attr_value);
extern void g_file_info_set_attribute_uint32(GFileInfo * info,
        const char *attribute,
        guint32 attr_value);
extern void g_file_info_set_attribute_uint64(GFileInfo * info,
        const char *attribute,

```

```

        guint64 attr_value);
extern void g_file_info_set_content_type(GFileInfo * info,
        const char *content_type);
extern void g_file_info_set_display_name(GFileInfo * info,
        const char *display_name);
extern void g_file_info_set_edit_name(GFileInfo * info,
        const char *edit_name);
extern void g_file_info_set_file_type(GFileInfo * info, GFileType
type);
extern void g_file_info_set_icon(GFileInfo * info, GIcon * icon);
extern void g_file_info_set_is_hidden(GFileInfo * info,
        gboolean is_hidden);
extern void g_file_info_set_is_symlink(GFileInfo * info,
        gboolean is_symlink);
extern void g_file_info_set_modification_time(GFileInfo * info,
        GTimeVal * mtime);
extern void g_file_info_set_name(GFileInfo * info, const char
*name);
extern void g_file_info_set_size(GFileInfo * info, goffset size);
extern void g_file_info_set_sort_order(GFileInfo * info,
        gint32 sort_order);
extern void g_file_info_set_symlink_target(GFileInfo * info,
        const char *symlink_target);
extern void g_file_info_unset_attribute_mask(GFileInfo * info);

```

17.12.61 glib-2.0/gio/gfileinputstream.h

```

#define G_TYPE_FILE_INPUT_STREAM
(g_file_input_stream_get_type ())
#define G_FILE_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_FILE_INPUT_STREAM,
GFileInputStreamClass))
#define G_IS_FILE_INPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_FILE_INPUT_STREAM))
#define G_FILE_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_FILE_INPUT_STREAM, GFileInputStream))
#define G_IS_FILE_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_FILE_INPUT_STREAM))
#define G_FILE_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_FILE_INPUT_STREAM,
GFileInputStreamClass))

typedef struct _GFileInputStreamClass {
    GInputStreamClass parent_class;
    goffset(*tell) (GFileInputStream * stream);
    gboolean(*can_seek) (GFileInputStream * stream);
    gboolean(*seek) (GFileInputStream * stream, goffset offset,
        GSeekType type, GCancellable * cancellable,
        GError * *error);
    GFileInfo *(*query_info) (GFileInputStream * stream,
        const char *attributes,
        GCancellable * cancellable, GError *
*error);
    void (*query_info_async) (GFileInputStream * stream,
        const char *attributes, int io_priority,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
    GFileInfo *(*query_info_finish) (GFileInputStream * stream,
        GAsyncResult * result,
        GError * *error);

    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);

```

```

    void (*_g_reserved5) (void);
} GFileInputStreamClass;
typedef struct _GFileInputStreamPrivate GFileInputStreamPrivate;
struct _GFileInputStream {
    GInputStream parent_instance;
    GFileInputStreamPrivate *priv;
};
extern GType g_file_input_stream_get_type(void);
extern GFileInfo *g_file_input_stream_query_info(GFileInputStream
* stream,

                                                const char *attributes,
                                                GCancellable *
cancellable,
                                                GError * *error);
extern void g_file_input_stream_query_info_async(GFileInputStream
* stream,

                                                const char *attributes,
                                                int io_priority,
                                                GCancellable *
cancellable,
                                                GAsyncReadyCallback
callback,
                                                gpointer user_data);
extern
                                                GFileInfo
*g_file_input_stream_query_info_finish(GFileInputStream *
stream,
GAsyncResult *
result,
GError * *error);

```

17.12.62 glib-2.0/gio/gfileiostream.h

```

#define G_TYPE_FILE_IO_STREAM (g_file_io_stream_get_type ())
#define G_FILE_IO_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_FILE_IO_STREAM_CLASS, G_TYPE_FILE_IO_STREAM, GFileIOStreamClass))
#define G_IS_FILE_IO_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_FILE_IO_STREAM))
#define G_FILE_IO_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_FILE_IO_STREAM, GFileIOStream))
#define G_IS_FILE_IO_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_FILE_IO_STREAM))
#define G_FILE_IO_STREAM_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_FILE_IO_STREAM, GFileIOStreamClass))

typedef struct _GFileIOStreamClass {
    GIOStreamClass parent_class;
    goffset(*tell) (GFileIOStream * stream);
    gboolean(*can_seek) (GFileIOStream * stream);
    gboolean(*seek) (GFileIOStream * stream, goffset offset, GSeekType type, GCancellable * cancellable, GError * *error);
    gboolean(*can_truncate) (GFileIOStream * stream);
    gboolean(*truncate_fn) (GFileIOStream * stream, goffset size, GCancellable * cancellable, GError *
*error);
    GFileInfo *(*query_info) (GFileIOStream * stream, const char *attributes, GCancellable * cancellable, GError *
*error);
    void (*query_info_async) (GFileIOStream * stream, const char *attributes, int io_priority, GCancellable * cancellable, GAsyncReadyCallback callback, gpointer user_data);

```



```

GFileInfo *(*query_info_finish) (GFileIOStream * stream,
                                GAsyncResult * result,
                                GError * *error);

char *(*get_etag) (GFileIOStream * stream);
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
void (*_g_reserved5) (void);
} GFileIOStreamClass;
typedef struct _GFileIOStreamPrivate GFileIOStreamPrivate;
struct _GFileIOStream {
    GIOStream parent_instance;
    GFileIOStreamPrivate *priv;
};
extern char *g_file_io_stream_get_etag(GFileIOStream * stream);
extern GType g_file_io_stream_get_type(void);
extern GFileInfo *g_file_io_stream_query_info(GFileIOStream *
stream,

                                const char *attributes,
                                GCancellable * cancellable,
                                GError * *error);
extern void g_file_io_stream_query_info_async(GFileIOStream *
stream,

                                const char *attributes,
                                int io_priority,
                                GCancellable * cancellable,
                                GAsyncReadyCallback

                                gpointer user_data);
extern GFileInfo *g_file_io_stream_query_info_finish(GFileIOStream
*
                                stream,
                                GAsyncResult * result,
                                GError * *error);

```

17.12.63 glib-2.0/gio/gfilemonitor.h

```

#define G_TYPE_FILE_MONITOR (g_file_monitor_get_type ())
#define G_FILE_MONITOR_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_FILE_MONITOR, GFileMonitorClass))
#define G_IS_FILE_MONITOR_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_FILE_MONITOR))
#define G_FILE_MONITOR(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_FILE_MONITOR, GFileMonitor))
#define G_IS_FILE_MONITOR(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_FILE_MONITOR))
#define G_FILE_MONITOR_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_FILE_MONITOR, GFileMonitorClass))
typedef struct _GFileMonitorClass {
    GObjectClass parent_class;
    void (*changed) (GFileMonitor * monitor, GFile * file,
                    GFile * other_file, GFileMonitorEvent event_type);
    gboolean(*cancel) (GFileMonitor * monitor);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GFileMonitorClass;
typedef struct _GFileMonitorPrivate GFileMonitorPrivate;
struct _GFileMonitor {
    GObject parent_instance;
    GFileMonitorPrivate *priv;

```

```

};
extern gboolean g_file_monitor_cancel(GFileMonitor * monitor);
extern void g_file_monitor_emit_event(GFileMonitor * monitor,
                                     GFile * child, GFile * other_file,
                                     GFileMonitorEvent event_type);
extern GType g_file_monitor_get_type(void);
extern gboolean g_file_monitor_is_cancelled(GFileMonitor *
monitor);
extern void g_file_monitor_set_rate_limit(GFileMonitor * monitor,
                                         gint limit_msecs);

```

17.12.64 glib-2.0/gio/gfilenamecompleter.h

```

#define G_TYPE_FILENAME_COMPLETER G_TYPE_FILENAME_COMPLETER
(g_filename_completer_get_type ())
#define G_FILENAME_COMPLETER_CLASS(k) G_FILENAME_COMPLETER_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_FILENAME_COMPLETER,
GFilenameCompleterClass))
#define G_IS_FILENAME_COMPLETER_CLASS(k) G_IS_FILENAME_COMPLETER_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_FILENAME_COMPLETER))
#define G_FILENAME_COMPLETER(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_FILENAME_COMPLETER, GFilenameCompleter))
#define G_IS_FILENAME_COMPLETER(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_FILENAME_COMPLETER))
#define G_FILENAME_COMPLETER_GET_CLASS(o) G_FILENAME_COMPLETER_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_FILENAME_COMPLETER,
GFilenameCompleterClass))

typedef struct _GFilenameCompleterClass {
    GObjectClass parent_class;
    void (*got_completion_data) (GFilenameCompleter *
filename_completer);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
} GFilenameCompleterClass;
extern char
*g_filename_completer_get_completion_suffix(GFilenameCompleter
* completer,
const char
*initial_text);
extern void
g_filename_completer_get_completions(GFilenameCompleter *
completer,
const char *initial_text);
extern GType g_filename_completer_get_type(void);
extern GFilenameCompleter *g_filename_completer_new(void);
extern void g_filename_completer_set_dirs_only(GFilenameCompleter
* completer,
gboolean dirs_only);

```

17.12.65 glib-2.0/gio/gfileoutputstream.h

```

#define G_TYPE_FILE_OUTPUT_STREAM G_TYPE_FILE_OUTPUT_STREAM
(g_file_output_stream_get_type ())
#define G_FILE_OUTPUT_STREAM_CLASS(k) G_FILE_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_FILE_OUTPUT_STREAM,
GFileOutputStreamClass))
#define G_IS_FILE_OUTPUT_STREAM_CLASS(k) G_IS_FILE_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_FILE_OUTPUT_STREAM))
#define G_FILE_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_FILE_OUTPUT_STREAM, GFileOutputStream))

```

```

#define G_IS_FILE_OUTPUT_STREAM(o)          (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_FILE_OUTPUT_STREAM))
#define G_FILE_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_FILE_OUTPUT_STREAM,
GFileOutputStreamClass))

typedef struct _GFileOutputStreamClass {
    GOutputStreamClass parent_class;
    goffset(*tell) (GFileOutputStream * stream);
    gboolean(*can_seek) (GFileOutputStream * stream);
    gboolean(*seek) (GFileOutputStream * stream, goffset offset,
                    GSeekType type, GCancellable * cancellable,
                    GError * *error);
    gboolean(*can_truncate) (GFileOutputStream * stream);
    gboolean(*truncate_fn) (GFileOutputStream * stream, goffset
size,
                        GCancellable * cancellable, GError *
*error);
    GFileInfo *(*query_info) (GFileOutputStream * stream,
const char *attributes,
                        GCancellable * cancellable, GError *
*error);
    void (*query_info_async) (GFileOutputStream * stream,
const char *attributes, int io_priority,
                        GCancellable * cancellable,
                        GAsyncReadyCallback callback,
                        gpointer user_data),
    GFileInfo *(*query_info_finish) (GFileOutputStream * stream,
                        GAsyncResult * result,
                        GError * *error);
    char *(*get_etag) (GFileOutputStream * stream);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GFileOutputStreamClass;
typedef struct _GFileOutputStreamPrivate GFileOutputStreamPrivate;
struct _GFileOutputStream {
    GOutputStream parent_instance;
    GFileOutputStreamPrivate *priv;
};
extern char *g_file_output_stream_get_etag(GFileOutputStream *
stream);
extern GType g_file_output_stream_get_type(void);
extern GFileInfo
*g_file_output_stream_query_info(GFileOutputStream *
stream,
const char *attributes,
GCancellable *
cancellable,
GError * *error);
extern void
g_file_output_stream_query_info_async(GFileOutputStream *
stream,
const char *attributes,
int io_priority,
GCancellable *
cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);
extern GFileInfo
*g_file_output_stream_query_info_finish(GFileOutputStream
* stream,
GAsyncResult *

```

```
result,
GError * *error);
```

17.12.66 glib-2.0/gio/gfilterinputstream.h

```
#define G_TYPE_FILTER_INPUT_STREAM
(g_filter_input_stream_get_type ())
#define G_FILTER_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_FILTER_INPUT_STREAM,
GFilterInputStreamClass))
#define G_IS_FILTER_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_FILTER_INPUT_STREAM))
#define G_FILTER_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_FILTER_INPUT_STREAM, GFilterInputStream))
#define G_IS_FILTER_INPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_FILTER_INPUT_STREAM))
#define G_FILTER_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_FILTER_INPUT_STREAM,
GFilterInputStreamClass))

typedef struct _GFilterInputStreamClass {
    GInputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
} GFilterInputStreamClass;
struct _GFilterInputStream {
    GInputStream parent_instance;
    GInputStream *base_stream;
};
extern GInputStream
*g_filter_input_stream_get_base_stream(GFilterInputStream *
stream);
extern gboolean
g_filter_input_stream_get_close_base_stream(GFilterInputStream *
stream);
extern GType g_filter_input_stream_get_type(void);
extern void
g_filter_input_stream_set_close_base_stream(GFilterInputStream
* stream,
gboolean
close_base);
```

17.12.67 glib-2.0/gio/gfilteroutputstream.h

```
#define G_TYPE_FILTER_OUTPUT_STREAM
(g_filter_output_stream_get_type ())
#define G_FILTER_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_FILTER_OUTPUT_STREAM,
GFilterOutputStreamClass))
#define G_IS_FILTER_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_FILTER_OUTPUT_STREAM))
#define G_FILTER_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_FILTER_OUTPUT_STREAM, GFilterOutputStream))
#define G_IS_FILTER_OUTPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_FILTER_OUTPUT_STREAM))
#define G_FILTER_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_FILTER_OUTPUT_STREAM,
GFilterOutputStreamClass))

typedef struct _GFilterOutputStreamClass {
    GOutputStreamClass parent_class;
    void (*_g_reserved1) (void);
```

```

    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
} GFilterOutputStreamClass;
struct _GFilterOutputStream {
    GOutputStream parent_instance;
    GOutputStream *base_stream;
};
extern GOutputStream
    *g_filter_output_stream_get_base_stream(GFilterOutputStream *
    stream);
extern gboolean
    g_filter_output_stream_get_close_base_stream(GFilterOutputStream *
    stream);
extern GType g_filter_output_stream_get_type(void);
extern void
    g_filter_output_stream_set_close_base_stream(GFilterOutputStream *
    stream,
                                           gboolean close_base);

```

17.12.68 glib-2.0/gio/gicon.h

```

#define G_TYPE_ICON      (g_icon_get_type ())
#define G_ICON(obj)      (G_TYPE_CHECK_INSTANCE_CAST ((obj),
    G_TYPE_ICON, GIcon))
#define G_IS_ICON(obj)    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
    G_TYPE_ICON))
#define G_ICON_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE
    ((obj), G_TYPE_ICON, GIconIface))

typedef struct _GIconIface {
    GTypeInterface g_iface;
    guint(*hash) (GIcon * icon);
    gboolean(*equal) (GIcon * icon1, GIcon * icon2);
    gboolean(*to_tokens) (GIcon * icon, GPtrArray * tokens,
        gint* out_version);
    GIcon *(*from_tokens) (GIcon * icon, gint num_tokens, gint
    version,
        GError * *error);
} GIconIface;
extern gboolean g_icon_equal(GIcon * icon1, GIcon * icon2);
extern GType g_icon_get_type(void);
extern guint g_icon_hash(gconstpointer icon);
extern GIcon *g_icon_new_for_string(const gchar * str, GError *
    *error);
extern gchar *g_icon_to_string(GIcon * icon);

```

17.12.69 glib-2.0/gio/ginetaddress.h

```

#define G_TYPE_INET_ADDRESS (g_inet_address_get_type ())
#define G_INET_ADDRESS_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
    G_TYPE_INET_ADDRESS, GInetAddressClass))
#define G_IS_INET_ADDRESS_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
    ((k), G_TYPE_INET_ADDRESS))
#define G_INET_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
    G_TYPE_INET_ADDRESS, GInetAddress))
#define G_IS_INET_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
    G_TYPE_INET_ADDRESS))
#define G_INET_ADDRESS_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
    ((o), G_TYPE_INET_ADDRESS, GInetAddressClass))

typedef struct _GInetAddressClass {
    GObjectClass parent_class;
    gchar *(*to_string) (GInetAddress * address);

```

```

    const guint8 *(*to_bytes) (GInetAddress * address);
} GInetAddressClass;
typedef struct _GInetAddressPrivate GInetAddressPrivate;
struct _GInetAddress {
    GObject parent_instance;
    GInetAddressPrivate *priv;
};
extern gboolean g_inet_address_equal(GInetAddress * address,
                                     GInetAddress * other_address);
extern GSocketFamily g_inet_address_get_family(GInetAddress *
address);
extern gboolean g_inet_address_get_is_any(GInetAddress * address);
extern gboolean g_inet_address_get_is_link_local(GInetAddress *
address);
extern gboolean g_inet_address_get_is_loopback(GInetAddress *
address);
extern gboolean g_inet_address_get_is_mc_global(GInetAddress *
address);
extern gboolean g_inet_address_get_is_mc_link_local(GInetAddress *
address);
extern gboolean g_inet_address_get_is_mc_node_local(GInetAddress *
address);
extern gboolean g_inet_address_get_is_mc_org_local(GInetAddress *
address);
extern gboolean g_inet_address_get_is_mc_site_local(GInetAddress *
address);
extern gboolean g_inet_address_get_is_multicast(GInetAddress *
address);
extern gboolean g_inet_address_get_is_site_local(GInetAddress *
address);
extern gsize g_inet_address_get_native_size(GInetAddress *
address);
extern GType g_inet_address_get_type(void);
extern GInetAddress *g_inet_address_new_any(GSocketFamily family);
extern GInetAddress *g_inet_address_new_from_bytes(const guint8 *
bytes,
                                                    GSocketFamily family);
extern GInetAddress *g_inet_address_new_from_string(const gchar *
string);
extern GInetAddress *g_inet_address_new_loopback(GSocketFamily
family);
extern const unsigned char *g_inet_address_to_bytes(GInetAddress *
address);
extern gchar *g_inet_address_to_string(GInetAddress * address);

```

17.12.70 glib-2.0/gio/ginetaddressmask.h

```

#define G_TYPE_INET_ADDRESS_MASK
(g_inet_address_mask_get_type ())
#define G_INET_ADDRESS_MASK_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_INET_ADDRESS_MASK,
GInetAddressMaskClass))
#define G_IS_INET_ADDRESS_MASK_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_INET_ADDRESS_MASK))
#define G_INET_ADDRESS_MASK(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_INET_ADDRESS_MASK, GInetAddressMask))
#define G_IS_INET_ADDRESS_MASK(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_INET_ADDRESS_MASK))
#define G_INET_ADDRESS_MASK_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_INET_ADDRESS_MASK,
GInetAddressMaskClass))

typedef struct _GInetAddressMaskClass {
    GObjectClass parent_class;
} GInetAddressMaskClass;

```

```

typedef struct _GInetAddressMaskPrivate GInetAddressMaskPrivate;
struct _GInetAddressMask {
    GObject parent_instance;
    GInetAddressMaskPrivate *priv;
};
extern gboolean g_inet_address_mask_equal(GInetAddressMask * mask,
                                           GInetAddressMask * mask2);
extern
                                           GInetAddress
*g_inet_address_mask_get_address(GInetAddressMask *
                                mask);
extern
                                           GSocketFamily
g_inet_address_mask_get_family(GInetAddressMask *
                                mask);
extern guint g_inet_address_mask_get_length(GInetAddressMask *
mask);
extern GType g_inet_address_mask_get_type(void);
extern gboolean g_inet_address_mask_matches(GInetAddressMask *
mask,
                                           GInetAddress * address);
extern GInetAddressMask *g_inet_address_mask_new(GInetAddress *
addr,
                                           guint length,
                                           GError * *error);
extern GInetAddressMask *g_inet_address_mask_new_from_string(const
gchar *
                                           mask_string,
                                           GError *
                                           *error);
extern gchar *g_inet_address_mask_to_string(GInetAddressMask *
mask);

```

17.12.71 glib-2.0/gio/ginetsocketaddress.h

```

#define G_TYPE_INET_SOCKET_ADDRESS
(g_inet_socket_address_get_type ())
#define G_INET_SOCKET_ADDRESS_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k),
                           G_TYPE_INET_SOCKET_ADDRESS,
                           GInetAddressClass))
#define G_IS_INET_SOCKET_ADDRESS_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_INET_SOCKET_ADDRESS))
#define G_INET_SOCKET_ADDRESS(o)
((o), G_TYPE_INET_SOCKET_ADDRESS, GInetAddressClass)
#define G_IS_INET_SOCKET_ADDRESS(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_INET_SOCKET_ADDRESS))
#define G_INET_SOCKET_ADDRESS_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_INET_SOCKET_ADDRESS,
                              GInetAddressClass))

typedef struct _GInetAddressClass {
    GSocketAddressClass parent_class;
} GInetAddressClass;
typedef struct _GInetAddressPrivate
GInetAddressPrivate;
struct _GInetAddress {
    GSocketAddress parent_instance;
    GInetAddressPrivate *priv;
};
extern
                                           GInetAddress
*g_inet_socket_address_get_address(GInetAddress *
                                address);
extern
                                           guint32
g_inet_socket_address_get_flowinfo(GInetAddress *
                                address);
extern guint16 g_inet_socket_address_get_port(GInetAddress *
                                address);

```

```

extern
g_inet_socket_address_get_scope_id(GInetSocketAddress *
                                   address);
extern GType g_inet_socket_address_get_type(void);
extern GSocketAddress *g_inet_socket_address_new(GInetAddress *
address,
                                                  guint16 port);

```

17.12.72 glib-2.0/gio/ginitable.h

```

#define G_TYPE_INITABLE (g_initable_get_type ())
#define G_INITABLE(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_INITABLE, GInitable))
#define G_IS_INITABLE(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_INITABLE))
#define G_INITABLE_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_INITABLE,
GInitableIface))
#define G_TYPE_IS_INITABLE(type) (g_type_is_a ((type),
G_TYPE_INITABLE))

typedef struct _GInitableIface {
    GTypeInterface g_iface;
    gboolean(*init)(GInitable * initable, GCancellable *
cancellable,
                  GError * *error);
} GInitableIface;
extern GType g_initable_get_type(void);
extern gboolean g_initable_init(GInitable * initable,
                                GCancellable * cancellable,
                                GError * *error);
extern gpointer g_initable_new(GType object_type,
                                GCancellable * cancellable, GError *
*error,
                                const gchar * first_property_name, ...);
extern GObject *g_initable_new_valist(GType object_type,
                                       const gchar * first_property_name,
                                       va_list var_args,
                                       GCancellable * cancellable,
                                       GError * *error);
extern gpointer g_initable_newv(GType object_type, guint
n_parameters,
                                GParameter * parameters,
                                GCancellable * cancellable,
                                GError * *error);

```

17.12.73 glib-2.0/gio/ginputstream.h

```

#define G_TYPE_INPUT_STREAM (g_input_stream_get_type ())
#define G_INPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_INPUT_STREAM, GInputStreamClass))
#define G_IS_INPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_INPUT_STREAM))
#define G_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_INPUT_STREAM, GInputStream))
#define G_IS_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_INPUT_STREAM))
#define G_INPUT_STREAM_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_INPUT_STREAM, GInputStreamClass))

typedef struct _GInputStreamClass {
    GObjectClass parent_class;

```



```

    gssize(*read_fn) (GInputStream * stream, void *buffer, gsize
count,
                    Gancellable * cancellable, GError * *error);
    gssize(*skip) (GInputStream * stream, gsize count,
                  Gancellable * cancellable, GError * *error);
    gboolean(*close_fn) (GInputStream * stream,
                        Gancellable * cancellable, GError * *error);
    void (*read_async) (GInputStream * stream, void *buffer, gsize
count,
                      int io_priority, Gancellable * cancellable,
                      GAsyncReadyCallback callback, gpointer
user_data);
    gssize(*read_finish) (GInputStream * stream, GAsyncResult *
result,
                        GError * *error);
    void (*skip_async) (GInputStream * stream, gsize count,
                      int io_priority, Gancellable * cancellable,
                      GAsyncReadyCallback callback, gpointer
user_data);
    gssize(*skip_finish) (GInputStream * stream, GAsyncResult *
result,
                        GError * *error);
    void (*close_async) (GInputStream * stream, int io_priority,
                        Gancellable * cancellable,
                        GAsyncReadyCallback callback, gpointer
user_data);
    gboolean(*close_finish) (GInputStream * stream, GAsyncResult *
result,
                            GError * *error);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GInputStreamClass;
typedef struct _GInputStreamPrivate GInputStreamPrivate;
struct _GInputStream {
    GObject parent_instance;
    GInputStreamPrivate *priv;
};
extern void g_input_stream_clear_pending(GInputStream * stream);
extern gboolean g_input_stream_close(GInputStream * stream,
                                     Gancellable * cancellable,
                                     GError * *error);
extern void g_input_stream_close_async(GInputStream * stream,
                                       int io_priority,
                                       Gancellable * cancellable,
                                       GAsyncReadyCallback callback,
                                       gpointer user_data);
extern gboolean g_input_stream_close_finish(GInputStream * stream,
                                           GAsyncResult * result,
                                           GError * *error);
extern GType g_input_stream_get_type(void);
extern gboolean g_input_stream_has_pending(GInputStream * stream);
extern gboolean g_input_stream_is_closed(GInputStream * stream);
extern gssize g_input_stream_read(GInputStream * stream, void
*buffer,
                                gsize count, Gancellable *
cancellable,
                                GError * *error);
extern gboolean g_input_stream_read_all(GInputStream * stream,
                                       void *buffer, gsize count,
                                       gsize * bytes_read,
                                       Gancellable * cancellable,
                                       GError * *error);

```

```

extern void g_input_stream_read_async(GInputStream * stream, void
*buffer,
                                gsize count, int io_priority,
                                GCancellable * cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
extern gssize g_input_stream_read_finish(GInputStream * stream,
                                GAsyncResult * result,
                                GError * *error);
extern gboolean g_input_stream_set_pending(GInputStream * stream,
                                GError * *error);
extern gssize g_input_stream_skip(GInputStream * stream, gsize
count,
                                GCancellable * cancellable,
                                GError * *error);
extern void g_input_stream_skip_async(GInputStream * stream, gsize
count,
                                int io_priority,
                                GCancellable * cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
extern gssize g_input_stream_skip_finish(GInputStream * stream,
                                GAsyncResult * result,
                                GError * *error);

```

17.12.74 glib-2.0/gio/gio.h

```
#define __GIO_GIO_H_INSIDE__
```

17.12.75 glib-2.0/gio/gioenums.h

```

typedef enum {
    G_APP_INFO_CREATE_NONE,
    G_APP_INFO_CREATE_NEEDS_TERMINAL = (1 << 0),
    G_APP_INFO_CREATE_SUPPORTS_URI = (1 << 1),
    G_APP_INFO_CREATE_SUPPORTS_STARTUP_NOTIFICATION = (1 << 2)
} GAppInfoCreateFlags;
typedef enum {
    G_CONVERTER_NO_FLAGS,
    G_CONVERTER_INPUT_AT_END = (1 << 0),
    G_CONVERTER_FLUSH = (1 << 1)
} GConverterFlags;
typedef enum {
    G_CONVERTER_ERROR,
    G_CONVERTER_CONVERTED,
    G_CONVERTER_FINISHED,
    G_CONVERTER_FLUSHED
} GConverterResult;
typedef enum {
    G_DATA_STREAM_BYTE_ORDER_BIG_ENDIAN,
    G_DATA_STREAM_BYTE_ORDER_LITTLE_ENDIAN,
    G_DATA_STREAM_BYTE_ORDER_HOST_ENDIAN
} GDataStreamByteOrder;
typedef enum {
    G_DATA_STREAM_NEWLINE_TYPE_LF,
    G_DATA_STREAM_NEWLINE_TYPE_CR,
    G_DATA_STREAM_NEWLINE_TYPE_CR_LF,
    G_DATA_STREAM_NEWLINE_TYPE_ANY
} GDataStreamNewlineType;
typedef enum {
    G_FILE_ATTRIBUTE_TYPE_INVALID,
    G_FILE_ATTRIBUTE_TYPE_STRING,
    G_FILE_ATTRIBUTE_TYPE_BYTE_STRING,

```

```

    G_FILE_ATTRIBUTE_TYPE_BOOLEAN,
    G_FILE_ATTRIBUTE_TYPE_UINT32,
    G_FILE_ATTRIBUTE_TYPE_INT32,
    G_FILE_ATTRIBUTE_TYPE_UINT64,
    G_FILE_ATTRIBUTE_TYPE_INT64,
    G_FILE_ATTRIBUTE_TYPE_OBJECT,
    G_FILE_ATTRIBUTE_TYPE_STRINGV
} GFileAttributeType;
typedef enum {
    G_FILE_ATTRIBUTE_INFO_NONE,
    G_FILE_ATTRIBUTE_INFO_COPY_WITH_FILE = (1 << 0),
    G_FILE_ATTRIBUTE_INFO_COPY_WHEN_MOVED = (1 << 1)
} GFileAttributeInfoFlags;
typedef enum {
    G_FILE_ATTRIBUTE_STATUS_UNSET,
    G_FILE_ATTRIBUTE_STATUS_SET,
    G_FILE_ATTRIBUTE_STATUS_ERROR_SETTING
} GFileAttributeStatus;
typedef enum {
    G_FILE_QUERY_INFO_NONE,
    G_FILE_QUERY_INFO_NOFOLLOW_SYMLINKS = (1 << 0)
} GFileQueryInfoFlags;
typedef enum {
    G_FILE_CREATE_NONE,
    G_FILE_CREATE_PRIVATE = (1 << 0),
    G_FILE_CREATE_REPLACE_DESTINATION = (1 << 1)
} GFileCreateFlags;
typedef enum {
    G_MOUNT_MOUNT_NONE
} GMountMountFlags;
typedef enum {
    G_MOUNT_UNMOUNT_NONE,
    G_MOUNT_UNMOUNT_FORCE = (1 << 0)
} GMountUnmountFlags;
typedef enum {
    G_DRIVE_START_NONE
} GDriveStartFlags;
typedef enum {
    G_DRIVE_START_STOP_TYPE_UNKNOWN,
    G_DRIVE_START_STOP_TYPE_SHUTDOWN,
    G_DRIVE_START_STOP_TYPE_NETWORK,
    G_DRIVE_START_STOP_TYPE_MULTIDISK,
    G_DRIVE_START_STOP_TYPE_PASSWORD
} GDriveStartStopType;
typedef enum {
    G_FILE_COPY_NONE,
    G_FILE_COPY_OVERWRITE = (1 << 0),
    G_FILE_COPY_BACKUP = (1 << 1),
    G_FILE_COPY_NOFOLLOW_SYMLINKS = (1 << 2),
    G_FILE_COPY_ALL_METADATA = (1 << 3),
    G_FILE_COPY_NO_FALLBACK_FOR_MOVE = (1 << 4),
    G_FILE_COPY_TARGET_DEFAULT_PERMS = (1 << 5)
} GFileCopyFlags;
typedef enum {
    G_FILE_MONITOR_NONE,
    G_FILE_MONITOR_WATCH_MOUNTS = (1 << 0),
    G_FILE_MONITOR_SEND_MOVED = (1 << 1)
} GFileMonitorFlags;
typedef enum {
    G_FILE_TYPE_UNKNOWN,
    G_FILE_TYPE_REGULAR,
    G_FILE_TYPE_DIRECTORY,
    G_FILE_TYPE_SYMBOLIC_LINK,
    G_FILE_TYPE_SPECIAL,
    G_FILE_TYPE_SHORTCUT,
    G_FILE_TYPE_MOUNTABLE

```

```

} GFileType;
typedef enum {
    G_FILESYSTEM_PREVIEW_TYPE_IF_ALWAYS,
    G_FILESYSTEM_PREVIEW_TYPE_IF_LOCAL,
    G_FILESYSTEM_PREVIEW_TYPE_NEVER
} GFilesystemPreviewType;
typedef enum {
    G_FILE_MONITOR_EVENT_CHANGED,
    G_FILE_MONITOR_EVENT_CHANGES_DONE_HINT,
    G_FILE_MONITOR_EVENT_DELETED,
    G_FILE_MONITOR_EVENT_CREATED,
    G_FILE_MONITOR_EVENT_ATTRIBUTE_CHANGED,
    G_FILE_MONITOR_EVENT_PRE_UNMOUNT,
    G_FILE_MONITOR_EVENT_UNMOUNTED,
    G_FILE_MONITOR_EVENT_MOVED
} GFileMonitorEvent;
typedef enum {
    G_IO_ERROR_FAILED,
    G_IO_ERROR_NOT_FOUND,
    G_IO_ERROR_EXISTS,
    G_IO_ERROR_IS_DIRECTORY,
    G_IO_ERROR_NOT_DIRECTORY,
    G_IO_ERROR_NOT_EMPTY,
    G_IO_ERROR_NOT_REGULAR_FILE,
    G_IO_ERROR_NOT_SYMBOLIC_LINK,
    G_IO_ERROR_NOT_MOUNTABLE_FILE,
    G_IO_ERROR_FILENAME_TOO_LONG,
    G_IO_ERROR_INVALID_FILENAME,
    G_IO_ERROR_TOO_MANY_LINKS,
    G_IO_ERROR_NO_SPACE,
    G_IO_ERROR_INVALID_ARGUMENT,
    G_IO_ERROR_PERMISSION_DENIED,
    G_IO_ERROR_NOT_SUPPORTED,
    G_IO_ERROR_NOT_MOUNTED,
    G_IO_ERROR_ALREADY_MOUNTED,
    G_IO_ERROR_CLOSED,
    G_IO_ERROR_CANCELLED,
    G_IO_ERROR_PENDING,
    G_IO_ERROR_READ_ONLY,
    G_IO_ERROR_CANT_CREATE_BACKUP,
    G_IO_ERROR_WRONG_ETAG,
    G_IO_ERROR_TIMED_OUT,
    G_IO_ERROR_WOULD_RECURSE,
    G_IO_ERROR_BUSY,
    G_IO_ERROR_WOULD_BLOCK,
    G_IO_ERROR_HOST_NOT_FOUND,
    G_IO_ERROR_WOULD_MERGE,
    G_IO_ERROR_FAILED_HANDLED,
    G_IO_ERROR_TOO_MANY_OPEN_FILES,
    G_IO_ERROR_NOT_INITIALIZED,
    G_IO_ERROR_ADDRESS_IN_USE,
    G_IO_ERROR_PARTIAL_INPUT,
    G_IO_ERROR_INVALID_DATA,
    G_IO_ERROR_DBUS_ERROR,
    G_IO_ERROR_HOST_UNREACHABLE,
    G_IO_ERROR_NETWORK_UNREACHABLE,
    G_IO_ERROR_CONNECTION_REFUSED,
    G_IO_ERROR_PROXY_FAILED,
    G_IO_ERROR_PROXY_AUTH_FAILED,
    G_IO_ERROR_PROXY_NEED_AUTH,
    G_IO_ERROR_PROXY_NOT_ALLOWED
} GIOErrorEnum;
typedef enum {
    G_ASK_PASSWORD_NEED_PASSWORD = (1 << 0),
    G_ASK_PASSWORD_NEED_USERNAME = (1 << 1),
    G_ASK_PASSWORD_NEED_DOMAIN = (1 << 2),

```

```

    G_ASK_PASSWORD_SAVING_SUPPORTED = (1 << 3),
    G_ASK_PASSWORD_ANONYMOUS_SUPPORTED = (1 << 4)
} GAskPasswordFlags;
typedef enum {
    G_PASSWORD_SAVE_NEVER,
    G_PASSWORD_SAVE_FOR_SESSION,
    G_PASSWORD_SAVE_PERMANENTLY
} GPasswordSave;
typedef enum {
    G_MOUNT_OPERATION_HANDLED,
    G_MOUNT_OPERATION_ABORTED,
    G_MOUNT_OPERATION_UNHANDLED
} GMountOperationResult;
typedef enum {
    G_OUTPUT_STREAM_SPLICE_NONE,
    G_OUTPUT_STREAM_SPLICE_CLOSE_SOURCE = (1 << 0),
    G_OUTPUT_STREAM_SPLICE_CLOSE_TARGET = (1 << 1)
} GOutputStreamSpliceFlags;
typedef enum {
    G_IO_STREAM_SPLICE_NONE,
    G_IO_STREAM_SPLICE_CLOSE_STREAM1 = (1 << 0),
    G_IO_STREAM_SPLICE_CLOSE_STREAM2 = (1 << 1),
    G_IO_STREAM_SPLICE_WAIT_FOR_BOTH = (1 << 2)
} GIOStreamSpliceFlags;
typedef enum {
    G_EMBLEM_ORIGIN_UNKNOWN,
    G_EMBLEM_ORIGIN_DEVICE,
    G_EMBLEM_ORIGIN_LIVEMETADATA,
    G_EMBLEM_ORIGIN_TAG
} GEmblemOrigin;
typedef enum {
    G_RESOLVER_ERROR_NOT_FOUND,
    G_RESOLVER_ERROR_TEMPORARY_FAILURE,
    G_RESOLVER_ERROR_INTERNAL
} GResolverError;
typedef enum {
    G_RESOURCE_ERROR_NOT_FOUND,
    G_RESOURCE_ERROR_INTERNAL
} GResourceError;
typedef enum {
    G_RESOURCE_FLAGS_NONE,
    G_RESOURCE_FLAGS_COMPRESSED = (1 << 0)
} GResourceFlags;
typedef enum {
    G_RESOURCE_LOOKUP_FLAGS_NONE
} GResourceLookupFlags;
typedef enum {
    G_SOCKET_FAMILY_INVALID,
    G_SOCKET_FAMILY_UNIX,
    G_SOCKET_FAMILY_IPV4,
    G_SOCKET_FAMILY_IPV6
} GSocketFamily;
typedef enum {
    G_SOCKET_TYPE_INVALID,
    G_SOCKET_TYPE_STREAM,
    G_SOCKET_TYPE_DATAGRAM,
    G_SOCKET_TYPE_SEQPACKET
} GSocketType;
typedef enum {
    G_SOCKET_MSG_NONE,
    G_SOCKET_MSG_OOB,
    G_SOCKET_MSG_PEEK,
    G_SOCKET_MSG_DONTROUTE
} GSocketMsgFlags;
typedef enum {
    G_SOCKET_PROTOCOL_UNKNOWN = -1,

```

```

        G_SOCKET_PROTOCOL_DEFAULT,
        G_SOCKET_PROTOCOL_TCP,
        G_SOCKET_PROTOCOL_UDP,
        G_SOCKET_PROTOCOL_SCTP
    } GSocketProtocol;
typedef enum {
    G_ZLIB_COMPRESSOR_FORMAT_ZLIB,
    G_ZLIB_COMPRESSOR_FORMAT_GZIP,
    G_ZLIB_COMPRESSOR_FORMAT_RAW
} GZlibCompressorFormat;
typedef enum {
    G_UNIX_SOCKET_ADDRESS_INVALID,
    G_UNIX_SOCKET_ADDRESS_ANONYMOUS,
    G_UNIX_SOCKET_ADDRESS_PATH,
    G_UNIX_SOCKET_ADDRESS_ABSTRACT,
    G_UNIX_SOCKET_ADDRESS_ABSTRACT_PADDED
} GUnixSocketAddressType;
typedef enum {
    G_BUS_TYPE_STARTER = -1,
    G_BUS_TYPE_NONE,
    G_BUS_TYPE_SYSTEM,
    G_BUS_TYPE_SESSION
} GBusType;
typedef enum {
    G_BUS_NAME_OWNER_FLAGS_NONE,
    G_BUS_NAME_OWNER_FLAGS_ALLOW_REPLACEMENT = (1 << 0),
    G_BUS_NAME_OWNER_FLAGS_REPLACE = (1 << 1)
} GBusNameOwnerFlags;
typedef enum {
    G_BUS_NAME_WATCHER_FLAGS_NONE,
    G_BUS_NAME_WATCHER_FLAGS_AUTO_START = (1 << 0)
} GBusNameWatcherFlags;
typedef enum {
    G_DBUS_PROXY_FLAGS_NONE,
    G_DBUS_PROXY_FLAGS_DO_NOT_LOAD_PROPERTIES = (1 << 0),
    G_DBUS_PROXY_FLAGS_DO_NOT_CONNECT_SIGNALS = (1 << 1),
    G_DBUS_PROXY_FLAGS_DO_NOT_AUTO_START = (1 << 2),
    G_DBUS_PROXY_FLAGS_GET_INVALIDATED_PROPERTIES = (1 << 3)
} GDBusProxyFlags;
typedef enum {
    G_DBUS_ERROR_FAILED,
    G_DBUS_ERROR_NO_MEMORY,
    G_DBUS_ERROR_SERVICE_UNKNOWN,
    G_DBUS_ERROR_NAME_HAS_NO_OWNER,
    G_DBUS_ERROR_NO_REPLY,
    G_DBUS_ERROR_IO_ERROR,
    G_DBUS_ERROR_BAD_ADDRESS,
    G_DBUS_ERROR_NOT_SUPPORTED,
    G_DBUS_ERROR_LIMITS_EXCEEDED,
    G_DBUS_ERROR_ACCESS_DENIED,
    G_DBUS_ERROR_AUTH_FAILED,
    G_DBUS_ERROR_NO_SERVER,
    G_DBUS_ERROR_TIMEOUT,
    G_DBUS_ERROR_NO_NETWORK,
    G_DBUS_ERROR_ADDRESS_IN_USE,
    G_DBUS_ERROR_DISCONNECTED,
    G_DBUS_ERROR_INVALID_ARGS,
    G_DBUS_ERROR_FILE_NOT_FOUND,
    G_DBUS_ERROR_FILE_EXISTS,
    G_DBUS_ERROR_UNKNOWN_METHOD,
    G_DBUS_ERROR_TIMED_OUT,
    G_DBUS_ERROR_MATCH_RULE_NOT_FOUND,
    G_DBUS_ERROR_MATCH_RULE_INVALID,
    G_DBUS_ERROR_SPAWN_EXEC_FAILED,
    G_DBUS_ERROR_SPAWN_FORK_FAILED,
    G_DBUS_ERROR_SPAWN_CHILD_EXITED,

```

```

    G_DBUS_ERROR_SPAWN_CHILD_SINGALED,
    G_DBUS_ERROR_SPAWN_FAILED,
    G_DBUS_ERROR_SPAWN_SETUP_FAILED,
    G_DBUS_ERROR_SPAWN_CONFIG_INVALID,
    G_DBUS_ERROR_SPAWN_SERVICE_INVALID,
    G_DBUS_ERROR_SPAWN_SERVICE_NOT_FOUND,
    G_DBUS_ERROR_SPAWN_PERMISSIONS_INVALID,
    G_DBUS_ERROR_SPAWN_FILE_INVALID,
    G_DBUS_ERROR_SPAWN_NO_MEMORY,
    G_DBUS_ERROR_UNIX_PROCESS_ID_UNKNOWN,
    G_DBUS_ERROR_INVALID_SIGNATURE,
    G_DBUS_ERROR_INVALID_FILE_CONTENT,
    G_DBUS_ERROR_SELINUX_SECURITY_CONTEXT_UNKNOWN,
    G_DBUS_ERROR_ADT_AUDIT_DATA_UNKNOWN,
    G_DBUS_ERROR_OBJECT_PATH_IN_USE
} GDBusError;
typedef enum {
    G_DBUS_CONNECTION_FLAGS_NONE,
    G_DBUS_CONNECTION_FLAGS_AUTHENTICATION_CLIENT = (1 << 0),
    G_DBUS_CONNECTION_FLAGS_AUTHENTICATION_SERVER = (1 << 1),
    G_DBUS_CONNECTION_FLAGS_AUTHENTICATION_ALLOW_ANONYMOUS = (1 <<
2),
    G_DBUS_CONNECTION_FLAGS_MESSAGE_BUS_CONNECTION = (1 << 3),
    G_DBUS_CONNECTION_FLAGS_DELAY_MESSAGE_PROCESSING = (1 << 4)
} GDBusConnectionFlags;
typedef enum {
    G_DBUS_CAPABILITY_FLAGS_NONE,
    G_DBUS_CAPABILITY_FLAGS_UNIX_FD_PASSING = (1 << 0)
} GDBusCapabilityFlags;
typedef enum {
    G_DBUS_CALL_FLAGS_NONE,
    G_DBUS_CALL_FLAGS_NO_AUTO_START = (1 << 0)
} GDBusCallFlags;
typedef enum {
    G_DBUS_MESSAGE_TYPE_INVALID,
    G_DBUS_MESSAGE_TYPE_METHOD_CALL,
    G_DBUS_MESSAGE_TYPE_METHOD_RETURN,
    G_DBUS_MESSAGE_TYPE_ERROR,
    G_DBUS_MESSAGE_TYPE_SIGNAL
} GDBusMessageType;
typedef enum {
    G_DBUS_MESSAGE_FLAGS_NONE,
    G_DBUS_MESSAGE_FLAGS_NO_REPLY_EXPECTED = (1 << 0),
    G_DBUS_MESSAGE_FLAGS_NO_AUTO_START = (1 << 1)
} GDBusMessageFlags;
typedef enum {
    G_DBUS_MESSAGE_HEADER_FIELD_INVALID,
    G_DBUS_MESSAGE_HEADER_FIELD_PATH,
    G_DBUS_MESSAGE_HEADER_FIELD_INTERFACE,
    G_DBUS_MESSAGE_HEADER_FIELD_MEMBER,
    G_DBUS_MESSAGE_HEADER_FIELD_ERROR_NAME,
    G_DBUS_MESSAGE_HEADER_FIELD_REPLY_SERIAL,
    G_DBUS_MESSAGE_HEADER_FIELD_DESTINATION,
    G_DBUS_MESSAGE_HEADER_FIELD_SENDER,
    G_DBUS_MESSAGE_HEADER_FIELD_SIGNATURE,
    G_DBUS_MESSAGE_HEADER_FIELD_NUM_UNIX_FDS
} GDBusMessageHeaderField;
typedef enum {
    G_DBUS_PROPERTY_INFO_FLAGS_NONE,
    G_DBUS_PROPERTY_INFO_FLAGS_READABLE = (1 << 0),
    G_DBUS_PROPERTY_INFO_FLAGS_WRITABLE = (1 << 1)
} GDBusPropertyInfoFlags;
typedef enum {
    G_DBUS_SUBTREE_FLAGS_NONE,
    G_DBUS_SUBTREE_FLAGS_DISPATCH_TO_UNENUMERATED_NODES = (1 << 0)
} GDBusSubtreeFlags;

```

```

typedef enum {
    G_DBUS_SERVER_FLAGS_NONE,
    G_DBUS_SERVER_FLAGS_RUN_IN_THREAD = (1 << 0),
    G_DBUS_SERVER_FLAGS_AUTHENTICATION_ALLOW_ANONYMOUS = (1 << 1)
} GDBusServerFlags;
typedef enum {
    G_DBUS_SIGNAL_FLAGS_NONE,
    G_DBUS_SIGNAL_FLAGS_NO_MATCH_RULE = (1 << 0)
} GDBusSignalFlags;
typedef enum {
    G_DBUS_SEND_MESSAGE_FLAGS_NONE,
    G_DBUS_SEND_MESSAGE_FLAGS_PRESERVE_SERIAL = (1 << 0)
} GDBusSendMessageFlags;
typedef enum {
    G_CREDENTIALS_TYPE_INVALID,
    G_CREDENTIALS_TYPE_LINUX_UCRED,
    G_CREDENTIALS_TYPE_FREEBSD_CMSGCRED,
    G_CREDENTIALS_TYPE_OPENBSD_SOCKPEERCRED
} GCredentialsType;
typedef enum {
    G_DBUS_MESSAGE_BYTE_ORDER_BIG_ENDIAN,
    G_DBUS_MESSAGE_BYTE_ORDER_LITTLE_ENDIAN
} GDBusMessageByteOrder;
typedef enum {
    G_APPLICATION_FLAGS_NONE,
    G_APPLICATION_IS_SERVICE = (1 << 0),
    G_APPLICATION_IS_LAUNCHER = (1 << 1),
    G_APPLICATION_HANDLES_OPEN = (1 << 2),
    G_APPLICATION_HANDLES_COMMAND_LINE = (1 << 3),
    G_APPLICATION_SEND_ENVIRONMENT = (1 << 4),
    G_APPLICATION_NON_UNIQUE = (1 << 5)
} GApplicationFlags;
typedef enum {
    G_TLS_ERROR_UNAVAILABLE,
    G_TLS_ERROR_MISC,
    G_TLS_ERROR_BAD_CERTIFICATE,
    G_TLS_ERROR_NOT_TLS,
    G_TLS_ERROR_HANDSHAKE,
    G_TLS_ERROR_CERTIFICATE_REQUIRED,
    G_TLS_ERROR_EOF
} GTlsError;
typedef enum {
    G_TLS_CERTIFICATE_UNKNOWN_CA = (1 << 0),
    G_TLS_CERTIFICATE_BAD_IDENTITY = (1 << 1),
    G_TLS_CERTIFICATE_NOT_ACTIVATED = (1 << 2),
    G_TLS_CERTIFICATE_EXPIRED = (1 << 3),
    G_TLS_CERTIFICATE_REVOKED = (1 << 4),
    G_TLS_CERTIFICATE_INSECURE = (1 << 5),
    G_TLS_CERTIFICATE_GENERIC_ERROR = (1 << 6),
    G_TLS_CERTIFICATE_VALIDATE_ALL = 0x007f
} GTlsCertificateFlags;
typedef enum {
    G_TLS_AUTHENTICATION_NONE,
    G_TLS_AUTHENTICATION_REQUESTED,
    G_TLS_AUTHENTICATION_REQUIRED
} GTlsAuthenticationMode;
typedef enum {
    G_TLS_REHANDSHAKE_NEVER,
    G_TLS_REHANDSHAKE_SAFELY,
    G_TLS_REHANDSHAKE_UNSAFELY
} GTlsRehandshakeMode;
typedef enum GTlsPasswordFlags {
    G_TLS_PASSWORD_NONE,
    G_TLS_PASSWORD_RETRY = 1 << 1,
    G_TLS_PASSWORD_MANY_TRIES = 1 << 2,
    G_TLS_PASSWORD_FINAL_TRY = 1 << 3
}

```



```

} GTlsPasswordFlags;
typedef enum {
    G_TLS_INTERACTION_UNHANDLED,
    G_TLS_INTERACTION_HANDLED,
    G_TLS_INTERACTION_FAILED
} GTlsInteractionResult;
typedef enum {
    G_DBUS_INTERFACE_SKELETON_FLAGS_NONE,

G_DBUS_INTERFACE_SKELETON_FLAGS_HANDLE_METHOD_INVOCATIONS_IN_THRE
AD =
    (1 << 0)
} GDBusInterfaceSkeletonFlags;
typedef enum {
    G_DBUS_OBJECT_MANAGER_CLIENT_FLAGS_NONE,
    G_DBUS_OBJECT_MANAGER_CLIENT_FLAGS_DO_NOT_AUTO_START = (1 << 0)
} GDBusObjectManagerClientFlags;
typedef enum {
    G_TLS_DATABASE_VERIFY_NONE
} GTlsDatabaseVerifyFlags;
typedef enum {
    G_TLS_DATABASE_LOOKUP_NONE,
    G_TLS_DATABASE_LOOKUP_KEYPAIR
} GTlsDatabaseLookupFlags;
typedef enum {
    G_IO_MODULE_SCOPE_NONE,
    G_IO_MODULE_SCOPE_BLOCK_DUPLICATES
} GIOModuleScopeFlags;
typedef enum {
    G_SOCKET_CLIENT_RESOLVING,
    G_SOCKET_CLIENT_RESOLVED,
    G_SOCKET_CLIENT_CONNECTING,
    G_SOCKET_CLIENT_CONNECTED,
    G_SOCKET_CLIENT_PROXY_NEGOTIATING,
    G_SOCKET_CLIENT_PROXY_NEGOTIATED,
    G_SOCKET_CLIENT_TLS_HANDSHAKING,
    G_SOCKET_CLIENT_TLS_HANDSHAKED,
    G_SOCKET_CLIENT_COMPLETE
} GSocketClientEvent;

```

17.12.76 glib-2.0/gio/gioenumtypes.h

```

#define G_TYPE_APPLICATION_FLAGS (g_application_flags_get_type ())
#define G_TYPE_APP_INFO_CREATE_FLAGS (g_app_info_create_flags_get_type ())
#define G_TYPE_ASK_PASSWORD_FLAGS (g_ask_password_flags_get_type ())
#define G_TYPE_BUS_NAME_OWNER_FLAGS (g_bus_name_owner_flags_get_type ())
#define G_TYPE_BUS_NAME_WATCHER_FLAGS (g_bus_name_watcher_flags_get_type ())
#define G_TYPE_BUS_TYPE (g_bus_type_get_type ())
#define G_TYPE_CONVERTER_FLAGS (g_converter_flags_get_type ())
#define G_TYPE_CONVERTER_RESULT (g_converter_result_get_type ())
#define G_TYPE_CREDENTIALS_TYPE (g_credentials_type_get_type ())
#define G_TYPE_DATA_STREAM_BYTE_ORDER (g_data_stream_byte_order_get_type ())
#define G_TYPE_DATA_STREAM_NEWLINE_TYPE (g_data_stream_newline_type_get_type ())
#define G_TYPE_DBUS_CALL_FLAGS (g_dbus_call_flags_get_type ())
#define G_TYPE_DBUS_CAPABILITY_FLAGS (g_dbus_capability_flags_get_type ())
#define G_TYPE_DBUS_CONNECTION_FLAGS (g_dbus_connection_flags_get_type ())

```

```

#define G_TYPE_DBUS_ERROR (g_dbus_error_get_type ())
#define G_TYPE_DBUS_INTERFACE_SKELETON_FLAGS (g_dbus_interface_skeleton_flags_get_type ())
#define G_TYPE_DBUS_MESSAGE_BYTE_ORDER (g_dbus_message_byte_order_get_type ())
#define G_TYPE_DBUS_MESSAGE_FLAGS (g_dbus_message_flags_get_type ())
#define G_TYPE_DBUS_MESSAGE_HEADER_FIELD (g_dbus_message_header_field_get_type ())
#define G_TYPE_DBUS_MESSAGE_TYPE (g_dbus_message_type_get_type ())
#define G_TYPE_DBUS_OBJECT_MANAGER_CLIENT_FLAGS (g_dbus_object_manager_client_flags_get_type ())
#define G_TYPE_DBUS_PROPERTY_INFO_FLAGS (g_dbus_property_info_flags_get_type ())
#define G_TYPE_DBUS_PROXY_FLAGS (g_dbus_proxy_flags_get_type ())
#define G_TYPE_DBUS_SEND_MESSAGE_FLAGS (g_dbus_send_message_flags_get_type ())
#define G_TYPE_DBUS_SERVER_FLAGS (g_dbus_server_flags_get_type ())
#define G_TYPE_DBUS_SIGNAL_FLAGS (g_dbus_signal_flags_get_type ())
#define G_TYPE_DBUS_SUBTREE_FLAGS (g_dbus_subtree_flags_get_type ())
#define G_TYPE_DRIVE_START_FLAGS (g_drive_start_flags_get_type ())
#define G_TYPE_DRIVE_START_STOP_TYPE (g_drive_start_stop_type_get_type ())
#define G_TYPE_EMBLEM_ORIGIN (g_emblem_origin_get_type ())
#define G_TYPE_FILESYSTEM_PREVIEW_TYPE (g_filesystem_preview_type_get_type ())
#define G_TYPE_FILE_ATTRIBUTE_INFO_FLAGS (g_file_attribute_info_flags_get_type ())
#define G_TYPE_FILE_ATTRIBUTE_STATUS (g_file_attribute_status_get_type ())
#define G_TYPE_FILE_ATTRIBUTE_TYPE (g_file_attribute_type_get_type ())
#define G_TYPE_FILE_COPY_FLAGS (g_file_copy_flags_get_type ())
#define G_TYPE_FILE_CREATE_FLAGS (g_file_create_flags_get_type ())
#define G_TYPE_FILE_MONITOR_EVENT (g_file_monitor_event_get_type ())
#define G_TYPE_FILE_MONITOR_FLAGS (g_file_monitor_flags_get_type ())
#define G_TYPE_FILE_QUERY_INFO_FLAGS (g_file_query_info_flags_get_type ())
#define G_TYPE_FILE_TYPE (g_file_type_get_type ())
#define G_TYPE_IO_ERROR_ENUM (g_io_error_enum_get_type ())
#define G_TYPE_IO_MODULE_SCOPE_FLAGS (g_io_module_scope_flags_get_type ())
#define G_TYPE_IO_STREAM_SPLICE_FLAGS (g_io_stream_splice_flags_get_type ())
#define G_TYPE_MOUNT_MOUNT_FLAGS (g_mount_mount_flags_get_type ())
#define G_TYPE_MOUNT_OPERATION_RESULT (g_mount_operation_result_get_type ())
#define G_TYPE_MOUNT_UNMOUNT_FLAGS (g_mount_unmount_flags_get_type ())
#define G_TYPE_OUTPUT_STREAM_SPLICE_FLAGS (g_output_stream_splice_flags_get_type ())
#define G_TYPE_PASSWORD_SAVE (g_password_save_get_type ())
#define G_TYPE_RESOLVER_ERROR (g_resolver_error_get_type ())
#define G_TYPE_RESOURCE_ERROR (g_resource_error_get_type ())
#define G_TYPE_RESOURCE_FLAGS (g_resource_flags_get_type ())
#define G_TYPE_RESOURCE_LOOKUP_FLAGS (g_resource_lookup_flags_get_type ())

```

```

#define G_TYPE_SETTINGS_BIND_FLAGS
(g_settings_bind_flags_get_type ())
#define G_TYPE_SOCKET_CLIENT_EVENT
(g_socket_client_event_get_type ())
#define G_TYPE_SOCKET_FAMILY (g_socket_family_get_type ())
#define G_TYPE_SOCKET_MSG_FLAGS (g_socket_msg_flags_get_type ())
#define G_TYPE_SOCKET_PROTOCOL (g_socket_protocol_get_type ())
#define G_TYPE_SOCKET_TYPE (g_socket_type_get_type ())
#define G_TYPE_TLS_AUTHENTICATION_MODE
(g_tls_authentication_mode_get_type ())
#define G_TYPE_TLS_CERTIFICATE_FLAGS
(g_tls_certificate_flags_get_type ())
#define G_TYPE_TLS_DATABASE_LOOKUP_FLAGS
(g_tls_database_lookup_flags_get_type ())
#define G_TYPE_TLS_DATABASE_VERIFY_FLAGS
(g_tls_database_verify_flags_get_type ())
#define G_TYPE_TLS_ERROR (g_tls_error_get_type ())
#define G_TYPE_TLS_INTERACTION_RESULT
(g_tls_interaction_result_get_type ())
#define G_TYPE_TLS_PASSWORD_FLAGS
(g_tls_password_flags_get_type ())
#define G_TYPE_TLS_REHANDSHAKE_MODE
(g_tls_rehandshake_mode_get_type ())
#define G_TYPE_UNIX_SOCKET_ADDRESS_TYPE
(g_unix_socket_address_type_get_type ())
#define G_TYPE_ZLIB_COMPRESSOR_FORMAT
(g_zlib_compressor_format_get_type ())

extern GType g_app_info_create_flags_get_type(void);
extern GType g_application_flags_get_type(void);
extern GType g_ask_password_flags_get_type(void);
extern GType g_bus_name_owner_flags_get_type(void);
extern GType g_bus_name_watcher_flags_get_type(void);
extern GType g_bus_type_get_type(void);
extern GType g_converter_flags_get_type(void);
extern GType g_converter_result_get_type(void);
extern GType g_credentials_type_get_type(void);
extern GType g_data_stream_byte_order_get_type(void);
extern GType g_data_stream_newline_type_get_type(void);
extern GType g_dbus_call_flags_get_type(void);
extern GType g_dbus_capability_flags_get_type(void);
extern GType g_dbus_connection_flags_get_type(void);
extern GType g_dbus_error_get_type(void);
extern GType g_dbus_interface_skeleton_flags_get_type(void);
extern GType g_dbus_message_byte_order_get_type(void);
extern GType g_dbus_message_flags_get_type(void);
extern GType g_dbus_message_header_field_get_type(void);
extern GType g_dbus_message_type_get_type(void);
extern GType g_dbus_object_manager_client_flags_get_type(void);
extern GType g_dbus_property_info_flags_get_type(void);
extern GType g_dbus_proxy_flags_get_type(void);
extern GType g_dbus_send_message_flags_get_type(void);
extern GType g_dbus_server_flags_get_type(void);
extern GType g_dbus_signal_flags_get_type(void);
extern GType g_dbus_subtree_flags_get_type(void);
extern GType g_drive_start_flags_get_type(void);
extern GType g_drive_start_stop_type_get_type(void);
extern GType g_emblem_origin_get_type(void);
extern GType g_file_attribute_info_flags_get_type(void);
extern GType g_file_attribute_status_get_type(void);
extern GType g_file_attribute_type_get_type(void);
extern GType g_file_copy_flags_get_type(void);
extern GType g_file_create_flags_get_type(void);
extern GType g_file_monitor_event_get_type(void);
extern GType g_file_monitor_flags_get_type(void);
extern GType g_file_query_info_flags_get_type(void);

```

```

extern GType g_file_type_get_type(void);
extern GType g_filesystem_preview_type_get_type(void);
extern GType g_io_error_enum_get_type(void);
extern GType g_io_module_scope_flags_get_type(void);
extern GType g_io_stream_splice_flags_get_type(void);
extern GType g_mount_mount_flags_get_type(void);
extern GType g_mount_operation_result_get_type(void);
extern GType g_mount_unmount_flags_get_type(void);
extern GType g_output_stream_splice_flags_get_type(void);
extern GType g_password_save_get_type(void);
extern GType g_resolver_error_get_type(void);
extern GType g_resource_error_get_type(void);
extern GType g_resource_flags_get_type(void);
extern GType g_resource_lookup_flags_get_type(void);
extern GType g_settings_bind_flags_get_type(void);
extern GType g_socket_client_event_get_type(void);
extern GType g_socket_family_get_type(void);
extern GType g_socket_msg_flags_get_type(void);
extern GType g_socket_protocol_get_type(void);
extern GType g_socket_type_get_type(void);
extern GType g_tls_authentication_mode_get_type(void);
extern GType g_tls_certificate_flags_get_type(void);
extern GType g_tls_database_lookup_flags_get_type(void);
extern GType g_tls_database_verify_flags_get_type(void);
extern GType g_tls_error_get_type(void);
extern GType g_tls_interaction_result_get_type(void);
extern GType g_tls_password_flags_get_type(void);
extern GType g_tls_rehandshake_mode_get_type(void);
extern GType g_unix_socket_address_type_get_type(void);
extern GType g_zlib_compressor_format_get_type(void);

```

17.12.77 glib-2.0/gio/gioerror.h

```

#define G_IO_ERROR      g_io_error_quark()

extern GIOErrorEnum g_io_error_from_errno(gint err_no);
extern GQuark g_io_error_quark(void);

```

17.12.78 glib-2.0/gio/giomodule.h

```

#define G_IO_TYPE_MODULE      (g_io_module_get_type ())
#define G_IO_MODULE_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_IO_TYPE_MODULE, GIOModuleClass))
#define G_IO_IS_MODULE_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_IO_TYPE_MODULE))
#define G_IO_MODULE(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_IO_TYPE_MODULE, GIOModule))
#define G_IO_IS_MODULE(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_IO_TYPE_MODULE))
#define G_IO_MODULE_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_IO_TYPE_MODULE, GIOModuleClass))

typedef struct _GIOModuleScope GIOModuleScope;
extern const char *g_io_extension_get_name(GIOExtension *
extension);
extern gint g_io_extension_get_priority(GIOExtension * extension);
extern GType g_io_extension_get_type(GIOExtension * extension);
extern GIOExtension
*g_io_extension_point_get_extension_by_name(GIOExtensionPoint
*
extension_point,
const char *name);

```

```

extern
*g_io_extension_point_get_extensions(GIOExtensionPoint *      GList
extension_point);

extern
g_io_extension_point_get_required_type(GIOExtensionPoint *      GType
extension_point);

extern GIOExtension *g_io_extension_point_implement(const char
*extension_point_name,
GType type,
const char
*extension_name,
gint priority);

extern GIOExtensionPoint *g_io_extension_point_lookup(const char
*name);

extern GIOExtensionPoint *g_io_extension_point_register(const char
*name);

extern
g_io_extension_point_set_required_type(GIOExtensionPoint *      void
extension_point,
GType type);

extern GTypeClass *g_io_extension_ref_class(GIOExtension *
extension);

extern GType g_io_module_get_type(void);

extern GIOModule *g_io_module_new(const gchar * filename);

extern void g_io_module_scope_block(GIOModuleScope * scope,
const gchar * basename);

extern void g_io_module_scope_free(GIOModuleScope * scope);

extern GIOModuleScope *g_io_module_scope_new(GIOModuleScopeFlags
flags);

extern GList *g_io_modules_load_all_in_directory(const gchar *
dirname);

extern GList *g_io_modules_load_all_in_directory_with_scope(const
gchar *
dirname,
GIOModuleScope
* scope);

extern void g_io_modules_scan_all_in_directory(const char
*dirname);

extern void g_io_modules_scan_all_in_directory_with_scope(const
gchar *
dirname,
GIOModuleScope *
scope);

```

17.12.79 glib-2.0/gio/gioscheduler.h

```

extern void g_io_scheduler_cancel_all_jobs(void);

extern
g_io_scheduler_job_send_to_mainloop(GIOSchedulerJob * job,      gboolean
GSourceFunc func,
gpointer user_data,
GDestroyNotify notify);

extern
g_io_scheduler_job_send_to_mainloop_async(GIOSchedulerJob *      void
job,
GSourceFunc func,
gpointer user_data,
GDestroyNotify
notify);

extern void g_io_scheduler_push_job(GIOSchedulerJobFunc job_func,
gpointer user_data,
GDestroyNotify notify,
gint io_priority,
GCancellable * cancellable);

```

17.12.80 glib-2.0/gio/giostream.h

```

#define G_TYPE_IO_STREAM (g_io_stream_get_type ())
#define G_IO_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_IO_STREAM, GIOStreamClass))
#define G_IS_IO_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_TYPE_IO_STREAM))
#define G_IO_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_IO_STREAM, GIOStream))
#define G_IS_IO_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_IO_STREAM))
#define G_IO_STREAM_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_IO_STREAM, GIOStreamClass))

typedef struct _GIOStreamPrivate GIOStreamPrivate;
typedef struct _GIOStreamClass {
    GObjectClass parent_class;
    GInputStream *(*get_input_stream) (GIOStream * stream);
    GOutputStream *(*get_output_stream) (GIOStream * stream);
    gboolean(*close_fn) (GIOStream * stream, GCancelable *
cancellable,
                        GError * *error);
    void (*close_async) (GIOStream * stream, int io_priority,
                        GCancelable * cancellable,
                        GAsyncReadyCallback callback, gpointer
user_data);
    gboolean(*close_finish) (GIOStream * stream, GAsyncResult *
result,
                        GError * *error);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
    void (*_g_reserved7) (void);
    void (*_g_reserved8) (void);
    void (*_g_reserved9) (void);
    void (*_g_reserved10) (void);
} GIOStreamClass;
struct _GIOStream {
    GObject parent_instance;
    GIOStreamPrivate *priv;
};
extern void g_io_stream_clear_pending(GIOStream * stream);
extern gboolean g_io_stream_close(GIOStream * stream,
                                GCancelable * cancellable,
                                GError * *error);
extern void g_io_stream_close_async(GIOStream * stream, int
io_priority,
                                GCancelable * cancellable,
                                GAsyncReadyCallback callback,
                                gpointer user_data);
extern gboolean g_io_stream_close_finish(GIOStream * stream,
                                        GAsyncResult * result,
                                        GError * *error);
extern GInputStream *g_io_stream_get_input_stream(GIOStream *
stream);
extern GOutputStream *g_io_stream_get_output_stream(GIOStream *
stream);
extern GType g_io_stream_get_type(void);
extern gboolean g_io_stream_has_pending(GIOStream * stream);
extern gboolean g_io_stream_is_closed(GIOStream * stream);
extern gboolean g_io_stream_set_pending(GIOStream * stream,
                                        GError * *error);

```

```

extern void g_io_stream_splice_async(GIOStream * stream1,
                                     GIOStream * stream2,
                                     GIOStreamSpliceFlags flags,
                                     int io_priority,
                                     Gancellable * cancellable,
                                     GAsyncReadyCallback callback,
                                     gpointer user_data);
extern gboolean g_io_stream_splice_finish(GAsyncResult * result,
                                           GError * *error);

```

17.12.81 glib-2.0/gio/giotypes.h

```

typedef struct _GAppLaunchContext GAppLaunchContext;
typedef struct _GAppInfo GAppInfo;
typedef struct _GAsyncResult GAsyncResult;
typedef struct _GAsyncInitable GAsyncInitable;
typedef struct _GBufferedInputStream GBufferedInputStream;
typedef struct _GBufferedOutputStream GBufferedOutputStream;
typedef struct _GCancelable GCancelable;
typedef struct _GCharsetConverter GCharsetConverter;
typedef struct _GConverter GConverter;
typedef struct _GConverterInputStream GConverterInputStream;
typedef struct _GConverterOutputStream GConverterOutputStream;
typedef struct _GDataInputStream GDataInputStream;
typedef struct _GSimplePermission GSimplePermission;
typedef struct _GZlibCompressor GZlibCompressor;
typedef struct _GZlibDecompressor GZlibDecompressor;
typedef struct _GSimpleActionGroup GSimpleActionGroup;
typedef struct _GRemoteActionGroup GRemoteActionGroup;
typedef struct _GDBusActionGroup GDBusActionGroup;
typedef struct _GActionMap GActionMap;
typedef struct _GActionGroup GActionGroup;
typedef struct _GSimpleAction GSimpleAction;
typedef struct _GAction GAction;
typedef struct _GApplication GApplication;
typedef struct _GApplicationCommandLine GApplicationCommandLine;
typedef struct _GSettingsBackend GSettingsBackend;
typedef struct _GSettings GSettings;
typedef struct _GPermission GPermission;
typedef struct _GMenuModel GMenuModel;
typedef struct _GDrive GDrive;
typedef struct _GFileEnumerator GFileEnumerator;
typedef struct _GFileMonitor GFileMonitor;
typedef struct _GFilterInputStream GFilterInputStream;
typedef struct _GFilterOutputStream GFilterOutputStream;
typedef struct _GFile GFile;
typedef struct _GFileInfo GFileInfo;
typedef struct _GFileAttributeMatcher GFileAttributeMatcher;
typedef struct _GFileAttributeInfo GFileAttributeInfo;
typedef struct _GFileAttributeInfoList GFileAttributeInfoList;
typedef struct _GFileDescriptorBased GFileDescriptorBased;
typedef struct _GFileInputStream GFileInputStream;
typedef struct _GFileOutputStream GFileOutputStream;
typedef struct _GFileIOStream GFileIOStream;
typedef struct _GFileIcon GFileIcon;
typedef struct _GFilenameCompleter GFilenameCompleter;
typedef struct _GIcon GIcon;
typedef struct _GInetAddress GInetAddress;
typedef struct _GInetAddressMask GInetAddressMask;
typedef struct _GInetSocketAddress GInetSocketAddress;
typedef struct _GInputStream GInputStream;
typedef struct _GInitable GInitable;
typedef struct _GIOModule GIOModule;
typedef struct _GIOExtensionPoint GIOExtensionPoint;
typedef struct _GIOExtension GIOExtension;

```



```

typedef struct _GIOSchedulerJob GIOSchedulerJob;
typedef struct _GIOStreamAdapter GIOStreamAdapter;
typedef struct _GLoadableIcon GLoadableIcon;
typedef struct _GMemoryInputStream GMemoryInputStream;
typedef struct _GMemoryOutputStream GMemoryOutputStream;
typedef struct _GMount GMount;
typedef struct _GMountOperation GMountOperation;
typedef struct _GNetworkAddress GNetworkAddress;
typedef struct _GNetworkMonitor GNetworkMonitor;
typedef struct _GNetworkService GNetworkService;
typedef struct _GOutputStream GOutputStream;
typedef struct _GIOStream GIOStream;
typedef struct _GPollableInputStream GPollableInputStream;
typedef struct _GPollableOutputStream GPollableOutputStream;
typedef struct _GResolver GResolver;
typedef struct _GResource GResource;
typedef struct _GSeekable GSeekable;
typedef struct _GSimpleAsyncResult GSimpleAsyncResult;
typedef struct _GSocket GSocket;
typedef struct _GSocketControlMessage GSocketControlMessage;
typedef struct _GSocketClient GSocketClient;
typedef struct _GSocketConnection GSocketConnection;
typedef struct _GSocketListener GSocketListener;
typedef struct _GSocketService GSocketService;
typedef struct _GSocketAddress GSocketAddress;
typedef struct _GSocketAddressEnumerator GSocketAddressEnumerator;
typedef struct _GSocketConnectable GSocketConnectable;
typedef struct _GSrvTarget GSrvTarget;
typedef struct _GTcpConnection GTcpConnection;
typedef struct _GTcpWrapperConnection GTcpWrapperConnection;
typedef struct _GThreadedSocketService GThreadedSocketService;
typedef struct _GThemedIcon GThemedIcon;
typedef struct _GTlsCertificate GTlsCertificate;
typedef struct _GTlsClientConnection GTlsClientConnection;
typedef struct _GTlsConnection GTlsConnection;
typedef struct _GTlsDatabase GTlsDatabase;
typedef struct _GTlsFileDatabase GTlsFileDatabase;
typedef struct _GTlsInteraction GTlsInteraction;
typedef struct _GTlsPassword GTlsPassword;
typedef struct _GTlsServerConnection GTlsServerConnection;
typedef struct _GVfs GVfs;
typedef struct _GProxyResolver GProxyResolver;
typedef struct _GProxy GProxy;
typedef struct _GProxyAddress GProxyAddress;
typedef struct _GProxyAddressEnumerator GProxyAddressEnumerator;
typedef struct _GVolume GVolume;
typedef struct _GVolumeMonitor GVolumeMonitor;
typedef void (*GAsyncReadyCallback) (GObject * source_object,
                                     GAsyncResult * res,
                                     gpointer user_data);
typedef void (*GFileProgressCallback) (goffset current_num_bytes,
                                       goffset total_num_bytes,
                                       gpointer user_data);
typedef gboolean (*GFileReadMoreCallback) (const char
*file_contents,
                                           goffset file_size,
                                           gpointer callback_data);
typedef gboolean (*GIOSchedulerJobFunc) (GIOSchedulerJob * job,
                                         Gancellable * cancellable,
                                         gpointer user_data);
typedef void (*GSimpleAsyncThreadFunc) (GSimpleAsyncResult * res,
                                         GObject * object,
                                         Gancellable * cancellable);

typedef struct _GInputVector {
    gpointer buffer;
    gsize size;
}

```



```

} GInputVector;
typedef struct _GOutputVector {
    gconstpointer buffer;
    gsize size;
} GOutputVector;
typedef struct _GCredentials GCredentials;
typedef struct _GUnixCredentialsMessage GUnixCredentialsMessage;
typedef struct _GUnixFDList GUnixFDList;
typedef struct _GDBusMessage GDBusMessage;
typedef struct _GDBusConnection GDBusConnection;
typedef struct _GDBusProxy GDBusProxy;
typedef struct _GDBusMethodInvocation GDBusMethodInvocation;
typedef struct _GDBusServer GDBusServer;
typedef struct _GDBusAuthObserver GDBusAuthObserver;
typedef struct _GDBusErrorEntry GDBusErrorEntry;
typedef struct _GDBusInterfaceVTable GDBusInterfaceVTable;
typedef struct _GDBusSubtreeVTable GDBusSubtreeVTable;
typedef struct _GDBusAnnotationInfo GDBusAnnotationInfo;
typedef struct _GDBusArgInfo GDBusArgInfo;
typedef struct _GDBusMethodInfo GDBusMethodInfo;
typedef struct _GDBusSignalInfo GDBusSignalInfo;
typedef struct _GDBusPropertyInfo GDBusPropertyInfo;
typedef struct _GDBusInterfaceInfo GDBusInterfaceInfo;
typedef struct _GDBusNodeInfo GDBusNodeInfo;
typedef struct _GDBusInterface GDBusInterface;
typedef struct _GDBusInterfaceSkeleton GDBusInterfaceSkeleton;
typedef struct _GDBusObject GDBusObject;
typedef struct _GDBusObjectSkeleton GDBusObjectSkeleton;
typedef struct _GDBusObjectProxy GDBusObjectProxy;
typedef struct _GDBusObjectManager GDBusObjectManager;
typedef struct _GDBusObjectManagerClient GDBusObjectManagerClient;
typedef struct _GDBusObjectManagerServer GDBusObjectManagerServer;
typedef GType (*GDBusProxyTypeFunc) (GDBusObjectManagerClient *
manager,
    const gchar * object_path,
    const gchar * interface_name,
    gpointer user_data);

```

17.12.82 glib-2.0/gio/gloadableicon.h

```

#define G_TYPE_LOADABLE_ICON (g_loadable_icon_get_type ())
#define G_LOADABLE_ICON(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_LOADABLE_ICON, GLoadableIcon))
#define G_IS_LOADABLE_ICON(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_LOADABLE_ICON))
#define G_LOADABLE_ICON_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_LOADABLE_ICON,
GLoadableIconIface))

typedef struct _GLoadableIconIface {
    GTypeInterface g_iface;
    GInputStream *(*load) (GLoadableIcon * icon, int size, char
**type,
        Gancellable * cancellable, GError * *error);
    void (*load_async) (GLoadableIcon * icon, int size,
        Gancellable * cancellable,
        GAsyncReadyCallback callback, gpointer
user_data);
    GInputStream *(*load_finish) (GLoadableIcon * icon,
GAsyncResult * res,
        char **type, GError * *error);
} GLoadableIconIface;
extern GType g_loadable_icon_get_type(void);
extern GInputStream *g_loadable_icon_load(GLoadableIcon * icon, int
size,

```

```

char **type,
GCancelable * cancellable,
GError * *error);

extern void g_loadable_icon_load_async(GLoadableIcon * icon, int
size,

GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);

extern GInputStream *g_loadable_icon_load_finish(GLoadableIcon *
icon,

GAsyncResult * res,
char **type,
GError * *error);

```

17.12.83 glib-2.0/gio/gmemoryinputstream.h

```

#define G_TYPE_MEMORY_INPUT_STREAM
(g_memory_input_stream_get_type ())
#define G_MEMORY_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_MEMORY_INPUT_STREAM,
GMemoryInputStreamClass))
#define G_IS_MEMORY_INPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_MEMORY_INPUT_STREAM))
#define G_MEMORY_INPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_MEMORY_INPUT_STREAM, GMemoryInputStream))
#define G_IS_MEMORY_INPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_MEMORY_INPUT_STREAM))
#define G_MEMORY_INPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_MEMORY_INPUT_STREAM,
GMemoryInputStreamClass))

typedef struct _GMemoryInputStreamClass {
GInputStreamClass parent_class;
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
void (*_g_reserved5) (void);
} GMemoryInputStreamClass;

typedef struct _GMemoryInputStreamPrivate
GMemoryInputStreamPrivate;
struct _GMemoryInputStream {
GInputStream parent_instance;
GMemoryInputStreamPrivate *priv;
};

extern void g_memory_input_stream_add_data(GMemoryInputStream *
stream,

const void *data, gssize len,
GDestroyNotify destroy);

extern GType g_memory_input_stream_get_type(void);
extern GInputStream *g_memory_input_stream_new(void);
extern GInputStream *g_memory_input_stream_new_from_data(const
void *data,

gssize len,
GDestroyNotify
destroy);

```

17.12.84 glib-2.0/gio/gmemoryoutputstream.h

```

#define G_TYPE_MEMORY_OUTPUT_STREAM
(g_memory_output_stream_get_type ())

```

```

#define G_MEMORY_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_MEMORY_OUTPUT_STREAM,
GMemoryOutputStreamClass))
#define G_IS_MEMORY_OUTPUT_STREAM_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_MEMORY_OUTPUT_STREAM))
#define G_MEMORY_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_MEMORY_OUTPUT_STREAM, GMemoryOutputStream))
#define G_IS_MEMORY_OUTPUT_STREAM(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_MEMORY_OUTPUT_STREAM))
#define G_MEMORY_OUTPUT_STREAM_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_MEMORY_OUTPUT_STREAM,
GMemoryOutputStreamClass))

typedef struct _GMemoryOutputStreamClass {
    GOutputStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
} GMemoryOutputStreamClass;
typedef struct _GMemoryOutputStreamPrivate
GMemoryOutputStreamPrivate;
struct _GMemoryOutputStream {
    GOutputStream parent_instance;
    GMemoryOutputStreamPrivate *priv;
};
typedef gpointer(*GReallocFunc) (gpointer data, gsize size);
extern gpointer
g_memory_output_stream_get_data(GMemoryOutputStream *
    ostream);
extern gsize
g_memory_output_stream_get_data_size(GMemoryOutputStream *
    ostream);
extern gsize g_memory_output_stream_get_size(GMemoryOutputStream *
    ostream);
extern GType g_memory_output_stream_get_type(void);
extern GOutputStream *g_memory_output_stream_new(gpointer data,
    gsize size,
    GReallocFunc
    realloc_function,
    GDestroyNotify
    destroy_function);
extern gpointer
g_memory_output_stream_steal_data(GMemoryOutputStream *
    ostream);

```

17.12.85 glib-2.0/gio/gmenu.h

```

#define G_TYPE_MENU (g_menu_get_type ())
#define G_TYPE_MENU_ITEM (g_menu_item_get_type ())
#define G_MENU(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_MENU, GMenu))
#define G_MENU_ITEM(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_MENU_ITEM, GMenuItem))
#define G_IS_MENU(inst) (G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_MENU))
#define G_IS_MENU_ITEM(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_MENU_ITEM))

typedef struct _GMenuItem GMenuItem;
typedef struct _GMenu GMenu;
extern void g_menu_append(GMenu * menu, const gchar * label,
    const gchar * detailed_action);
extern void g_menu_append_item(GMenu * menu, GMenuItem * item);

```

```

extern void g_menu_append_section(GMenu * menu, const gchar * label,
                                GMenuModel * section);
extern void g_menu_append_submenu(GMenu * menu, const gchar * label,
                                GMenuModel * submenu);
extern void g_menu_freeze(GMenu * menu);
extern GType g_menu_get_type(void);
extern void g_menu_insert(GMenu * menu, gint position, const gchar
* label,
                        const gchar * detailed_action);
extern void g_menu_insert_item(GMenu * menu, gint position,
                              GMenuItem * item);
extern void g_menu_insert_section(GMenu * menu, gint position,
                                const gchar * label,
                                GMenuModel * section);
extern void g_menu_insert_submenu(GMenu * menu, gint position,
                                const gchar * label,
                                GMenuModel * submenu);
extern GType g_menu_item_get_type(void);
extern GMenuItem *g_menu_item_new(const gchar * label,
                                const gchar * detailed_action);
extern GMenuItem *g_menu_item_new_section(const gchar * label,
                                          GMenuModel * section);
extern GMenuItem *g_menu_item_new_submenu(const gchar * label,
                                          GMenuModel * submenu);
extern void g_menu_item_set_action_and_target(GMenuItem *
menu_item,
                                          const gchar * action,
                                          const gchar * format_string,
                                          ...);
extern void g_menu_item_set_action_and_target_value(GMenuItem *
menu_item,
                                          const gchar * action,
                                          GVariant *
target_value);
extern void g_menu_item_set_attribute(GMenuItem * menu_item,
                                    const gchar * attribute,
                                    const gchar * format_string, ...);
extern void g_menu_item_set_attribute_value(GMenuItem * menu_item,
                                           const gchar * attribute,
                                           GVariant * value);
extern void g_menu_item_set_detailed_action(GMenuItem * menu_item,
                                           const gchar *
detailed_action);
extern void g_menu_item_set_label(GMenuItem * menu_item,
                                const gchar * label);
extern void g_menu_item_set_link(GMenuItem * menu_item, const gchar
* link,
                                GMenuModel * model);
extern void g_menu_item_set_section(GMenuItem * menu_item,
                                   GMenuModel * section);
extern void g_menu_item_set_submenu(GMenuItem * menu_item,
                                   GMenuModel * submenu);
extern GMenu *g_menu_new(void);
extern void g_menu_prepend(GMenu * menu, const gchar * label,
                          const gchar * detailed_action);
extern void g_menu_prepend_item(GMenu * menu, GMenuItem * item);
extern void g_menu_prepend_section(GMenu * menu, const gchar *
label,
                          GMenuModel * section);
extern void g_menu_prepend_submenu(GMenu * menu, const gchar *
label,
                          GMenuModel * submenu);
extern void g_menu_remove(GMenu * menu, gint position);

```

17.12.86 glib-2.0/gio/gmenuexporter.h

```
extern guint g_dbus_connection_export_menu_model(GDBusConnection *
                                                connection,
                                                const gchar * object_path,
                                                GMenuModel * menu,
                                                GError * *error);
extern void g_dbus_connection_unexport_menu_model(GDBusConnection
*
                                                connection,
                                                guint export_id);
```

17.12.87 glib-2.0/gio/gmenumodel.h

```
#define G_TYPE_MENU_ATTRIBUTE_ITER
(g_menu_attribute_iter_get_type ())
#define G_TYPE_MENU_LINK_ITER (g_menu_link_iter_get_type ())
#define G_TYPE_MENU_MODEL (g_menu_model_get_type ())
#define G_MENU_ATTRIBUTE_ITER_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_MENU_ATTRIBUTE_ITER,
GMenuAttributeIterClass))
#define G_MENU_LINK_ITER_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_MENU_LINK_ITER, GMenuLinkIterClass))
#define G_MENU_MODEL_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_MENU_MODEL, GMenuModelClass))
#define G_IS_MENU_ATTRIBUTE_ITER_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_MENU_ATTRIBUTE_ITER))
#define G_IS_MENU_LINK_ITER_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_MENU_LINK_ITER))
#define G_IS_MENU_MODEL_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_MENU_MODEL))
#define G_MENU_ATTRIBUTE_ITER(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_MENU_ATTRIBUTE_ITER,
GMenuAttributeIter))
#define G_MENU_LINK_ITER(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_MENU_LINK_ITER, GMenuLinkIter))
#define G_MENU_MODEL(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_MENU_MODEL, GMenuModel))
#define G_IS_MENU_ATTRIBUTE_ITER(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_MENU_ATTRIBUTE_ITER))
#define G_IS_MENU_LINK_ITER(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_MENU_LINK_ITER))
#define G_IS_MENU_MODEL(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_MENU_MODEL))
#define G_MENU_ATTRIBUTE_ITER_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_MENU_ATTRIBUTE_ITER,
GMenuAttributeIterClass))
#define G_MENU_LINK_ITER_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_MENU_LINK_ITER,
GMenuLinkIterClass))
#define G_MENU_MODEL_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_MENU_MODEL, GMenuModelClass))
#define G_MENU_ATTRIBUTE_ACTION "action"
#define G_MENU_ATTRIBUTE_LABEL "label"
#define G_MENU_LINK_SECTION "section"
#define G_MENU_LINK_SUBMENU "submenu"
#define G_MENU_ATTRIBUTE_TARGET "target"

typedef struct _GMenuModelPrivate GMenuModelPrivate;
typedef struct _GMenuModelClass {
    GObjectClass parent_class;
    gboolean(*is_mutable) (GMenuModel * model);
    gint(*get_n_items) (GMenuModel * model);
```

```

    void (*get_item_attributes) (GMenuModel * model, gint
item_index,
                                GHashTable * *attributes);
    GMenuAttributeIter *(*iterate_item_attributes) (GMenuModel *
model,
                                                    gint item_index);
    GVariant *(*get_item_attribute_value) (GMenuModel * model,
                                           gint item_index,
                                           const gchar * attribute,
                                           const GVariantType *
expected_type);
    void (*get_item_links) (GMenuModel * model, gint item_index,
                             GHashTable * *attributes);
    GMenuLinkIter *(*iterate_item_links) (GMenuModel * model,
                                           gint item_index);
    GMenuModel *(*get_item_link) (GMenuModel * model, gint
item_index,
                                const gchar * link);
} GMenuModelClass;
typedef struct _GMenuAttributeIterPrivate
GMenuAttributeIterPrivate;
typedef struct _GMenuAttributeIterClass {
    GObjectClass parent_class;
    gboolean(*get_next) (GMenuAttributeIter * iter,
                        const gchar * *out_type, GVariant * *value);
} GMenuAttributeIterClass;
typedef struct _GMenuAttributeIter {
    GObject parent_instance;
    GMenuAttributeIterPrivate *priv;
} GMenuAttributeIter;
typedef struct _GMenuLinkIterPrivate
GMenuLinkIterPrivate;
typedef struct _GMenuLinkIterClass {
    GObjectClass parent_class;
    gboolean(*get_next) (GMenuLinkIter * iter, const gchar *
*out_name,
                        GMenuModel * *value);
} GMenuLinkIterClass;
typedef struct _GMenuLinkIter {
    GObject parent_instance;
    GMenuLinkIterPrivate *priv;
} GMenuLinkIter;
struct _GMenuModel {
    GObject parent_instance;
    GMenuModelPrivate *priv;
};
extern const char
*g_menu_attribute_iter_get_name(GMenuAttributeIter *
iter);
extern gboolean g_menu_attribute_iter_get_next(GMenuAttributeIter
*iter,
                                              const gchar * *out_name,
                                              GVariant * *value);
extern GType g_menu_attribute_iter_get_type(void);
extern GVariant
*g_menu_attribute_iter_get_value(GMenuAttributeIter *
iter);
extern gboolean g_menu_attribute_iter_next(GMenuAttributeIter *
iter);
extern const char *g_menu_link_iter_get_name(GMenuLinkIter * iter);
extern gboolean g_menu_link_iter_get_next(GMenuLinkIter * iter,
                                           const gchar * *out_link,
                                           GMenuModel * *value);
extern GType g_menu_link_iter_get_type(void);
extern GMenuModel *g_menu_link_iter_get_value(GMenuLinkIter *
iter);
extern gboolean g_menu_link_iter_next(GMenuLinkIter * iter);

```

```

extern gboolean g_menu_model_get_item_attribute(GMenuModel * model,
                                                gint item_index,
                                                const gchar * attribute,
                                                const gchar *
                                                format_string, ...);
extern GVariant *g_menu_model_get_item_attribute_value(GMenuModel
* model,
                                                gint item_index,
                                                const gchar *
                                                attribute,
                                                const GVariantType *
                                                expected_type);
extern GMenuModel *g_menu_model_get_item_link(GMenuModel * model,
                                                gint item_index,
                                                const gchar * link);
extern gint g_menu_model_get_n_items(GMenuModel * model);
extern GType g_menu_model_get_type(void);
extern gboolean g_menu_model_is_mutable(GMenuModel * model);
extern void g_menu_model_items_changed(GMenuModel * model, gint
position,
                                      gint removed, gint added);
extern
                                      GMenuItemIter
*g_menu_model_iterate_item_attributes(GMenuModel
* model,
                                      gint
                                      item_index);
extern GMenuItemIter *g_menu_model_iterate_item_links(GMenuModel *
model,
                                      gint item_index);

```

17.12.88 glib-2.0/gio/gmount.h

```

#define G_TYPE_MOUNT (g_mount_get_type ())
#define G_MOUNT(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_MOUNT, GMount))
#define G_IS_MOUNT(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_MOUNT))
#define G_MOUNT_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE
((obj), G_TYPE_MOUNT, GMountIface))

extern gboolean g_mount_can_eject(GMount * mount);
extern gboolean g_mount_can_unmount(GMount * mount);
extern void g_mount_eject(GMount * mount, GMountUnmountFlags flags,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
extern gboolean g_mount_eject_finish(GMount * mount, GAsyncResult
* result,
GError * *error);
extern void g_mount_eject_with_operation(GMount * mount,
GMountUnmountFlags flags,
GMountOperation *
mount_operation,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
extern gboolean g_mount_eject_with_operation_finish(GMount * mount,
GAsyncResult * result,
GError * *error);
extern GFile *g_mount_get_default_location(GMount * mount);
extern GDrive *g_mount_get_drive(GMount * mount);
extern GIcon *g_mount_get_icon(GMount * mount);
extern char *g_mount_get_name(GMount * mount);
extern GFile *g_mount_get_root(GMount * mount);
extern const char *g_mount_get_sort_key(GMount * mount);

```

```

extern GType g_mount_get_type(void);
extern char *g_mount_get_uuid(GMount * mount);
extern GVolume *g_mount_get_volume(GMount * mount);
extern void g_mount_guess_content_type(GMount * mount,
                                       gboolean force_rescan,
                                       Gancellable * cancellable,
                                       GAsyncReadyCallback callback,
                                       gpointer user_data);
extern gchar **g_mount_guess_content_type_finish(GMount * mount,
                                                  GAsyncResult * result,
                                                  GError * *error);
extern gchar **g_mount_guess_content_type_sync(GMount * mount,
                                                gboolean force_rescan,
                                                Gancellable * cancellable,
                                                GError * *error);
extern gboolean g_mount_is_shadowed(GMount * mount);
extern void g_mount_remount(GMount * mount, GMountMountFlags flags,
                            GMountOperation * mount_operation,
                            Gancellable * cancellable,
                            GAsyncReadyCallback callback,
                            gpointer user_data);
extern gboolean g_mount_remount_finish(GMount * mount,
                                       GAsyncResult * result,
                                       GError * *error);
extern void g_mount_shadow(GMount * mount);
extern void g_mount_unmount(GMount * mount, GMountUnmountFlags
                             flags,
                             Gancellable * cancellable,
                             GAsyncReadyCallback callback,
                             gpointer user_data);
extern gboolean g_mount_unmount_finish(GMount * mount,
                                       GAsyncResult * result,
                                       GError * *error);
extern void g_mount_unmount_with_operation(GMount * mount,
                                           GMountUnmountFlags flags,
                                           GMountOperation *
                                           mount_operation,
                                           Gancellable * cancellable,
                                           GAsyncReadyCallback callback,
                                           gpointer user_data);
extern gboolean g_mount_unmount_with_operation_finish(GMount *
mount,
                                                    GAsyncResult *
                                                    result,
                                                    GError * *error);
extern void g_mount_unshadow(GMount * mount);

```

17.12.89 glib-2.0/gio/gmountoperation.h

```

#define G_TYPE_MOUNT_OPERATION (g_mount_operation_get_type ())
#define G_MOUNT_OPERATION_CLASS(k) (G_TYPE_MOUNT_OPERATION,
(G_TYPE_CHECK_CLASS_CAST((k),
GMountOperationClass))
#define G_IS_MOUNT_OPERATION_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_MOUNT_OPERATION))
#define G_MOUNT_OPERATION(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_MOUNT_OPERATION, GMountOperation))
#define G_IS_MOUNT_OPERATION(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_MOUNT_OPERATION))
#define G_MOUNT_OPERATION_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_MOUNT_OPERATION, GMountOperationClass))

typedef struct _GMountOperationClass {
    GObjectClass parent_class;
    void (*ask_password) (void);

```



```

void (*ask_question) (void);
void (*reply) (void);
void (*aborted) (void);
void (*show_processes) (void);
void (*show_unmount_progress) (void);
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
void (*_g_reserved5) (void);
void (*_g_reserved6) (void);
void (*_g_reserved7) (void);
void (*_g_reserved8) (void);
void (*_g_reserved9) (void);
} GMountOperationClass;
typedef struct _GMountOperationPrivate GMountOperationPrivate;
struct _GMountOperation {
    GObject parent_instance;
    GMountOperationPrivate *priv;
};
extern gboolean g_mount_operation_get_anonymous(GMountOperation *
op);
extern int g_mount_operation_get_choice(GMountOperation * op);
extern const char *g_mount_operation_get_domain(GMountOperation *
op);
extern const char *g_mount_operation_get_password(GMountOperation
* op);
extern
                                GPasswordSave
g_mount_operation_get_password_save(GMountOperation *
                                op);
extern GType g_mount_operation_get_type(void);
extern const char *g_mount_operation_get_username(GMountOperation
* op);
extern GMountOperation *g_mount_operation_new(void);
extern void g_mount_operation_reply(GMountOperation * op,
                                GMountOperationResult result);
extern void g_mount_operation_set_anonymous(GMountOperation * op,
                                gboolean anonymous);
extern void g_mount_operation_set_choice(GMountOperation * op, int
choice);
extern void g_mount_operation_set_domain(GMountOperation * op,
                                const char *domain);
extern void g_mount_operation_set_password(GMountOperation * op,
                                const char *password);
extern void g_mount_operation_set_password_save(GMountOperation *
op,
                                GPasswordSave save);
extern void g_mount_operation_set_username(GMountOperation * op,
                                const char *username);

```

17.12.90 glib-2.0/gio/gnativevolumemonitor.h

```

#define                                G_TYPE_NATIVE_VOLUME_MONITOR
(g_native_volume_monitor_get_type ())
#define                                G_NATIVE_VOLUME_MONITOR_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k),                                G_TYPE_NATIVE_VOLUME_MONITOR,
GNativeVolumeMonitorClass))
#define                                G_IS_NATIVE_VOLUME_MONITOR_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_NATIVE_VOLUME_MONITOR))
#define G_NATIVE_VOLUME_MONITOR(o) (G_TYPE_CHECK_INSTANCE_CAST
((o), G_TYPE_NATIVE_VOLUME_MONITOR, GNativeVolumeMonitor))
#define                                G_IS_NATIVE_VOLUME_MONITOR(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_NATIVE_VOLUME_MONITOR))
#define G_NATIVE_VOLUME_MONITOR_EXTENSION_POINT_NAME "gio-
native-volume-monitor"

```

```

typedef struct _GNativeVolumeMonitor {
    GVolumeMonitor parent_instance;
} GNativeVolumeMonitor;
typedef struct _GNativeVolumeMonitorClass {
    GVolumeMonitorClass parent_class;
    GMount *(*get_mount_for_mount_path) (const char *mount_path,
                                           Gancellable * cancellable);
} GNativeVolumeMonitorClass;
extern GType g_native_volume_monitor_get_type(void);

```

17.12.91 glib-2.0/gio/gnetworkaddress.h

```

#define G_TYPE_NETWORK_ADDRESS (g_network_address_get_type ())
#define G_NETWORK_ADDRESS_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_NETWORK_ADDRESS, GNetworkAddressClass))
#define G_IS_NETWORK_ADDRESS_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_NETWORK_ADDRESS))
#define G_NETWORK_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_NETWORK_ADDRESS, GNetworkAddress))
#define G_IS_NETWORK_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_NETWORK_ADDRESS))
#define G_NETWORK_ADDRESS_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_NETWORK_ADDRESS, GNetworkAddressClass))

typedef struct _GNetworkAddressClass {
    GObjectClass parent_class;
} GNetworkAddressClass;
typedef struct _GNetworkAddressPrivate GNetworkAddressPrivate;
struct _GNetworkAddress {
    GObject parent_instance;
    GNetworkAddressPrivate *priv;
};
extern const char *g_network_address_get_hostname(GNetworkAddress * addr);
extern guint16 g_network_address_get_port(GNetworkAddress * addr);
extern const char *g_network_address_get_scheme(GNetworkAddress * addr);
extern GType g_network_address_get_type(void);
extern GSocketConnectable *g_network_address_new(const gchar * hostname,
                                                  guint16 port);
extern GSocketConnectable *g_network_address_parse(const gchar * host_and_port,
                                                  guint16 default_port,
                                                  GError * *error);
extern GSocketConnectable *g_network_address_parse_uri(const gchar * uri,
                                                         guint16 default_port,
                                                         GError * *error);

```

17.12.92 glib-2.0/gio/gnetworkmonitor.h

```

#define G_TYPE_NETWORK_MONITOR (g_network_monitor_get_type ())
#define G_NETWORK_MONITOR(o) (G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_NETWORK_MONITOR, GNetworkMonitor))
#define G_IS_NETWORK_MONITOR(o) (G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_NETWORK_MONITOR))
#define G_NETWORK_MONITOR_GET_INTERFACE(o) (G_TYPE_INSTANCE_GET_INTERFACE((o), G_TYPE_NETWORK_MONITOR, GNetworkMonitorInterface))

```

```

#define G_NETWORK_MONITOR_EXTENSION_POINT_NAME "gio-network-
monitor"

typedef struct _GNetworkMonitorInterface {
    GTypeInterface g_iface;
    void (*network_changed) (GNetworkMonitor * monitor,
                             gboolean available);
    gboolean(*can_reach) (GNetworkMonitor * monitor,
                          GSocketConnectable * connectable,
                          Gancellable * cancellable, GError * *error);
    void (*can_reach_async) (GNetworkMonitor * monitor,
                             GSocketConnectable * connectable,
                             Gancellable * cancellable,
                             GAsyncReadyCallback callback,
                             gpointer user_data);
    gboolean(*can_reach_finish) (GNetworkMonitor * monitor,
                                 GAsyncResult * result, GError *
*error);
} GNetworkMonitorInterface;
extern gboolean g_network_monitor_can_reach(GNetworkMonitor *
monitor,
                                           GSocketConnectable *
connectable,
                                           Gancellable * cancellable,
                                           GError * *error);
extern void g_network_monitor_can_reach_async(GNetworkMonitor *
monitor,
                                              GSocketConnectable *
connectable,
                                              Gancellable * cancellable,
                                              GAsyncReadyCallback
callback,
                                              gpointer user_data);
extern gboolean g_network_monitor_can_reach_finish(GNetworkMonitor
*
monitor,
                                                  GAsyncResult * result,
                                                  GError * *error);
extern GNetworkMonitor *g_network_monitor_get_default(void);
extern gboolean g_network_monitor_get_network_available(GNetworkMonitor *
monitor);
extern GType g_network_monitor_get_type(void);

```

17.12.93 glib-2.0/gio/gnetworkservice.h

```

#define G_TYPE_NETWORK_SERVICE (g_network_service_get_type ())
#define G_NETWORK_SERVICE_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_NETWORK_SERVICE,
GNetworkServiceClass))
#define G_IS_NETWORK_SERVICE_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_NETWORK_SERVICE))
#define G_NETWORK_SERVICE(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_NETWORK_SERVICE, GNetworkService))
#define G_IS_NETWORK_SERVICE(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_NETWORK_SERVICE))
#define G_NETWORK_SERVICE_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_NETWORK_SERVICE, GNetworkServiceClass))

typedef struct _GNetworkServiceClass {
    GObjectClass parent_class;
} GNetworkServiceClass;
typedef struct _GNetworkServicePrivate GNetworkServicePrivate;
struct _GNetworkService {
    GObject parent_instance;

```

```

    GNetworkServicePrivate *priv;
};
extern const gchar *g_network_service_get_domain(GNetworkService *
srv);
extern const gchar *g_network_service_get_protocol(GNetworkService
* srv);
extern const gchar *g_network_service_get_scheme(GNetworkService *
srv);
extern const gchar *g_network_service_get_service(GNetworkService
* srv);
extern GType g_network_service_get_type(void);
extern GSocketConnectable *g_network_service_new(const gchar *
service,
                                                const gchar * protocol,
                                                const gchar * domain);
extern void g_network_service_set_scheme(GNetworkService * srv,
                                        const gchar * scheme);

```

17.12.94 glib-2.0/gio/goutputstream.h

```

#define G_TYPE_OUTPUT_STREAM      (g_output_stream_get_type ())
#define G_OUTPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
GOutputStreamClass), G_TYPE_OUTPUT_STREAM,
GOutputStreamClass))
#define G_IS_OUTPUT_STREAM_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_OUTPUT_STREAM))
#define G_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_OUTPUT_STREAM, GOutputStream))
#define G_IS_OUTPUT_STREAM(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_OUTPUT_STREAM))
#define G_OUTPUT_STREAM_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_OUTPUT_STREAM, GOutputStreamClass))

typedef struct _GOutputStreamClass {
    GObjectClass parent_class;
    gssize(*write_fn) (GOutputStream * stream, const void *buffer,
                       gsize count, GCancellable * cancellable,
                       GError * *error);
    gssize(*splice) (GOutputStream * stream, GInputStream * source,
                     GOutputStreamSpliceFlags flags,
                     GCancellable * cancellable, GError * *error);
    gboolean(*flush) (GOutputStream * stream, GCancellable *
cancellable,
                     GError * *error);
    gboolean(*close_fn) (GOutputStream * stream,
                         GCancellable * cancellable, GError * *error);
    void (*write_async) (GOutputStream * stream, const void *buffer,
                        gsize count, int io_priority,
                        GCancellable * cancellable,
                        GAsyncReadyCallback callback, gpointer
user_data);
    gssize(*write_finish) (GOutputStream * stream, GAsyncResult *
result,
                          GError * *error);
    void (*splice_async) (GOutputStream * stream, GInputStream *
source,
                          GOutputStreamSpliceFlags flags, int
io_priority,
                          GCancellable * cancellable,
                          GAsyncReadyCallback callback,
                          gpointer user_data);
    gssize(*splice_finish) (GOutputStream * stream, GAsyncResult *
result,
                          GError * *error);
    void (*flush_async) (GOutputStream * stream, int io_priority,

```

```

        GCancelable * cancellable,
        GAsyncReadyCallback callback,      gpointer
user_data);
    gboolean(*flush_finish) (GOutputStream * stream,
                             GAsyncResult * result, GError * *error);
    void (*close_async) (GOutputStream * stream, int io_priority,
                         GCancelable * cancellable,
                         GAsyncReadyCallback callback,      gpointer
user_data);
    gboolean(*close_finish) (GOutputStream * stream,
                             GAsyncResult * result, GError * *error);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
    void (*_g_reserved7) (void);
    void (*_g_reserved8) (void);
} GOutputStreamClass;
typedef struct _GOutputStreamPrivate GOutputStreamPrivate;
struct _GOutputStream {
    GObject parent_instance;
    GOutputStreamPrivate *priv;
};
extern void g_output_stream_clear_pending(GOutputStream * stream);
extern gboolean g_output_stream_close(GOutputStream * stream,
                                       GCancelable * cancellable,
                                       GError * *error);
extern void g_output_stream_close_async(GOutputStream * stream,
                                       int io_priority,
                                       GCancelable * cancellable,
                                       GAsyncReadyCallback callback,
                                       gpointer user_data);
extern gboolean g_output_stream_close_finish(GOutputStream *
stream,
                                             GAsyncResult * result,
                                             GError * *error);
extern gboolean g_output_stream_flush(GOutputStream * stream,
                                       GCancelable * cancellable,
                                       GError * *error);
extern void g_output_stream_flush_async(GOutputStream * stream,
                                       int io_priority,
                                       GCancelable * cancellable,
                                       GAsyncReadyCallback callback,
                                       gpointer user_data);
extern gboolean g_output_stream_flush_finish(GOutputStream *
stream,
                                             GAsyncResult * result,
                                             GError * *error);
extern GType g_output_stream_get_type(void);
extern gboolean g_output_stream_has_pending(GOutputStream *
stream);
extern gboolean g_output_stream_is_closed(GOutputStream * stream);
extern gboolean g_output_stream_is_closing(GOutputStream * stream);
extern gboolean g_output_stream_set_pending(GOutputStream * stream,
                                             GError * *error);
extern gssize g_output_stream_splice(GOutputStream * stream,
                                     GInputStream * source,
                                     GOutputStreamSpliceFlags flags,
                                     GCancelable * cancellable,
                                     GError * *error);
extern void g_output_stream_splice_async(GOutputStream * stream,
                                     GInputStream * source,
                                     GOutputStreamSpliceFlags flags,
                                     int io_priority,

```

```

        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gssize g_output_stream_splice_finish(GOutputStream * stream,
        GAsyncResult * result,
        GError * *error);
extern gssize g_output_stream_write(GOutputStream * stream,
        const void *buffer, gsize count,
        GCancelable * cancellable,
        GError * *error);
extern gboolean g_output_stream_write_all(GOutputStream * stream,
        const void *buffer, gsize count,
        gsize * bytes_written,
        GCancelable * cancellable,
        GError * *error);
extern void g_output_stream_write_async(GOutputStream * stream,
        const void *buffer, gsize count,
        int io_priority,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gssize g_output_stream_write_finish(GOutputStream * stream,
        GAsyncResult * result,
        GError * *error);

```

17.12.95 glib-2.0/gio/gpermission.h

```

#define G_TYPE_PERMISSION (g_permission_get_type ())
#define G_PERMISSION_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_PERMISSION, GPermissionClass))
#define G_IS_PERMISSION_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_PERMISSION))
#define G_PERMISSION(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_PERMISSION, GPermission))
#define G_IS_PERMISSION(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_PERMISSION))
#define G_PERMISSION_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_PERMISSION, GPermissionClass))

typedef struct _GPermissionPrivate GPermissionPrivate;
typedef struct _GPermissionClass {
    GObjectClass parent_class;
    gboolean(*acquire) (GPermission * permission,
        GCancelable * cancellable, GError * *error);
    void(*acquire_async) (GPermission * permission,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
    gboolean(*acquire_finish) (GPermission * permission,
        GAsyncResult * result, GError * *error);
    gboolean(*release) (GPermission * permission,
        GCancelable * cancellable, GError * *error);
    void(*release_async) (GPermission * permission,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
    gboolean(*release_finish) (GPermission * permission,
        GAsyncResult * result, GError * *error);
    gpointer reserved[16];
} GPermissionClass;
struct _GPermission {
    GObject parent_instance;
    GPermissionPrivate *priv;
};
extern gboolean g_permission_acquire(GPermission * permission,

```

```

        Gancellable * cancellable,
        GError * *error);
extern void g_permission_acquire_async(GPermission * permission,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_permission_acquire_finish(GPermission *
permission,
        GAsyncResult * result,
        GError * *error);
extern gboolean g_permission_get_allowed(GPermission * permission);
extern gboolean g_permission_get_can_acquire(GPermission *
permission);
extern gboolean g_permission_get_can_release(GPermission *
permission);
extern GType g_permission_get_type(void);
extern void g_permission_impl_update(GPermission * permission,
        gboolean allowed,
        gboolean can_acquire,
        gboolean can_release);
extern gboolean g_permission_release(GPermission * permission,
        Gancellable * cancellable,
        GError * *error);
extern void g_permission_release_async(GPermission * permission,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_permission_release_finish(GPermission *
permission,
        GAsyncResult * result,
        GError * *error);

```

17.12.96 glib-2.0/gio/gpollableinputstream.h

```

#define G_TYPE_POLLABLE_INPUT_STREAM
(g_pollable_input_stream_get_type ())
#define G_POLLABLE_INPUT_STREAM(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), G_TYPE_POLLABLE_INPUT_STREAM,
GPollableInputStream))
#define G_IS_POLLABLE_INPUT_STREAM(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), G_TYPE_POLLABLE_INPUT_STREAM))
#define G_POLLABLE_INPUT_STREAM_GET_INTERFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj),
G_TYPE_POLLABLE_INPUT_STREAM, GPollableInputStreamInterface))

typedef struct GPollableInputStreamInterface {
    GTypeInterface g_iface;
    gboolean(*can_poll) (GPollableInputStream * stream);
    gboolean(*is_readable) (GPollableInputStream * stream);
    GSource *(*create_source) (GPollableInputStream * stream,
        Gancellable * cancellable);
    gssize(*read_nonblocking) (GPollableInputStream * stream,
        void *buffer, gsize size, GError *
        *error);
} GPollableInputStreamInterface;
extern gboolean g_pollable_input_stream_can_poll(GPollableInputStream *
stream);
extern GSource *g_pollable_input_stream_create_source(GPollableInputStream
* stream,
        Gancellable *
        cancellable);
extern GType g_pollable_input_stream_get_type(void);

```

```

extern                                     gboolean
g_pollable_input_stream_is_readable(GPollableInputStream *
                                     stream);

extern                                     gssize
g_pollable_input_stream_read_nonblocking(GPollableInputStream
                                         * stream,
                                         void *buffer,
                                         gsize size,
                                         Gancellable *
                                         cancellable,
                                         GError * *error);

extern GSource *g_pollable_source_new(GObject * pollable_stream);

```

17.12.97 glib-2.0/gio/gpollableoutputstream.h

```

#define                                     G_TYPE_POLLABLE_OUTPUT_STREAM
(g_pollable_output_stream_get_type ())
#define                                     G_POLLABLE_OUTPUT_STREAM(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), G_TYPE_POLLABLE_OUTPUT_STREAM,
GPollableOutputStream))
#define                                     G_IS_POLLABLE_OUTPUT_STREAM(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), G_TYPE_POLLABLE_OUTPUT_STREAM))
#define                                     G_POLLABLE_OUTPUT_STREAM_GET_INTERFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj),
G_TYPE_POLLABLE_OUTPUT_STREAM, GPollableOutputStreamInterface))

typedef struct _GPollableOutputStreamInterface {
    GTypeInterface g_iface;
    gboolean(*can_poll) (GPollableOutputStream * stream);
    gboolean(*is_writable) (GPollableOutputStream * stream);
    GSource *(*create_source) (GPollableOutputStream * stream,
                              Gancellable * cancellable);
    gssize(*write_nonblocking) (GPollableOutputStream * stream,
                                const void *buffer, gsize size,
                                GError * *error);
} GPollableOutputStreamInterface;

extern                                     gboolean
g_pollable_output_stream_can_poll(GPollableOutputStream *
                                   stream);

extern GSource
    *g_pollable_output_stream_create_source(GPollableOutputStream
    * stream,
                                           Gancellable * cancellable);

extern GType g_pollable_output_stream_get_type(void);
extern                                     gboolean
g_pollable_output_stream_is_writable(GPollableOutputStream
                                     * stream);

extern gssize
g_pollable_output_stream_write_nonblocking(GPollableOutputStream *
stream,
                                           const void *buffer, gsize size,
                                           Gancellable * cancellable,
                                           GError * *error);

```

17.12.98 glib-2.0/gio/gproxy.h

```

#define G_TYPE_PROXY (g_proxy_get_type ())
#define G_PROXY(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_PROXY, GProxy))
#define G_IS_PROXY(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_PROXY))
#define G_PROXY_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE
((obj), G_TYPE_PROXY, GProxyInterface))

```



```

#define G_PROXY_EXTENSION_POINT_NAME    "gio-proxy"

typedef struct _GProxyInterface {
    GTypeInterface g_iface;
    GIOStream *(*connect) (GProxy * proxy, GIOStream * connection,
                           GProxyAddress * proxy_address,
                           Gancellable * cancellable, GError * *error);
    void (*connect_async) (GProxy * proxy, GIOStream * connection,
                           GProxyAddress * proxy_address,
                           Gancellable * cancellable,
                           GAsyncReadyCallback callback,
                           gpointer user_data);
    GIOStream *(*connect_finish) (GProxy * proxy, GAsyncResult *
    result,
                                   GError * *error);
    gboolean(*supports_hostname) (GProxy * proxy);
} GProxyInterface;
extern GIOStream *g_proxy_connect(GProxy * proxy, GIOStream *
    connection,
                                   GProxyAddress * proxy_address,
                                   Gancellable * cancellable,
                                   GError * *error);
extern void g_proxy_connect_async(GProxy * proxy, GIOStream *
    connection,
                                   GProxyAddress * proxy_address,
                                   Gancellable * cancellable,
                                   GAsyncReadyCallback callback,
                                   gpointer user_data);
extern GIOStream *g_proxy_connect_finish(GProxy * proxy,
    GAsyncResult * result,
    GError * *error);
extern GProxy *g_proxy_get_default_for_protocol(const gchar *
    protocol);
extern GType g_proxy_get_type(void);
extern gboolean g_proxy_supports_hostname(GProxy * proxy);

```

17.12.99 glib-2.0/gio/gproxyaddress.h

```

#define G_TYPE_PROXY_ADDRESS    (g_proxy_address_get_type ())
#define G_PROXY_ADDRESS_CLASS(k) G_TYPE_PROXY_ADDRESS,
(G_TYPE_CHECK_CLASS_CAST((k), G_PROXY_ADDRESS_CLASS,
    GProxyAddressClass))
#define G_IS_PROXY_ADDRESS_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
    ((k), G_PROXY_ADDRESS_CLASS))
#define G_PROXY_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
    G_PROXY_ADDRESS, GProxyAddress))
#define G_IS_PROXY_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
    G_PROXY_ADDRESS))
#define G_PROXY_ADDRESS_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
    ((o), G_PROXY_ADDRESS, GProxyAddressClass))

typedef struct _GProxyAddressClass {
    GInetSocketAddressClass parent_class;
} GProxyAddressClass;
typedef struct _GProxyAddressPrivate GProxyAddressPrivate;
struct _GProxyAddress {
    GInetSocketAddress parent_instance;
    GProxyAddressPrivate *priv;
};
extern const gchar
*g_proxy_address_get_destination_hostname(GProxyAddress
    * proxy);
extern guint16 g_proxy_address_get_destination_port(GProxyAddress
    * proxy);

```

```

extern const gchar *g_proxy_address_get_password(GProxyAddress *
proxy);
extern const gchar *g_proxy_address_get_protocol(GProxyAddress *
proxy);
extern GType g_proxy_address_get_type(void);
extern const gchar *g_proxy_address_get_username(GProxyAddress *
proxy);
extern GSocketAddress *g_proxy_address_new(GInetAddress * inetaddr,
                                           guint16 port,
                                           const gchar * protocol,
                                           const gchar * dest_hostname,
                                           guint16 dest_port,
                                           const gchar * username,
                                           const gchar * password);

```

17.12.100 glib-2.0/gio/gproxyaddressenumerator.h

```

#define G_TYPE_PROXY_ADDRESS_ENUMERATOR
(g_proxy_address_enumerator_get_type ())
#define G_PROXY_ADDRESS_ENUMERATOR_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_PROXY_ADDRESS_ENUMERATOR,
GProxyAddressEnumeratorClass))
#define G_IS_PROXY_ADDRESS_ENUMERATOR_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_PROXY_ADDRESS_ENUMERATOR))
#define G_PROXY_ADDRESS_ENUMERATOR(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_PROXY_ADDRESS_ENUMERATOR,
GProxyAddressEnumerator))
#define G_IS_PROXY_ADDRESS_ENUMERATOR(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_PROXY_ADDRESS_ENUMERATOR))
#define G_PROXY_ADDRESS_ENUMERATOR_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_PROXY_ADDRESS_ENUMERATOR,
GProxyAddressEnumeratorClass))

typedef struct _GProxyAddressEnumeratorClass {
    GSocketAddressEnumeratorClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
    void (*_g_reserved7) (void);
} GProxyAddressEnumeratorClass;
typedef struct _GProxyAddressEnumeratorPrivate
    GProxyAddressEnumeratorPrivate;
struct _GProxyAddressEnumerator {
    GSocketAddressEnumerator parent_instance;
    GProxyAddressEnumeratorPrivate *priv;
};
extern GType g_proxy_address_enumerator_get_type(void);

```

17.12.101 glib-2.0/gio/gproxyresolver.h

```

#define G_TYPE_PROXY_RESOLVER (g_proxy_resolver_get_type ())
#define G_PROXY_RESOLVER(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_PROXY_RESOLVER, GProxyResolver))
#define G_IS_PROXY_RESOLVER(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_PROXY_RESOLVER))
#define G_PROXY_RESOLVER_GET_IFACE(o)
(G_TYPE_INSTANCE_GET_INTERFACE ((o), G_TYPE_PROXY_RESOLVER,
GProxyResolverInterface))
#define G_PROXY_RESOLVER_EXTENSION_POINT_NAME "gio-proxy-
resolver"

```

```

typedef struct _GProxyResolverInterface {
    GTypeInterface g_iface;
    gboolean(*is_supported) (GProxyResolver * resolver);
    gchar **(*lookup) (GProxyResolver * resolver, const gchar * uri,
                      GCancellable * cancellable, GError * *error);
    void(*lookup_async) (GProxyResolver * resolver, const gchar *
uri,
                      GCancellable * cancellable,
                      GAsyncReadyCallback callback,
                      gpointer user_data);
    gchar **(*lookup_finish) (GProxyResolver * resolver,
                      GAsyncResult * result, GError * *error);
} GProxyResolverInterface;
extern GProxyResolver *g_proxy_resolver_get_default(void);
extern GType g_proxy_resolver_get_type(void);
extern gboolean g_proxy_resolver_is_supported(GProxyResolver *
resolver);
extern gchar **g_proxy_resolver_lookup(GProxyResolver * resolver,
const gchar * uri,
GCancellable * cancellable,
GError * *error);
extern void g_proxy_resolver_lookup_async(GProxyResolver *
resolver,
const gchar * uri,
GCancellable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
extern gchar **g_proxy_resolver_lookup_finish(GProxyResolver *
resolver,
GAsyncResult * result,
GError * *error);

```

17.12.102 glib-2.0/gio/gremoteactiongroup.h

```

#define G_TYPE_REMOTE_ACTION_GROUP
(g_remote_action_group_get_type ())
#define G_REMOTE_ACTION_GROUP(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_REMOTE_ACTION_GROUP,
GRemoteActionGroup))
#define G_IS_REMOTE_ACTION_GROUP(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_REMOTE_ACTION_GROUP))
#define G_REMOTE_ACTION_GROUP_GET_IFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst), G_TYPE_REMOTE_ACTION_GROUP,
GRemoteActionGroupInterface))

typedef struct _GRemoteActionGroupInterface {
    GTypeInterface g_iface;
    void(*activate_action_full) (GRemoteActionGroup * resolver,
const gchar * action_name,
GVariant * parameter,
GVariant * platform_data);
    void(*change_action_state_full) (GRemoteActionGroup * resolver,
const gchar * action_name,
GVariant * value,
GVariant * platform_data);
} GRemoteActionGroupInterface;
extern void
g_remote_action_group_activate_action_full(GRemoteActionGroup *
remote,
const gchar *
action_name,
GVariant *
parameter,
GVariant *

```

```

platform_data);

extern void
g_remote_action_group_change_action_state_full(GRemoteActionGroup
* remote,

const gchar * action_name,
GVariant * value,
GVariant * platform_data);

extern GType g_remote_action_group_get_type(void);

```

17.12.103 glib-2.0/gio/gresolver.h

```

#define G_RESOLVER_ERROR (g_resolver_error_quark ())
#define G_TYPE_RESOLVER (g_resolver_get_type ())
#define G_RESOLVER_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_RESOLVER, GResolverClass))
#define G_IS_RESOLVER_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_TYPE_RESOLVER))
#define G_RESOLVER(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_RESOLVER, GResolver))
#define G_IS_RESOLVER(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_RESOLVER))
#define G_RESOLVER_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o),
G_TYPE_RESOLVER, GResolverClass))

typedef struct _GResolverPrivate GResolverPrivate;
typedef struct _GResolverClass {
    GObjectClass parent_class;
    void (*reload) (GResolver * resolver);
    GList *(*lookup_by_name) (GResolver * resolver, const gchar *
hostname,

GCancelable * cancellable, GError *
*error);
    void (*lookup_by_name_async) (GResolver * resolver,
const gchar * hostname,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
    GList *(*lookup_by_name_finish) (GResolver * resolver,
GAsyncResult * result,
GError * *error);
    gchar *(*lookup_by_address) (GResolver * resolver,
GInetAddress * address,
GCancelable * cancellable,
GError * *error);
    void (*lookup_by_address_async) (GResolver * resolver,
GInetAddress * address,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
    gchar *(*lookup_by_address_finish) (GResolver * resolver,
GAsyncResult * result,
GError * *error);
    GList(*lookup_service) (GResolver * resolver, const gchar *
rrname,

GCancelable * cancellable, GError *
*error);
    void (*lookup_service_async) (GResolver * resolver,
const gchar * rrname,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);
    GList *(*lookup_service_finish) (GResolver * resolver,
GAsyncResult * result,
GError * *error);
    void (*_g_reserved1) (void);

```

```

    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
} GResolverClass;
struct _GResolver {
    GObject parent_instance;
    GResolverPrivate *priv;
};
extern GQuark g_resolver_error_quark(void);
extern void g_resolver_free_addresses(GList * addresses);
extern void g_resolver_free_targets(GList * targets);
extern GResolver *g_resolver_get_default(void);
extern GType g_resolver_get_type(void);
extern gchar *g_resolver_lookup_by_address(GResolver * resolver,
                                           GInetAddress * address,
                                           GCancellable * cancellable,
                                           GError * *error);
extern void g_resolver_lookup_by_address_async(GResolver * resolver,
                                              GInetAddress * address,
                                              GCancellable * cancellable,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer user_data);
extern gchar *g_resolver_lookup_by_address_finish(GResolver * resolver,
                                                  GAsyncResult * result,
                                                  GError * *error);
extern GList *g_resolver_lookup_by_name(GResolver * resolver,
                                        const gchar * hostname,
                                        GCancellable * cancellable,
                                        GError * *error);
extern void g_resolver_lookup_by_name_async(GResolver * resolver,
                                           const gchar * hostname,
                                           GCancellable * cancellable,
                                           GAsyncReadyCallback callback,
                                           gpointer user_data);
extern GList *g_resolver_lookup_by_name_finish(GResolver * resolver,
                                              GAsyncResult * result,
                                              GError * *error);
extern GList *g_resolver_lookup_service(GResolver * resolver,
                                       const gchar * service,
                                       const gchar * protocol,
                                       const gchar * domain,
                                       GCancellable * cancellable,
                                       GError * *error);
extern void g_resolver_lookup_service_async(GResolver * resolver,
                                           const gchar * service,
                                           const gchar * protocol,
                                           const gchar * domain,
                                           GCancellable * cancellable,
                                           GAsyncReadyCallback callback,
                                           gpointer user_data);
extern GList *g_resolver_lookup_service_finish(GResolver * resolver,
                                              GAsyncResult * result,
                                              GError * *error);
extern void g_resolver_set_default(GResolver * resolver);

```

17.12.104 glib-2.0/gio/gresource.h

```

#define G_RESOURCE_ERROR (g_resource_error_quark ())

```

```

#define G_TYPE_RESOURCE (g_resource_get_type ())

typedef struct _GStaticResource {
    const guint8 *data;
    gsize data_len;
    GResource *resource;
    GStaticResource *next;
    gpointer padding;
} GStaticResource;
extern void g_resource_enumerate_children(GResource * resource,
    const gchar * path,
    GResourceLookupFlags
    lookup_flags, GError * *error);
extern GQuark g_resource_error_quark(void);
extern gboolean g_resource_get_info(GResource * resource,
    const gchar * path,
    GResourceLookupFlags lookup_flags,
    gsize * size, guint32 * flags,
    GError * *error);
extern GType g_resource_get_type(void);
extern GResource *g_resource_load(const gchar * filename, GError *
    *error);
extern GBytes *g_resource_lookup_data(GResource * resource,
    const gchar * path,
    GResourceLookupFlags lookup_flags,
    GError * *error);
extern GResource *g_resource_new_from_data(GBytes * data, GError *
    *error);
extern GInputStream *g_resource_open_stream(GResource * resource,
    const gchar * path,
    GResourceLookupFlags
    lookup_flags, GError *
    *error);
extern GResource *g_resource_ref(GResource * resource);
extern void g_resource_unref(GResource * resource);
extern void g_resources_enumerate_children(const gchar * path,
    GResourceLookupFlags
    lookup_flags, GError * *error);
extern gboolean g_resources_get_info(const gchar * path,
    GResourceLookupFlags lookup_flags,
    gsize * size, guint32 * flags,
    GError * *error);
extern GBytes *g_resources_lookup_data(const gchar * path,
    GResourceLookupFlags lookup_flags,
    GError * *error);
extern GInputStream *g_resources_open_stream(const gchar * path,
    GResourceLookupFlags
    lookup_flags,
    GError * *error);
extern void g_resources_register(GResource * resource);
extern void g_resources_unregister(GResource * resource);
extern void g_static_resource_fini(GStaticResource *
    static_resource);
extern GResource *g_static_resource_get_resource(GStaticResource *
    static_resource);
extern void g_static_resource_init(GStaticResource *
    static_resource);

```

17.12.105 glib-2.0/gio/gseekable.h

```

#define G_TYPE_SEEKABLE (g_seekable_get_type ())
#define G_SEEKABLE(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
    G_TYPE_SEEKABLE, GSeekable))
#define G_IS_SEEKABLE(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
    G_TYPE_SEEKABLE))

```

```

#define G_SEEKABLE_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_SEEKABLE,
GSeekableIface))

typedef struct _GSeekableIface {
    GTypeInterface g_iface;
    goffset(*tell) (GSeekable * seekable);
    gboolean(*can_seek) (GSeekable * seekable);
    gboolean(*seek) (GSeekable * seekable, goffset offset,
GSeekType type,
GCancelable * cancellable, GError * *error);
    gboolean(*can_truncate) (GSeekable * seekable);
    gboolean(*truncate_fn) (GSeekable * seekable, goffset offset,
GCancelable * cancellable, GError *
*error);
} GSeekableIface;
extern gboolean g_seekable_can_seek(GSeekable * seekable);
extern gboolean g_seekable_can_truncate(GSeekable * seekable);
extern GType g_seekable_get_type(void);
extern gboolean g_seekable_seek(GSeekable * seekable, goffset
offset,
GSeekType type, GCancelable *
cancellable,
GError * *error);
extern goffset g_seekable_tell(GSeekable * seekable);
extern gboolean g_seekable_truncate(GSeekable * seekable, goffset
offset,
GCancelable * cancellable,
GError * *error);

```

17.12.106 glib-2.0/gio/gsettings.h

```

#define G_TYPE_SETTINGS (g_settings_get_type ())
#define G_SETTINGS_CLASS(class) (G_TYPE_CHECK_CLASS_CAST ((class),
G_TYPE_SETTINGS, GSettingsClass))
#define G_IS_SETTINGS_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_SETTINGS))
#define G_SETTINGS(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_SETTINGS, GSettings))
#define G_IS_SETTINGS(inst) (G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_SETTINGS))
#define G_SETTINGS_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_SETTINGS, GSettingsClass))

typedef struct _GSettingsPrivate GSettingsPrivate;
typedef struct _GSettingsClass {
    GObjectClass parent_class;
    void (*writable_changed) (GSettings * settings, const gchar *
key);
    void (*changed) (GSettings * settings, const gchar * key);
    gboolean(*writable_change_event) (GSettings * settings, GQuark
key);
    gboolean(*change_event) (GSettings * settings, const GQuark *
keys,
gint n_keys);
    gpointer padding[20];
} GSettingsClass;
struct _GSettings {
    GObject parent_instance;
    GSettingsPrivate *priv;
};
typedef GVariant *(*GSettingsBindSetMapping) (const GValue * value,
const GVariantType *
expected_type,
gpointer user_data);

```

```

typedef gboolean(*GSettingsBindGetMapping) (GValue * value,
                                           GVariant * variant,
                                           gpointer user_data);
typedef gboolean(*GSettingsGetMapping) (GVariant * value,
                                       gpointer * result,
                                       gpointer user_data);

typedef enum {
    G_SETTINGS_BIND_DEFAULT,
    G_SETTINGS_BIND_GET = (1 << 0),
    G_SETTINGS_BIND_SET = (1 << 1),
    G_SETTINGS_BIND_NO_SENSITIVITY = (1 << 2),
    G_SETTINGS_BIND_GET_NO_CHANGES = (1 << 3),
    G_SETTINGS_BIND_INVERT_BOOLEAN = (1 << 4)
} GSettingsBindFlags;
extern void g_settings_apply(GSettings * settings);
extern void g_settings_bind(GSettings * settings, const gchar * key,
                           void *object, const gchar * property,
                           GSettingsBindFlags flags);
extern void g_settings_bind_with_mapping(GSettings * settings,
                                         const gchar * key, void *object,
                                         const gchar * property,
                                         GSettingsBindFlags flags,
                                         GSettingsBindGetMapping
                                         get_mapping,
                                         GSettingsBindSetMapping
                                         set_mapping, void *user_data,
                                         GDestroyNotify destroy);
extern void g_settings_bind_writable(GSettings * settings,
                                     const gchar * key, void *object,
                                     const gchar * property,
                                     gboolean inverted);
extern GAction *g_settings_create_action(GSettings * settings,
                                         const gchar * key);
extern void g_settings_delay(GSettings * settings);
extern void g_settings_get(GSettings * settings, const gchar * key,
                           const gchar * format, ...);
extern gboolean g_settings_get_boolean(GSettings * settings,
                                       const gchar * key);
extern GSettings *g_settings_get_child(GSettings * settings,
                                       const gchar * name);
extern gdouble g_settings_get_double(GSettings * settings,
                                     const gchar * key);
extern gint g_settings_get_enum(GSettings * settings, const gchar
* key);
extern guint g_settings_get_flags(GSettings * settings, const gchar
* key);
extern gboolean g_settings_get_has_unapplied(GSettings * settings);
extern gint g_settings_get_int(GSettings * settings, const gchar *
key);
extern void *g_settings_get_mapped(GSettings * settings, const
gchar * key,
                                GSettingsGetMapping mapping,
                                void *user_data);
extern GVariant *g_settings_get_range(GSettings * settings,
                                       const gchar * key);
extern gchar *g_settings_get_string(GSettings * settings,
                                    const gchar * key);
extern gchar **g_settings_get_strv(GSettings * settings,
                                   const gchar * key);
extern GType g_settings_get_type(void);
extern guint g_settings_get_uint(GSettings * settings, const gchar
* key);
extern GVariant *g_settings_get_value(GSettings * settings,
                                      const gchar * key);
extern gboolean g_settings_is_writable(GSettings * settings,
                                      const gchar * name);

```



```

extern gchar **g_settings_list_children(GSettings * settings);
extern gchar **g_settings_list_keys(GSettings * settings);
extern      const      gchar      *const
*g_settings_list_relocatable_schemas(void);
extern const gchar *const *g_settings_list_schemas(void);
extern GSettings *g_settings_new(const gchar * schema_id);
extern GSettings *g_settings_new_full(GSettingsSchema * schema,
                                     GSettingsBackend * backend,
                                     const gchar * path);
extern GSettings *g_settings_new_with_backend(const gchar *
schema_id,
                                     GSettingsBackend * backend);
extern GSettings *g_settings_new_with_backend_and_path(const gchar *
*
                                     schema_id,
                                     GSettingsBackend * backend,
                                     backend,
                                     const gchar * path);
extern GSettings *g_settings_new_with_path(const gchar * schema_id,
const gchar * path);
extern gboolean g_settings_range_check(GSettings * settings,
const gchar * key,
GVariant * value);
extern void g_settings_reset(GSettings * settings, const gchar *
key);
extern void g_settings_revert(GSettings * settings);
extern gboolean g_settings_set(GSettings * settings, const gchar *
key,
const gchar * format, ...);
extern gboolean g_settings_set_boolean(GSettings * settings,
const gchar * key, gboolean value);
extern gboolean g_settings_set_double(GSettings * settings,
const gchar * key, gdouble value);
extern gboolean g_settings_set_enum(GSettings * settings,
const gchar * key, gint value);
extern gboolean g_settings_set_flags(GSettings * settings,
const gchar * key, guint value);
extern gboolean g_settings_set_int(GSettings * settings, const
gchar * key,
gint value);
extern gboolean g_settings_set_string(GSettings * settings,
const gchar * key,
const gchar * value);
extern gboolean g_settings_set_strv(GSettings * settings,
const gchar * key,
const gchar * const *value);
extern gboolean g_settings_set_uint(GSettings * settings,
const gchar * key, guint value);
extern gboolean g_settings_set_value(GSettings * settings,
const gchar * key, GVariant * value);
extern void g_settings_sync(void);
extern void g_settings_unbind(void *object, const gchar * property);

```

17.12.107 glib-2.0/gio/gsettingsbackend.h

```

#define G_TYPE_SETTINGS_BACKEND (g_settings_backend_get_type ())
#define G_SETTINGS_BACKEND_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_SETTINGS_BACKEND, GSettingsBackendClass))
#define G_IS_SETTINGS_BACKEND_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_SETTINGS_BACKEND))
#define G_SETTINGS_BACKEND(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_SETTINGS_BACKEND, GSettingsBackend))
#define G_IS_SETTINGS_BACKEND(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_SETTINGS_BACKEND))

```

```

#define G_SETTINGS_BACKEND_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_SETTINGS_BACKEND,
GSettingsBackendClass))
#define G_SETTINGS_BACKEND_EXTENSION_POINT_NAME "gsettings-
backend"

typedef struct _GSettingsBackendPrivate GSettingsBackendPrivate;
typedef struct _GSettingsBackendClass {
    GObjectClass parent_class;
    GVariant *(*read) (GSettingsBackend * backend, const gchar *
key,
                    const GVariantType * expected_type,
                    gboolean default_value);
    gboolean(*get_writable) (GSettingsBackend * backend,
                    const gchar * key);
    gboolean(*write) (GSettingsBackend * backend, const gchar * key,
                    GVariant * value, gpointer origin_tag);
    gboolean(*write_tree) (GSettingsBackend * backend, GTree * tree,
                    gpointer origin_tag);
    void (*reset) (GSettingsBackend * backend, const gchar * key,
                    gpointer origin_tag);
    void (*subscribe) (GSettingsBackend * backend, const gchar *
name);
    void (*unsubscribe) (GSettingsBackend * backend, const gchar *
name);
    void (*sync) (GSettingsBackend * backend);
    GPermission *(*get_permission) (GSettingsBackend * backend,
                    const gchar * path);
    gpointer padding[24];
} GSettingsBackendClass;
struct _GSettingsBackend {
    GObject parent_instance;
    GSettingsBackendPrivate *priv;
};
extern GSettingsBackend *g_keyfile_settings_backend_new(const
gchar *
                    filename,
                    const gchar *
                    root_path,
                    const gchar *
                    root_group);
extern GSettingsBackend *g_memory_settings_backend_new(void);
extern GSettingsBackend *g_null_settings_backend_new(void);
extern void g_settings_backend_changed(GSettingsBackend * backend,
                    const gchar * key,
                    gpointer origin_tag);
extern void g_settings_backend_changed_tree(GSettingsBackend *
backend,
                    GTree * tree,
                    gpointer origin_tag);
extern void g_settings_backend_flatten_tree(GTree * tree, gchar *
*path,
                    const gchar * **keys,
                    GVariant * **values);
extern GSettingsBackend *g_settings_backend_get_default(void);
extern GType g_settings_backend_get_type(void);
extern void g_settings_backend_keys_changed(GSettingsBackend *
backend,
                    const gchar * path,
                    const gchar * const *items,
                    gpointer origin_tag);
extern void g_settings_backend_path_changed(GSettingsBackend *
backend,
                    const gchar * path,
                    gpointer origin_tag);

```

```

extern void g_settings_backend_path_writable_changed(GSettingsBackend *
                                                    backend,
                                                    const gchar * path);
extern void g_settings_backend_writable_changed(GSettingsBackend *
                                                backend,
                                                const gchar * key);

```

17.12.108 glib-2.0/gio/gsettingsschema.h

```

#define G_TYPE_SETTINGS_SCHEMA (g_settings_schema_get_type ())
#define G_TYPE_SETTINGS_SCHEMA_SOURCE (g_settings_schema_source_get_type ())

typedef struct _GSettingsSchemaSource GSettingsSchemaSource;
typedef struct _GSettingsSchema GSettingsSchema;
extern const gchar *g_settings_schema_get_id(GSettingsSchema *
schema);
extern const gchar *g_settings_schema_get_path(GSettingsSchema *
schema);
extern GType g_settings_schema_get_type(void);
extern GSettingsSchema *g_settings_schema_ref(GSettingsSchema *
schema);
extern GSettingsSchemaSource *g_settings_schema_source_ref(GSettingsSchemaSource
*source);
extern GType g_settings_schema_source_get_type(void);
extern GSettingsSchema *g_settings_schema_source_lookup(GSettingsSchemaSource
*source,
const gchar * schema_id,
gboolean recursive);
extern GSettingsSchemaSource *g_settings_schema_source_new_from_directory(const
gchar *
directory,
GSettingsSchemaSource *
parent, gboolean trusted,
GError * *error);
extern GSettingsSchemaSource *g_settings_schema_source_ref(GSettingsSchemaSource
*source);
extern void g_settings_schema_source_unref(GSettingsSchemaSource *
source);
extern void g_settings_schema_unref(GSettingsSchema * schema);

```

17.12.109 glib-2.0/gio/gsimpleaction.h

```

#define G_TYPE_SIMPLE_ACTION (g_simple_action_get_type ())
#define G_SIMPLE_ACTION(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_SIMPLE_ACTION, GSimpleAction))
#define G_IS_SIMPLE_ACTION(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_SIMPLE_ACTION))

extern GType g_simple_action_get_type(void);
extern GSimpleAction *g_simple_action_new(const gchar * name,
GVariantType * parameter_type);
extern GSimpleAction *g_simple_action_new_stateful(const gchar *
name,
GVariantType *
parameter_type,
GVariant * state);
extern void g_simple_action_set_enabled(GSimpleAction * simple,
gboolean enabled);
extern void g_simple_action_set_state(GSimpleAction * simple,
GVariant * value);

```

17.12.110 glib-2.0/gio/gsimpleactiongroup.h

```

#define G_TYPE_SIMPLE_ACTION_GROUP
(g_simple_action_group_get_type ())
#define G_SIMPLE_ACTION_GROUP_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_SIMPLE_ACTION_GROUP,
GSimpleActionGroupClass))
#define G_IS_SIMPLE_ACTION_GROUP_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_SIMPLE_ACTION_GROUP))
#define G_SIMPLE_ACTION_GROUP(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_SIMPLE_ACTION_GROUP,
GSimpleActionGroup))
#define G_IS_SIMPLE_ACTION_GROUP(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_SIMPLE_ACTION_GROUP))
#define G_SIMPLE_ACTION_GROUP_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_SIMPLE_ACTION_GROUP,
GSimpleActionGroupClass))

typedef struct _GSimpleActionGroupPrivate
GSimpleActionGroupPrivate;
typedef struct _GSimpleActionGroupClass {
    GObjectClass parent_class;
    gpointer padding[12];
} GSimpleActionGroupClass;
struct _GSimpleActionGroup {
    GObject parent_instance;
    GSimpleActionGroupPrivate *priv;
};
extern void g_simple_action_group_add_entries(GSimpleActionGroup *
simple,
const GActionEntry * entries,
gint n_entries,
gpointer user_data);
extern GType g_simple_action_group_get_type(void);
extern void g_simple_action_group_insert(GSimpleActionGroup *
simple,
GAction * action);
extern GAction *g_simple_action_group_lookup(GSimpleActionGroup *
simple,
const gchar * action_name);
extern GSimpleActionGroup *g_simple_action_group_new(void);
extern void g_simple_action_group_remove(GSimpleActionGroup *
simple,
const gchar * action_name);

```

17.12.111 glib-2.0/gio/gsimpleasyncresult.h

```

#define G_TYPE_SIMPLE_ASYNC_RESULT
(g_simple_async_result_get_type ())
#define G_SIMPLE_ASYNC_RESULT_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST ((k), G_TYPE_SIMPLE_ASYNC_RESULT,
GSimpleAsyncResultClass))
#define G_IS_SIMPLE_ASYNC_RESULT_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_SIMPLE_ASYNC_RESULT))
#define G_SIMPLE_ASYNC_RESULT(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_SIMPLE_ASYNC_RESULT, GSimpleAsyncResult))
#define G_IS_SIMPLE_ASYNC_RESULT(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_SIMPLE_ASYNC_RESULT))
#define G_SIMPLE_ASYNC_RESULT_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_SIMPLE_ASYNC_RESULT,
GSimpleAsyncResultClass))

typedef struct _GSimpleAsyncResultClass GSimpleAsyncResultClass;

```

```

extern void g_simple_async_report_error_in_idle(GObject * object,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer user_data,
                                              GQuark domain, gint code,
                                              const char *format, ...);
extern void g_simple_async_report_gerror_in_idle(GObject * object,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer user_data,
                                              const GError * error);
extern void g_simple_async_report_take_gerror_in_idle(GObject *
object,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer user_data,
                                              GError * error);

extern void g_simple_async_result_complete(GSimpleAsyncResult *
simple);
extern void g_simple_async_result_complete_in_idle(GSimpleAsyncResult *
simple);

extern gboolean
g_simple_async_result_get_op_res_gboolean(GSimpleAsyncResult *
simple);
extern gpointer
g_simple_async_result_get_op_res_gpointer(GSimpleAsyncResult *
simple);
extern gssize
g_simple_async_result_get_op_res_gssize(GSimpleAsyncResult *
simple);

extern gpointer
g_simple_async_result_get_source_tag(GSimpleAsyncResult *
simple);
extern GType g_simple_async_result_get_type(void);
extern gboolean g_simple_async_result_is_valid(GAsyncResult *
result,
                                              GObject * source,
                                              gpointer source_tag);
extern GSimpleAsyncResult *g_simple_async_result_new(GObject *
source_object,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer user_data,
                                              gpointer source_tag);
extern GSimpleAsyncResult *g_simple_async_result_new_error(GObject *
source_object,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer
                                              user_data,
                                              GQuark domain,
                                              gint code,
                                              const char
                                              *format, ...);
extern GSimpleAsyncResult *g_simple_async_result_new_from_error(GObject *
source_object,
                                              GAsyncReadyCallback
                                              callback,
                                              gpointer
                                              user_data,

```

```

const
GError *
error);
extern
GSimpleAsyncResult
*g_simple_async_result_new_take_error(GObject *
source_object,
GAsyncReadyCallback
callback,
gpointer
user_data,
GError *
error);
extern
gboolean
g_simple_async_result_propagate_error(GSimpleAsyncResult *
simple,
GError * *dest);
extern void g_simple_async_result_run_in_thread(GSimpleAsyncResult
*
simple,
GSimpleAsyncThreadFunc
func, int io_priority,
GCancelable *
cancellable);
extern
void
g_simple_async_result_set_check_cancellable(GSimpleAsyncResult
* simple,
GCancelable *
check_cancellable);
extern void g_simple_async_result_set_error(GSimpleAsyncResult *
simple,
GQuark domain, gint code,
const char *format, ...);
extern void g_simple_async_result_set_error_va(GSimpleAsyncResult
* simple,
GQuark domain, gint code,
const char *format,
va_list args);
extern
void
g_simple_async_result_set_from_error(GSimpleAsyncResult *
simple,
const GError * error);
extern void
g_simple_async_result_set_handle_cancellation(GSimpleAsyncResult *
simple,
gboolean
handle_cancellation);
extern
void
g_simple_async_result_set_op_res_gboolean(GSimpleAsyncResult *
simple,
gboolean op_res);
extern
void
g_simple_async_result_set_op_res_gpointer(GSimpleAsyncResult *
simple,
gpointer op_res,
GDestroyNotify
destroy_op_res);
extern
void
g_simple_async_result_set_op_res_gssize(GSimpleAsyncResult *
simple, gssize op_res);
extern void g_simple_async_result_take_error(GSimpleAsyncResult *
simple,
GError * error);

```

17.12.112 glib-2.0/gio/gsimplepermission.h

```

#define G_TYPE_SIMPLE_PERMISSION
(g_simple_permission_get_type ())
#define G_SIMPLE_PERMISSION(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_SIMPLE_PERMISSION, GSimplePermission))
#define G_IS_SIMPLE_PERMISSION(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_SIMPLE_PERMISSION))

extern GType g_simple_permission_get_type(void);
extern GPermission *g_simple_permission_new(gboolean allowed);

```

17.12.113 glib-2.0/gio/gsocket.h

```

#define G_TYPE_SOCKET (g_socket_get_type ())
#define G_SOCKET_CLASS(class) (G_TYPE_CHECK_CLASS_CAST ((class),
G_TYPE_SOCKET, GSocketClass))
#define G_IS_SOCKET_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_SOCKET))
#define G_SOCKET(inst) (G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_SOCKET, GSocket))
#define G_IS_SOCKET(inst) (G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_SOCKET))
#define G_SOCKET_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_SOCKET, GSocketClass))

typedef struct _GSocketPrivate GSocketPrivate;
typedef struct _GSocketClass {
    GObjectClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
    void (*_g_reserved7) (void);
    void (*_g_reserved8) (void);
    void (*_g_reserved9) (void);
    void (*_g_reserved10) (void);
} GSocketClass;
struct _GSocket {
    GObject parent_instance;
    GSocketPrivate *priv;
};

extern GSocket *g_socket_accept(GSocket * socket,
                                GCancellable * cancellable,
                                GError * *error);
extern gboolean g_socket_bind(GSocket * socket, GSocketAddress *
address,
                                gboolean allow_reuse, GError * *error);
extern gboolean g_socket_check_connect_result(GSocket * socket,
                                                GError * *error);
extern gboolean g_socket_close(GSocket * socket, GError * *error);
extern GIOCondition g_socket_condition_check(GSocket * socket,
                                                GIOCondition condition);
extern gboolean g_socket_condition_timed_wait(GSocket * socket,
                                                GIOCondition condition,
                                                gint64 timeout,
                                                GCancellable * cancellable,
                                                GError * *error);
extern gboolean g_socket_condition_wait(GSocket * socket,
                                                GIOCondition condition,
                                                GCancellable * cancellable,

```

```

        GError * *error);
extern gboolean g_socket_connect(GSocket * socket,
                                GSocketAddress * address,
                                Gancellable * cancellable,
                                GError * *error);
extern GSource *g_socket_create_source(GSocket * socket,
                                       GIOCondition condition,
                                       Gancellable * cancellable);
extern gssize g_socket_get_available_bytes(GSocket * socket);
extern gboolean g_socket_get_blocking(GSocket * socket);
extern gboolean g_socket_get_broadcast(GSocket * socket);
extern GCredentials *g_socket_get_credentials(GSocket * socket,
                                              GError * *error);
extern GSocketFamily g_socket_get_family(GSocket * socket);
extern int g_socket_get_fd(GSocket * socket);
extern gboolean g_socket_get_keepalive(GSocket * socket);
extern gint g_socket_get_listen_backlog(GSocket * socket);
extern GSocketAddress *g_socket_get_local_address(GSocket * socket,
                                                  GError * *error);
extern gboolean g_socket_get_multicast_loopback(GSocket * socket);
extern guint g_socket_get_multicast_ttl(GSocket * socket);
extern GSocketProtocol g_socket_get_protocol(GSocket * socket);
extern GSocketAddress *g_socket_get_remote_address(GSocket *
socket,
                                                  GError * *error);
extern GSocketType g_socket_get_socket_type(GSocket * socket);
extern guint g_socket_get_timeout(GSocket * socket);
extern guint g_socket_get_ttl(GSocket * socket);
extern GType g_socket_get_type(void);
extern gboolean g_socket_is_closed(GSocket * socket);
extern gboolean g_socket_is_connected(GSocket * socket);
extern gboolean g_socket_join_multicast_group(GSocket * socket,
                                              GInetAddress * group,
                                              gboolean source_specific,
                                              const char *iface,
                                              GError * *error);
extern gboolean g_socket_leave_multicast_group(GSocket * socket,
                                              GInetAddress * group,
                                              gboolean source_specific,
                                              const char *iface,
                                              GError * *error);
extern gboolean g_socket_listen(GSocket * socket, GError * *error);
extern GSocket *g_socket_new(GSocketFamily family, GSocketType type,
                             GSocketProtocol protocol, GError * *error);
extern GSocket *g_socket_new_from_fd(gint fd, GError * *error);
extern gssize g_socket_receive(GSocket * socket, gchar * buffer,
                               gsize size, Gancellable * cancellable,
                               GError * *error);
extern gssize g_socket_receive_from(GSocket * socket,
                                    GSocketAddress * *address,
                                    gchar * buffer, gsize size,
                                    Gancellable * cancellable,
                                    GError * *error);
extern gssize g_socket_receive_message(GSocket * socket,
                                       GSocketAddress * *address,
                                       GInputVector * vectors,
                                       gint num_vectors,
                                       GSocketControlMessage *
**messages,
                                       gint * num_messages, gint * flags,
                                       Gancellable * cancellable,
                                       GError * *error);
extern gssize g_socket_receive_with_blocking(GSocket * socket,
                                             gchar * buffer, gsize size,
                                             gboolean blocking,
                                             Gancellable * cancellable,

```



```

        GError * *error);
extern gssize g_socket_send(GSocket * socket, const char *buffer,
                           gsize size, GCancellable * cancellable,
                           GError * *error);
extern gssize g_socket_send_message(GSocket * socket,
                                   GSocketAddress * address,
                                   GOutputVector * vectors,
                                   gint num_vectors,
                                   GSocketControlMessage * *messages,
                                   gint num_messages, gint flags,
                                   GCancellable * cancellable,
                                   GError * *error);
extern gssize g_socket_send_to(GSocket * socket, GSocketAddress *
address,
                              const char *buffer, gsize size,
                              GCancellable * cancellable,
                              GError * *error);
extern gssize g_socket_send_with_blocking(GSocket * socket,
                                          const char *buffer, gsize size,
                                          gboolean blocking,
                                          GCancellable * cancellable,
                                          GError * *error);
extern void g_socket_set_blocking(GSocket * socket, gboolean
blocking);
extern void g_socket_set_broadcast(GSocket * socket, gboolean
broadcast);
extern void g_socket_set_keepalive(GSocket * socket, gboolean
keepalive);
extern void g_socket_set_listen_backlog(GSocket * socket, gint
backlog);
extern void g_socket_set_multicast_loopback(GSocket * socket,
                                             gboolean loopback);
extern void g_socket_set_multicast_ttl(GSocket * socket, guint ttl);
extern void g_socket_set_timeout(GSocket * socket, guint timeout);
extern void g_socket_set_ttl(GSocket * socket, guint ttl);
extern gboolean g_socket_shutdown(GSocket * socket, gboolean
shutdown_read,
                                  gboolean shutdown_write,
                                  GError * *error);
extern gboolean g_socket_speaks_ipv4(GSocket * socket);

```

17.12.114 glib-2.0/gio/gsocketaddress.h

```

#define G_TYPE_SOCKET_ADDRESS (g_socket_address_get_type ())
#define G_SOCKET_ADDRESS_CLASS(k) G_TYPE_SOCKET_ADDRESS_CLASS(k)
#define G_TYPE_SOCKET_ADDRESS_CHECK_CLASS_CAST(k, G_TYPE_SOCKET_ADDRESS, GSocketAddressClass)
#define G_IS_SOCKET_ADDRESS_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_SOCKET_ADDRESS))
#define G_SOCKET_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_SOCKET_ADDRESS, GSocketAddress))
#define G_IS_SOCKET_ADDRESS(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_SOCKET_ADDRESS))
#define G_SOCKET_ADDRESS_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_SOCKET_ADDRESS, GSocketAddressClass))

typedef struct _GSocketAddressClass {
    GObjectClass parent_class;
    GSocketFamily(*get_family) (GSocketAddress * address);
    gssize(*get_native_size) (GSocketAddress * address);
    gboolean(*to_native) (GSocketAddress * address, gpointer dest, gsize destlen, GError * *error);
} GSocketAddressClass;
struct _GSocketAddress {
    GObject parent_instance;

```

```

};
extern GSocketFamily g_socket_address_get_family(GSocketAddress *
address);
extern gssize g_socket_address_get_native_size(GSocketAddress *
address);
extern GType g_socket_address_get_type(void);
extern GSocketAddress *g_socket_address_new_from_native(gpointer
native,
                                gsize len);
extern gboolean g_socket_address_to_native(GSocketAddress *
address,
                                gpointer dest, gsize destlen,
                                GError * *error);

```

17.12.115 glib-2.0/gio/gsocketaddressenumerator.h

```

#define G_TYPE_SOCKET_ADDRESS_ENUMERATOR
(g_socket_address_enumerator_get_type ())
#define G_SOCKET_ADDRESS_ENUMERATOR_CLASS(k)
(G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_SOCKET_ADDRESS_ENUMERATOR,
GSocketAddressEnumeratorClass))
#define G_IS_SOCKET_ADDRESS_ENUMERATOR_CLASS(k)
(G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_SOCKET_ADDRESS_ENUMERATOR))
#define G_SOCKET_ADDRESS_ENUMERATOR(o)
(G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_SOCKET_ADDRESS_ENUMERATOR,
GSocketAddressEnumerator))
#define G_IS_SOCKET_ADDRESS_ENUMERATOR(o)
(G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_SOCKET_ADDRESS_ENUMERATOR))
#define G_SOCKET_ADDRESS_ENUMERATOR_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_SOCKET_ADDRESS_ENUMERATOR,
GSocketAddressEnumeratorClass))

typedef struct _GSocketAddressEnumeratorClass {
    GObjectClass parent_class;
    GSocketAddress *(*next) (GSocketAddressEnumerator * enumerator,
GCancelable * cancellable, GError *
*error);
    void (*next_async) (GSocketAddressEnumerator * enumerator,
GCancelable * cancellable,
GAsyncReadyCallback callback, gpointer
user_data);
    GSocketAddress *(*next_finish) (GSocketAddressEnumerator *
enumerator,
GAsyncResult * result,
GError * *error);
} GSocketAddressEnumeratorClass;
struct _GSocketAddressEnumerator {
    GObject parent_instance;
};
extern GType g_socket_address_enumerator_get_type(void);
extern GSocketAddress
*g_socket_address_enumerator_next(GSocketAddressEnumerator *
enumerator,
GCancelable * cancellable,
GError * *error);

extern void
g_socket_address_enumerator_next_async(GSocketAddressEnumerator
* enumerator,
GCancelable *
cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);

extern GSocketAddress

```

```

*g_socket_address_enumerator_next_finish(GSocketAddressEnumerator
*
enumerator,
GAsyncResult * result,
GError * *error);

```

17.12.116 glib-2.0/gio/gsocketclient.h

```

#define G_TYPE_SOCKET_CLIENT (g_socket_client_get_type ())
#define G_SOCKET_CLIENT_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_SOCKET_CLIENT, GSocketClientClass))
#define G_IS_SOCKET_CLIENT_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_SOCKET_CLIENT))
#define G_SOCKET_CLIENT(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_SOCKET_CLIENT, GSocketClient))
#define G_IS_SOCKET_CLIENT(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_SOCKET_CLIENT))
#define G_SOCKET_CLIENT_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_SOCKET_CLIENT, GSocketClientClass))

typedef struct _GSocketClientPrivate GSocketClientPrivate;
typedef struct _GSocketClientClass {
    GObjectClass parent_class;
    void (*event) (GSocketClient * client, GSocketClientEvent event,
        GSocketConnectable * connectable,
        GIOStream * connection);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
} GSocketClientClass;
struct _GSocketClient {
    GObject parent_instance;
    GSocketClientPrivate *priv;
};

extern void g_socket_client_add_application_proxy(GSocketClient *
client,
const gchar * protocol);

extern GSocketConnection *g_socket_client_connect(GSocketClient *
client,
GSocketConnectable *
connectable,
GCancelable *
cancellable,
GError * *error);

extern void g_socket_client_connect_async(GSocketClient * client,
GSocketConnectable *
connectable,
GCancelable * cancellable,
GAsyncReadyCallback callback,
gpointer user_data);

extern GSocketConnection *g_socket_client_connect_finish(GSocketClient *
client,
GAsyncResult *
result,
GError * *error);

extern GSocketConnection *g_socket_client_connect_to_host(GSocketClient *
client,
const gchar *
host_and_port,
guint16
default_port,

```

```

        Gancellable *
        cancellable,
        GError * *error);
extern void g_socket_client_connect_to_host_async(GSocketClient *
client,
        const gchar *
        host_and_port,
        guint16 default_port,
        Gancellable *
        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern GSocketConnection
        *g_socket_client_connect_to_host_finish(GSocketClient * client,
        GAsyncResult * result,
        GError * *error);
extern
        GSocketConnection
        *g_socket_client_connect_to_service(GSocketClient
        * client,
        const gchar *
        domain,
        const gchar *
        service,
        Gancellable *
        cancellable,
        GError *
        *error);
extern void g_socket_client_connect_to_service_async(GSocketClient
*
        client,
        const gchar * domain,
        const gchar * service,
        Gancellable *
        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern GSocketConnection
        *g_socket_client_connect_to_service_finish(GSocketClient *
client,
        GAsyncResult * result,
        GError * *error);
extern
        GSocketConnection
        *g_socket_client_connect_to_uri(GSocketClient *
        client,
        const gchar * uri,
        guint16
        default_port,
        Gancellable *
        cancellable,
        GError * *error);
extern void g_socket_client_connect_to_uri_async(GSocketClient *
client,
        const gchar * uri,
        guint16 default_port,
        Gancellable *
        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern GSocketConnection
        *g_socket_client_connect_to_uri_finish(GSocketClient * client,
        GAsyncResult * result,
        GError * *error);

```

```

extern gboolean g_socket_client_get_enable_proxy(GSocketClient *
client);
extern GSocketFamily g_socket_client_get_family(GSocketClient *
client);
extern
GSocketAddress
*g_socket_client_get_local_address(GSocketClient *
client);
extern GSocketProtocol g_socket_client_get_protocol(GSocketClient
*
client);
extern GSocketType g_socket_client_get_socket_type(GSocketClient *
client);
extern guint g_socket_client_get_timeout(GSocketClient * client);
extern gboolean g_socket_client_get_tls(GSocketClient * client);
extern GTlsCertificateFlags
g_socket_client_get_tls_validation_flags(GSocketClient * client);
extern GType g_socket_client_get_type(void);
extern GSocketClient *g_socket_client_new(void);
extern void g_socket_client_set_enable_proxy(GSocketClient *
client,
gboolean enable);
extern void g_socket_client_set_family(GSocketClient * client,
GSocketFamily family);
extern void g_socket_client_set_local_address(GSocketClient *
client,
GSocketAddress * address);
extern void g_socket_client_set_protocol(GSocketClient * client,
GSocketProtocol protocol);
extern void g_socket_client_set_socket_type(GSocketClient * client,
GSocketType type);
extern void g_socket_client_set_timeout(GSocketClient * client,
guint timeout);
extern void g_socket_client_set_tls(GSocketClient * client,
gboolean tls);
extern void g_socket_client_set_tls_validation_flags(GSocketClient
*
client,
GTlsCertificateFlags
flags);

```

17.12.117 glib-2.0/gio/gsocketconnectable.h

```

#define G_TYPE_SOCKET_CONNECTABLE
(g_socket_connectable_get_type ())
#define G_SOCKET_CONNECTABLE(obj) (G_TYPE_CHECK_INSTANCE_CAST
((obj), G_TYPE_SOCKET_CONNECTABLE, GSocketConnectable))
#define G_IS_SOCKET_CONNECTABLE(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), G_TYPE_SOCKET_CONNECTABLE))
#define G_SOCKET_CONNECTABLE_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_SOCKET_CONNECTABLE,
GSocketConnectableIface))

typedef struct _GSocketConnectableIface {
GTypeInterface g_iface;
GSocketAddressEnumerator *(*enumerate) (GSocketConnectable *
connectable);
GSocketAddressEnumerator *(*proxy_enumerate)
(GSocketConnectable *
connectable);
} GSocketConnectableIface;
extern GSocketAddressEnumerator
*g_socket_connectable_enumerate(GSocketConnectable *
connectable);
extern GType g_socket_connectable_get_type(void);
extern GSocketAddressEnumerator

```

```
*g_socket_connectable_proxy_enumerate(GSocketConnectable *
                                     connectable);
```

17.12.118 glib-2.0/gio/gsocketconnection.h

```
#define G_TYPE_SOCKET_CONNECTION
(g_socket_connection_get_type ())
#define G_SOCKET_CONNECTION_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_SOCKET_CONNECTION,
GSocketConnectionClass))
#define G_IS_SOCKET_CONNECTION_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_SOCKET_CONNECTION))
#define G_SOCKET_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_SOCKET_CONNECTION,
GSocketConnection))
#define G_IS_SOCKET_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_SOCKET_CONNECTION))
#define G_SOCKET_CONNECTION_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_SOCKET_CONNECTION,
GSocketConnectionClass))

typedef struct _GSocketConnectionPrivate GSocketConnectionPrivate;
typedef struct _GSocketConnectionClass {
    GIOStreamClass parent_class;
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
} GSocketConnectionClass;
struct _GSocketConnection {
    GIOStream parent_instance;
    GSocketConnectionPrivate *priv;
};
extern gboolean g_socket_connection_connect(GSocketConnection *
connection,
                                           GSocketAddress * address,
                                           GCancellable * cancellable,
                                           GError * *error);
extern void g_socket_connection_connect_async(GSocketConnection *
connection,
                                           GSocketAddress * address,
                                           GCancellable * cancellable,
                                           GAsyncReadyCallback
callback,
                                           gpointer user_data);
extern gboolean g_socket_connection_connect_finish(GSocketConnection *
connection,
                                           GAsyncResult * result,
                                           GError * *error);
extern GSocketConnection
*g_socket_connection_factory_create_connection(GSocket
socket);
extern GType g_socket_connection_factory_lookup_type(GSocketFamily
family,
                                           GSocketType type,
                                           gint protocol_id);
extern void g_socket_connection_factory_register_type(GType g_type,
                                           GSocketFamily family,
                                           GSocketType type,
                                           gint protocol);
extern GSocketAddress
*g_socket_connection_get_local_address(GSocketConnection *
connection,
```

```

        GError * *error);

extern GSocketAddress
    *g_socket_connection_get_remote_address(GSocketConnection
        connection,
            GError * *error);

extern GSocket *g_socket_connection_get_socket(GSocketConnection *
        connection);

extern GType g_socket_connection_get_type(void);

extern gboolean g_socket_connection_is_connected(GSocketConnection
    *
        connection);

```

17.12.119 glib-2.0/gio/gsocketcontrolmessage.h

```

#define G_TYPE_SOCKET_CONTROL_MESSAGE
(g_socket_control_message_get_type ())
#define G_SOCKET_CONTROL_MESSAGE_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_SOCKET_CONTROL_MESSAGE,
GSocketControlMessageClass))
#define G_IS_SOCKET_CONTROL_MESSAGE_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_SOCKET_CONTROL_MESSAGE))
#define G_SOCKET_CONTROL_MESSAGE(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_SOCKET_CONTROL_MESSAGE,
GSocketControlMessage))
#define G_IS_SOCKET_CONTROL_MESSAGE(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_SOCKET_CONTROL_MESSAGE))
#define G_SOCKET_CONTROL_MESSAGE_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_SOCKET_CONTROL_MESSAGE,
GSocketControlMessageClass))

typedef struct _GSocketControlMessagePrivate
GSocketControlMessagePrivate;

typedef struct _GSocketControlMessageClass {
    GObjectClass parent_class;
    gsize(*get_size) (GSocketControlMessage * message);
    int (*get_level) (GSocketControlMessage * message);
    int (*get_type) (GSocketControlMessage * message);
    void (*serialize) (GSocketControlMessage * message, gpointer
data);
    GSocketControlMessage *(*deserialize) (int level, int type,
gsize size,
        gpointer data);

    void (*g_reserved1) (void);
    void (*g_reserved2) (void);
    void (*g_reserved3) (void);
    void (*g_reserved4) (void);
    void (*g_reserved5) (void);
} GSocketControlMessageClass;

struct _GSocketControlMessage {
    GObject parent_instance;
    GSocketControlMessagePrivate *priv;
};

extern GSocketControlMessage
*g_socket_control_message_deserialize(int
        level,
        int
        type,
        gsize
        size,
        gpointer
        data);

extern int
g_socket_control_message_get_level(GSocketControlMessage *
        message);

```

```

extern                                     int
g_socket_control_message_get_msg_type(GSocketControlMessage *
                                     message);

extern                                     gsize
g_socket_control_message_get_size(GSocketControlMessage *
                                     message);

extern GType g_socket_control_message_get_type(void);

extern                                     void
g_socket_control_message_serialize(GSocketControlMessage *
                                     message, gpointer data);

```

17.12.120 glib-2.0/gio/gsocketlistener.h

```

#define G_TYPE_SOCKET_LISTENER (g_socket_listener_get_type ())
#define G_SOCKET_LISTENER_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_SOCKET_LISTENER, GSocketListenerClass))
#define G_IS_SOCKET_LISTENER_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_SOCKET_LISTENER))
#define G_SOCKET_LISTENER(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_SOCKET_LISTENER, GSocketListener))
#define G_IS_SOCKET_LISTENER(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_SOCKET_LISTENER))
#define G_SOCKET_LISTENER_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_SOCKET_LISTENER,
GSocketListenerClass))

typedef struct _GSocketListenerPrivate GSocketListenerPrivate;
typedef struct _GSocketListenerClass {
    GObjectClass parent_class;
    void (*changed) (GSocketListener * listener);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
} GSocketListenerClass;
struct _GSocketListener {
    GObject parent_instance;
    GSocketListenerPrivate *priv;
};

extern GSocketConnection *g_socket_listener_accept(GSocketListener
*
                                     listener,
                                     GObject *
                                     *source_object,
                                     GCancellable *
                                     cancellable,
                                     GError * *error);

extern void g_socket_listener_accept_async(GSocketListener *
listener,
                                     GCancellable * cancellable,
                                     GAsyncReadyCallback callback,
                                     gpointer user_data);

extern GSocketConnection
*g_socket_listener_accept_finish(GSocketListener *
listener,
                                     GAsyncResult *
                                     result,
                                     GObject *
                                     *source_object,
                                     GError * *error);

extern GSocket *g_socket_listener_accept_socket(GSocketListener *
listener,
                                     GObject * *source_object,

```



```

        Gancellable * cancellable,
        GError * *error);

extern void g_socket_listener_accept_socket_async(GSocketListener
*
        listener,
        Gancellable *
        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);

extern GSocket
*g_socket_listener_accept_socket_finish(GSocketListener *
        listener,
        GAsyncResult *
        result,
        GObject *
        *source_object,
        GError * *error);

extern gboolean g_socket_listener_add_address(GSocketListener *
listener,
        GSocketAddress * address,
        GSocketType type,
        GSocketProtocol protocol,
        GObject * source_object,
        GSocketAddress *
        *effective_address,
        GError * *error);

extern guint16 g_socket_listener_add_any_inet_port(GSocketListener
*
        listener,
        GObject * source_object,
        GError * *error);

extern gboolean g_socket_listener_add_inet_port(GSocketListener *
listener,
        guint16 port,
        GObject * source_object,
        GError * *error);

extern gboolean g_socket_listener_add_socket(GSocketListener *
listener,
        GSocket * socket,
        GObject * source_object,
        GError * *error);

extern void g_socket_listener_close(GSocketListener * listener);
extern GType g_socket_listener_get_type(void);
extern GSocketListener *g_socket_listener_new(void);
extern void g_socket_listener_set_backlog(GSocketListener *
listener,
        int listen_backlog);

```

17.12.121 glib-2.0/gio/gsocketservice.h

```

#define G_TYPE_SOCKET_SERVICE (g_socket_service_get_type ())
#define G_SOCKET_SERVICE_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_SOCKET_SERVICE, GSocketServiceClass))
#define G_IS_SOCKET_SERVICE_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_SOCKET_SERVICE))
#define G_SOCKET_SERVICE(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_SOCKET_SERVICE, GSocketService))
#define G_IS_SOCKET_SERVICE(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_SOCKET_SERVICE))
#define G_SOCKET_SERVICE_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_SOCKET_SERVICE,
GSocketServiceClass))

typedef struct _GSocketServicePrivate GSocketServicePrivate;

```

```

typedef struct _GSocketServiceClass {
    GSocketListenerClass parent_class;
    gboolean(*incoming) (GSocketService * service,
                        GSocketConnection * connection,
                        GObject * source_object);

    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
} GSocketServiceClass;
struct _GSocketService {
    GSocketListener parent_instance;
    GSocketServicePrivate *priv;
};
extern GType g_socket_service_get_type(void);
extern gboolean g_socket_service_is_active(GSocketService *
service);
extern GSocketService *g_socket_service_new(void);
extern void g_socket_service_start(GSocketService * service);
extern void g_socket_service_stop(GSocketService * service);

```

17.12.122 glib-2.0/gio/gsrvtarget.h

```

#define G_TYPE_SRV_TARGET (g_srv_target_get_type ())

extern GSrvTarget *g_srv_target_copy(GSrvTarget * target);
extern void g_srv_target_free(GSrvTarget * target);
extern const char *g_srv_target_get_hostname(GSrvTarget * target);
extern guint16 g_srv_target_get_port(GSrvTarget * target);
extern guint16 g_srv_target_get_priority(GSrvTarget * target);
extern GType g_srv_target_get_type(void);
extern guint16 g_srv_target_get_weight(GSrvTarget * target);
extern GList *g_srv_target_list_sort(GList * targets);
extern GSrvTarget *g_srv_target_new(const gchar * hostname, guint16
port,
                                guint16 priority, guint16 weight);

```

17.12.123 glib-2.0/gio/gtcpconnection.h

```

#define G_TYPE_TCP_CONNECTION (g_tcp_connection_get_type ())
#define G_TCP_CONNECTION_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_TCP_CONNECTION, GTcpConnectionClass))
#define G_IS_TCP_CONNECTION_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_TCP_CONNECTION))
#define G_TCP_CONNECTION(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_TCP_CONNECTION, GTcpConnection))
#define G_IS_TCP_CONNECTION(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_TCP_CONNECTION))
#define G_TCP_CONNECTION_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_TCP_CONNECTION, GTcpConnectionClass))

typedef struct _GTcpConnectionPrivate GTcpConnectionPrivate;
typedef struct _GTcpConnectionClass {
    GSocketConnectionClass parent_class;
} GTcpConnectionClass;
struct _GTcpConnection {
    GSocketConnection parent_instance;
    GTcpConnectionPrivate *priv;
};

```

```

extern                                     gboolean
g_tcp_connection_get_graceful_disconnect(GTcpConnection *
                                         connection);

extern GType g_tcp_connection_get_type(void);
extern                                     void
g_tcp_connection_set_graceful_disconnect(GTcpConnection *
                                         connection,
                                         gboolean
                                         graceful_disconnect);

```

17.12.124 glib-2.0/gio/gtcpwrapperconnection.h

```

#define                                     G_TYPE_TCP_WRAPPER_CONNECTION
(g_tcp_wrapper_connection_get_type ())
#define                                     G_TCP_WRAPPER_CONNECTION_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_TCP_WRAPPER_CONNECTION,
GTcpWrapperConnectionClass))
#define                                     G_IS_TCP_WRAPPER_CONNECTION_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_TCP_WRAPPER_CONNECTION))
#define                                     G_TCP_WRAPPER_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_TCP_WRAPPER_CONNECTION,
GTcpWrapperConnection))
#define                                     G_IS_TCP_WRAPPER_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_TCP_WRAPPER_CONNECTION))
#define                                     G_TCP_WRAPPER_CONNECTION_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_TCP_WRAPPER_CONNECTION,
GTcpWrapperConnectionClass))

typedef struct _GTcpWrapperConnectionPrivate
GTcpWrapperConnectionPrivate;
typedef struct _GTcpWrapperConnectionClass {
    GTcpConnectionClass parent_class;
} GTcpWrapperConnectionClass;
struct _GTcpWrapperConnection {
    GTcpConnection parent_instance;
    GTcpWrapperConnectionPrivate *priv;
};
extern GIOStream

*g_tcp_wrapper_connection_get_base_io_stream(GTcpWrapperConnectio
n *
                                         conn);
extern GType g_tcp_wrapper_connection_get_type(void);
extern GSocketConnection *g_tcp_wrapper_connection_new(GIOStream *
base_io_stream,
GSocket * socket);

```

17.12.125 glib-2.0/gio/gthemedicon.h

```

#define G_TYPE_THEMED_ICON (g_themed_icon_get_type ())
#define G_THEMED_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_THEMED_ICON, GThemedIconClass))
#define G_IS_THEMED_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_THEMED_ICON))
#define G_THEMED_ICON(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_THEMED_ICON, GThemedIcon))
#define G_IS_THEMED_ICON(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_THEMED_ICON))
#define G_THEMED_ICON_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_THEMED_ICON, GThemedIconClass))

typedef struct _GThemedIconClass GThemedIconClass;

```

```

extern void g_themed_icon_append_name(GThemedIcon * icon,
                                     const char *iconname);
extern const char *const *g_themed_icon_get_names(GThemedIcon *
icon);
extern GType g_themed_icon_get_type(void);
extern GIcon *g_themed_icon_new(const char *iconname);
extern GIcon *g_themed_icon_new_from_names(char **iconnames, int
len);
extern GIcon *g_themed_icon_new_with_default_fallbacks(const char
*iconname);
extern void g_themed_icon_prepend_name(GThemedIcon * icon,
                                     const char *iconname);

```

17.12.126 glib-2.0/gio/gthreadedsocketservice.h

```

#define G_TYPE_THREADED_SOCKET_SERVICE
(g_threaded_socket_service_get_type ())
#define G_THREADED_SOCKET_SERVICE_CLASS(class)
(G_TYPE_CHECK_CLASS_CAST ((class), G_TYPE_THREADED_SOCKET_SERVICE,
GThreadedSocketServiceClass))
#define G_IS_THREADED_SOCKET_SERVICE(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_THREADED_SOCKET_SERVICE))
#define G_THREADED_SOCKET_SERVICE(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst),
G_TYPE_THREADED_SOCKET_SERVICE, GThreadedSocketService))
#define G_IS_THREADED_SOCKET_SERVICE(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst),
G_TYPE_THREADED_SOCKET_SERVICE))
#define G_THREADED_SOCKET_SERVICE_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_THREADED_SOCKET_SERVICE,
GThreadedSocketServiceClass))

typedef struct _GThreadedSocketServicePrivate
GThreadedSocketServicePrivate;
typedef struct _GThreadedSocketServiceClass {
GSocketServiceClass parent_class;
gboolean(*run) (GThreadedSocketService * service,
GSocketConnection * connection,
GObject * source_object);
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
void (*_g_reserved5) (void);
} GThreadedSocketServiceClass;
struct _GThreadedSocketService {
GSocketService parent_instance;
GThreadedSocketServicePrivate *priv;
};
extern GType g_threaded_socket_service_get_type(void);
extern GSocketService *g_threaded_socket_service_new(int
max_threads);

```

17.12.127 glib-2.0/gio/gtlsbackend.h

```

#define G_TYPE_TLS_BACKEND (g_tls_backend_get_type ())
#define G_TLS_BACKEND(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
G_TYPE_TLS_BACKEND, GTlsBackend))
#define G_IS_TLS_BACKEND(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
G_TYPE_TLS_BACKEND))
#define G_TLS_BACKEND_GET_INTERFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), G_TYPE_TLS_BACKEND,
GTlsBackendInterface))

```

```

#define G_TLS_BACKEND_EXTENSION_POINT_NAME    "gio-tls-backend"

typedef struct _GTlsBackend GTlsBackend;
typedef struct _GTlsBackendInterface {
    GTypeInterface g_iface;
    gboolean(*supports_tls) (GTlsBackend * backend);
    GType(*get_certificate_type) (void);
    GType(*get_client_connection_type) (void);
    GType(*get_server_connection_type) (void);
    GType(*get_file_database_type) (void);
    GTlsDatabase *(*get_default_database) (GTlsBackend * backend);
} GTlsBackendInterface;
extern GType g_tls_backend_get_certificate_type(GTlsBackend *
backend);
extern GType g_tls_backend_get_client_connection_type(GTlsBackend
*
backend);
extern GTlsBackend *g_tls_backend_get_default(void);
extern GTlsDatabase g_tls_backend_get_default_database(GTlsBackend *
backend);
extern GType g_tls_backend_get_file_database_type(GTlsBackend *
backend);
extern GType g_tls_backend_get_server_connection_type(GTlsBackend
*
backend);
extern GType g_tls_backend_get_type(void);
extern gboolean g_tls_backend_supports_tls(GTlsBackend * backend);

```

17.12.128 glib-2.0/gio/gtlscertificate.h

```

#define G_TYPE_TLS_CERTIFICATE (g_tls_certificate_get_type ())
#define G_TLS_CERTIFICATE_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_TLS_CERTIFICATE, GTlsCertificateClass))
#define G_IS_TLS_CERTIFICATE_CLASS(class)
(G_TYPE_CHECK_CLASS_TYPE ((class), G_TYPE_TLS_CERTIFICATE))
#define G_TLS_CERTIFICATE(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_TLS_CERTIFICATE, GTlsCertificate))
#define G_IS_TLS_CERTIFICATE(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_TLS_CERTIFICATE))
#define G_TLS_CERTIFICATE_GET_CLASS(inst)
(G_TYPE_INSTANCE_GET_CLASS ((inst), G_TYPE_TLS_CERTIFICATE,
GTlsCertificateClass))

typedef struct _GTlsCertificateClass {
    GObjectClass parent_class;
    GTlsCertificateFlags(*verify) (GTlsCertificate * cert,
GSocketConnectable * identity,
GTlsCertificate * trusted_ca);
    gpointer padding[8];
} GTlsCertificateClass;
typedef struct _GTlsCertificatePrivate GTlsCertificatePrivate;
extern GTlsCertificate g_tls_certificate_get_issuer(GTlsCertificate *
cert);
extern GType g_tls_certificate_get_type(void);
extern GList *g_tls_certificate_list_new_from_file(const gchar *
file,
GError * *error);
extern GTlsCertificate *g_tls_certificate_new_from_file(const
gchar * file,
GError * *error);
extern GTlsCertificate *g_tls_certificate_new_from_files(const
gchar *
cert_file,

```

```

const gchar *
key_file,
GError * *error);
extern GTlsCertificate *g_tls_certificate_new_from_pem(const gchar
* data,
gssize length,
GError * *error);
extern
g_tls_certificate_verify(GTlsCertificate *
cert,
GSocketConnectable *
identity,
GTlsCertificate *
trusted_ca);

```

17.12.129 glib-2.0/gio/gtlsclientconnection.h

```

#define G_TYPE_TLS_CLIENT_CONNECTION
(g_tls_client_connection_get_type ())
#define G_TLS_CLIENT_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_TLS_CLIENT_CONNECTION,
GTlsClientConnection))
#define G_IS_TLS_CLIENT_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_TLS_CLIENT_CONNECTION))
#define G_TLS_CLIENT_CONNECTION_GET_INTERFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst),
G_TYPE_TLS_CLIENT_CONNECTION, GTlsClientConnectionInterface))

typedef struct _GTlsClientConnectionInterface {
GTypeInterface g_iface;
} GTlsClientConnectionInterface;
extern
GList
*g_tls_client_connection_get_accepted_cas(GTlsClientConnection
* conn);
extern GSocketConnectable
*g_tls_client_connection_get_server_identity(GTlsClientConnection
*
conn);
extern GType g_tls_client_connection_get_type(void);
extern
gboolean
g_tls_client_connection_get_use_ssl3(GTlsClientConnection *
conn);
extern GTlsCertificateFlags
g_tls_client_connection_get_validation_flags(GTlsClientConnection
* conn);
extern GIOStream *g_tls_client_connection_new(GIOStream *
base_io_stream,
GSocketConnectable *
server_identity,
GError * *error);
extern void
g_tls_client_connection_set_server_identity(GTlsClientConnection *
conn,
GSocketConnectable *
identity);
extern
void
g_tls_client_connection_set_use_ssl3(GTlsClientConnection *
conn, gboolean use_ssl3);
extern void
g_tls_client_connection_set_validation_flags(GTlsClientConnection
* conn,
GTlsCertificateFlags flags);

```

17.12.130 glib-2.0/gio/gtlsconnection.h

```

#define G_TYPE_TLS_CONNECTION (g_tls_connection_get_type ())
#define G_TLS_ERROR (g_tls_error_quark ())
#define G_TLS_CONNECTION_CLASS(class) (G_TYPE_CHECK_CLASS_CAST((class), G_TYPE_TLS_CONNECTION, GTlsConnectionClass))
#define G_IS_TLS_CONNECTION_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE((class), G_TYPE_TLS_CONNECTION))
#define G_TLS_CONNECTION(inst) (G_TYPE_CHECK_INSTANCE_CAST((inst), G_TYPE_TLS_CONNECTION, GTlsConnection))
#define G_IS_TLS_CONNECTION(inst) (G_TYPE_CHECK_INSTANCE_TYPE((inst), G_TYPE_TLS_CONNECTION))
#define G_TLS_CONNECTION_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS((inst), G_TYPE_TLS_CONNECTION, GTlsConnectionClass))

typedef struct _GTlsConnectionClass {
    GIOStreamClass parent_class;
    gboolean(*accept_certificate)(GTlsConnection * connection,
                                GTlsCertificate * peer_cert,
                                GTlsCertificateFlags errors);
    gboolean(*handshake)(GTlsConnection * conn,
                        Gancellable * cancellable, GError * *error);
    void(*handshake_async)(GTlsConnection * conn, int io_priority,
                          Gancellable * cancellable,
                          GAsyncReadyCallback callback,
                          gpointer user_data);
    gboolean(*handshake_finish)(GTlsConnection * conn,
                              GAsyncResult * result, GError *
                              *error);
    gpointer padding[8];
} GTlsConnectionClass;
typedef struct _GTlsConnectionPrivate GTlsConnectionPrivate;
struct _GTlsConnection {
    GIOStream parent_instance;
    GTlsConnectionPrivate *priv;
};
extern gboolean
g_tls_connection_emit_accept_certificate(GTlsConnection *
                                        conn,
                                        GTlsCertificate *
                                        peer_cert,

                                        GTlsCertificateFlags
                                        errors);
extern GTlsCertificate
*g_tls_connection_get_certificate(GTlsConnection *
                                 conn);
extern GTlsDatabase *g_tls_connection_get_database(GTlsConnection
* conn);
extern GTlsInteraction
*g_tls_connection_get_interaction(GTlsConnection *
                                 conn);
extern GTlsCertificate
*g_tls_connection_get_peer_certificate(GTlsConnection * conn);
extern GTlsCertificateFlags
g_tls_connection_get_peer_certificate_errors(GTlsConnection *
conn);
extern GTlsRehandshakeMode
g_tls_connection_get_rehandshake_mode(GTlsConnection * conn);
extern gboolean
g_tls_connection_get_require_close_notify(GTlsConnection *
                                           conn);
extern GType g_tls_connection_get_type(void);

```

```

extern gboolean g_tls_connection_get_use_system_certdb(GTlsConnection *
                                                         conn);
extern gboolean g_tls_connection_handshake(GTlsConnection * conn,
                                           Gancellable * cancellable,
                                           GError * *error);
extern void g_tls_connection_handshake_async(GTlsConnection * conn,
                                             int io_priority,
                                             Gancellable * cancellable,
                                             GAsyncReadyCallback callback,
                                             gpointer user_data);
extern gboolean g_tls_connection_handshake_finish(GTlsConnection *
conn,
                                                  GAsyncResult * result,
                                                  GError * *error);
extern void g_tls_connection_set_certificate(GTlsConnection * conn,
                                           GTlsCertificate *
                                           certificate);
extern void g_tls_connection_set_database(GTlsConnection * conn,
                                           GTlsDatabase * database);
extern void g_tls_connection_set_interaction(GTlsConnection * conn,
                                           GTlsInteraction *
                                           interaction);
extern void g_tls_connection_set_rehandshake_mode(GTlsConnection *
conn,
                                                  GTlsRehandshakeMode
                                                  mode);
extern void g_tls_connection_set_require_close_notify(GTlsConnection *
conn,
                                                      gboolean
                                                      require_close_notify);
extern void g_tls_connection_set_use_system_certdb(GTlsConnection
* conn,
                                                    gboolean
                                                    use_system_certdb);
extern GQuark g_tls_error_quark(void);

```

17.12.131 glib-2.0/gio/gtlsdatabase.h

```

#define G_TYPE_TLS_DATABASE (g_tls_database_get_type ())
#define G_TLS_DATABASE_CLASS(class) (G_TYPE_CHECK_CLASS_CAST
((class), G_TYPE_TLS_DATABASE, GTlsDatabaseClass))
#define G_IS_TLS_DATABASE_CLASS(class) (G_TYPE_CHECK_CLASS_TYPE
((class), G_TYPE_TLS_DATABASE))
#define G_TLS_DATABASE(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_TLS_DATABASE, GTlsDatabase))
#define G_IS_TLS_DATABASE(inst) (G_TYPE_CHECK_INSTANCE_TYPE
((inst), G_TYPE_TLS_DATABASE))
#define G_TLS_DATABASE_GET_CLASS(inst) (G_TYPE_INSTANCE_GET_CLASS
((inst), G_TYPE_TLS_DATABASE, GTlsDatabaseClass))
#define G_TLS_DATABASE_PURPOSE_AUTHENTICATE_SERVER
"1.3.6.1.5.5.7.3.1"
#define G_TLS_DATABASE_PURPOSE_AUTHENTICATE_CLIENT
"1.3.6.1.5.5.7.3.2"

typedef struct _GTlsDatabaseClass {
    GObjectClass parent_class;
    GTlsCertificateFlags(*verify_chain)(GTlsDatabase * self,
                                        GTlsCertificate * chain,
                                        const gchar * purpose,
                                        GSocketConnectable * identity,
                                        GTlsInteraction * interaction,
                                        GTlsDatabaseVerifyFlags flags,

```



```

        GCancelable * cancellable,
        GError * *error);
void (*verify_chain_async) (GTlsDatabase * self,
        GTlsCertificate * chain,
        const gchar * purpose,
        GSocketConnectable * identity,
        GTlsInteraction * interaction,
        GTlsDatabaseVerifyFlags flags,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
GTlsCertificateFlags (*verify_chain_finish) (GTlsDatabase *
self,
        GAsyncResult * result,
        GError * *error);
gchar *(*create_certificate_handle) (GTlsDatabase * self,
        GTlsCertificate * certificate);
GTlsCertificate (*lookup_certificate_for_handle)
(GTlsDatabase * self,
        const gchar *
        handle,
        GTlsInteraction *
        interaction,
        GTlsDatabaseLookupFlags
        flags,
        GCancelable *
        cancellable,
        GError * *error);
void (*lookup_certificate_for_handle_async) (GTlsDatabase *
self,
        const gchar * handle,
        GTlsInteraction *
        interaction,
        GTlsDatabaseLookupFlags
        flags,
        GCancelable *
        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
GTlsCertificate (*lookup_certificate_for_handle_finish)
(GTlsDatabase *
        * self,
        GAsyncResult
        * result,
        GError *
        *error);
GTlsCertificate (*lookup_certificate_issuer) (GTlsDatabase *
self,
        GTlsCertificate *
        certificate,
        GTlsInteraction *
        interaction,
        GTlsDatabaseLookupFlags
        flags,
        GCancelable *
        cancellable,
        GError * *error);
void (*lookup_certificate_issuer_async) (GTlsDatabase * self,
        GTlsCertificate *
        certificate,
        GTlsInteraction *
        interaction,
        GTlsDatabaseLookupFlags
        flags,

```

```

        GCancelable * cancellable,
        GAsyncResult * result,
        gpointer user_data);
    GTlsCertificate (*lookup_certificate_issuer_finish)
    (GTlsDatabase *
        self,
        GAsyncResult *
        result,
        GError * *error);
    GList *(*lookup_certificates_issued_by) (GTlsDatabase * self,
        GByteArray * issuer_raw_dn,
        GTlsInteraction *
            interaction,
            flags,
            GCancelable * cancellable,
            GError * *error);
    void (*lookup_certificates_issued_by_async) (GTlsDatabase *
        self,
        GByteArray *
            issuer_raw_dn,
            GTlsInteraction *
            interaction,
            GTlsDatabaseLookupFlags
            flags,
            GCancelable *
            cancellable,
            GAsyncResult * result,
            GError * *error);
        GList *(*lookup_certificates_issued_by_finish) (GTlsDatabase *
        self,
        GAsyncResult * result,
        GError * *error);

        gpointer padding[16];
    } GTlsDatabaseClass;
    typedef struct _GTlsDatabasePrivate GTlsDatabasePrivate;
    struct _GTlsDatabase {
        GObject parent_instance;
        GTlsDatabasePrivate *priv;
    };
    extern gchar
    *g_tls_database_create_certificate_handle(GTlsDatabase * self,
        GTlsCertificate *
        certificate);
    extern GType g_tls_database_get_type(void);
    extern GTlsCertificate
    *g_tls_database_lookup_certificate_for_handle(GTlsDatabase *
        self,
        const gchar * handle,
        GTlsInteraction *
        interaction,
        GTlsDatabaseLookupFlags
        flags,
        GCancelable *
        cancellable,
        GError * *error);
    extern void
    g_tls_database_lookup_certificate_for_handle_async(GTlsDatabase
        * self,
        const gchar
        * handle,
        GTlsInteraction
            *
            interaction,

```

```

GTlsDatabaseLookupFlags
                                flags,
                                Gancellable *
                                cancellable,

GAsyncReadyCallback
                                callback,
                                gpointer
                                user_data);

extern GTlsCertificate

*g_tls_database_lookup_certificate_for_handle_finish(GTlsDatabase
*
                                self,
                                GAsyncResult *
                                result,
                                GError * *error);

extern GTlsCertificate
    *g_tls_database_lookup_certificate_issuer(GTlsDatabase * self,
                                GTlsCertificate *
                                certificate,
                                GTlsInteraction *
                                interaction,
                                GTlsDatabaseLookupFlags
                                flags,
                                Gancellable * cancellable,
                                GError * *error);

extern
                                void
g_tls_database_lookup_certificate_issuer_async(GTlsDatabase *
                                self,
                                GTlsCertificate
                                * certificate,
                                GTlsInteraction
                                * interaction,

GTlsDatabaseLookupFlags
                                flags,
                                Gancellable *
                                cancellable,

GAsyncReadyCallback
                                callback,
                                gpointer
                                user_data);

extern GTlsCertificate
    *g_tls_database_lookup_certificate_issuer_finish(GTlsDatabase
* self,
                                GAsyncResult * result,
                                GError * *error);

extern
                                GList
*g_tls_database_lookup_certificates_issued_by(GTlsDatabase *
                                self,
                                GByteArray *
                                issuer_raw_dn,
                                GTlsInteraction
                                * interaction,

GTlsDatabaseLookupFlags
                                flags,
                                Gancellable *
                                cancellable,
                                GError *
                                *error);

```

```

extern void
g_tls_database_lookup_certificates_issued_by_async(GTlsDatabase
                                                    * self,
                                                    GByteArray *
                                                    issuer_raw_dn,

GTlsInteraction
                                                    *
                                                    interaction,

GTlsDatabaseLookupFlags
                                                    flags,
                                                    Gancellable
                                                    *
                                                    cancellable,

GAsyncReadyCallback
                                                    callback,
                                                    gpointer
                                                    user_data);

extern GList
*g_tls_database_lookup_certificates_issued_by_finish(GTlsDatabase
*
                                                    self,
                                                    GAsyncResult *
                                                    result,
                                                    GError * *error);
extern
g_tls_database_verify_chain(GTlsDatabase *
                                                    self,
                                                    GTlsCertificate *
                                                    chain,
                                                    const gchar *
                                                    purpose,
                                                    GSocketConnectable
                                                    * identity,
                                                    GTlsInteraction *
                                                    interaction,

GTlsDatabaseVerifyFlags
                                                    flags,
                                                    Gancellable *
                                                    cancellable,
                                                    GError * *error);
extern void g_tls_database_verify_chain_async(GTlsDatabase * self,
GTlsCertificate * chain,
const gchar * purpose,
GSocketConnectable *
identity,
GTlsInteraction *
interaction,
GTlsDatabaseVerifyFlags
flags,
Gancellable * cancellable,
GAsyncReadyCallback
callback,
gpointer user_data);
extern
g_tls_database_verify_chain_finish(GTlsDatabase
* self,
GAsyncResult
* result,
GError *
*error);

```

17.12.132 glib-2.0/gio/gtlsfiledatabase.h

```

#define G_TYPE_TLS_FILE_DATABASE
(g_tls_file_database_get_type ())
#define G_TLS_FILE_DATABASE(inst) (G_TYPE_CHECK_INSTANCE_CAST
((inst), G_TYPE_TLS_FILE_DATABASE, GTlsFileDatabase))
#define G_IS_TLS_FILE_DATABASE(inst) (G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_TLS_FILE_DATABASE))
#define G_TLS_FILE_DATABASE_GET_INTERFACE(inst) (G_TYPE_INSTANCE_GET_INTERFACE ((inst), G_TYPE_TLS_FILE_DATABASE,
GTlsFileDatabaseInterface))

typedef struct _GTlsFileDatabaseInterface {
    GTypeInterface g_iface;
    gpointer padding[8];
} GTlsFileDatabaseInterface;
extern GType g_tls_file_database_get_type(void);
extern GTlsDatabase *g_tls_file_database_new(const gchar * anchors,
                                              GError * *error);

```

17.12.133 glib-2.0/gio/gtlsinteraction.h

```

#define G_TYPE_TLS_INTERACTION (g_tls_interaction_get_type ())
#define G_TLS_INTERACTION_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_TLS_INTERACTION,
GTlsInteractionClass))
#define G_IS_TLS_INTERACTION_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k), G_TYPE_TLS_INTERACTION))
#define G_TLS_INTERACTION(o) (G_TYPE_CHECK_INSTANCE_CAST ((o), G_TYPE_TLS_INTERACTION, GTlsInteraction))
#define G_IS_TLS_INTERACTION(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o), G_TYPE_TLS_INTERACTION))
#define G_TLS_INTERACTION_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o), G_TYPE_TLS_INTERACTION, GTlsInteractionClass))

typedef struct _GTlsInteractionClass {
    GObjectClass parent_class;
    GTlsInteractionResult(*ask_password) (GTlsInteraction *
interaction,
                                          GTlsPassword * password,
                                          Gancellable * cancellable,
                                          GError * *error);
    void(*ask_password_async) (GTlsInteraction * interaction,
                              GTlsPassword * password,
                              Gancellable * cancellable,
                              GAsyncReadyCallback callback,
                              gpointer user_data);
    GTlsInteractionResult(*ask_password_finish) (GTlsInteraction *
interaction,
                                                GAsyncResult * result,
                                                GError * *error);
    gpointer padding[24];
} GTlsInteractionClass;
typedef struct _GTlsInteractionPrivate GTlsInteractionPrivate;
struct _GTlsInteraction {
    GObject parent_instance;
    GTlsInteractionPrivate *priv;
};
extern GTlsInteractionResult
g_tls_interaction_ask_password(GTlsInteraction
                              * interaction,
                              GTlsPassword *
password,

```

```

                                Gancellable *
                                cancellable,
                                GError *
                                *error);

extern void g_tls_interaction_ask_password_async(GTlsInteraction *
                                                interaction,
                                                GTlsPassword * password,
                                                Gancellable *
                                                cancellable,
                                                GAsyncReadyCallback
                                                callback,
                                                gpointer user_data);

extern GTlsInteractionResult
g_tls_interaction_ask_password_finish(GTlsInteraction *
                                      interaction,
                                      GAsyncResult * result,
                                      GError * *error);

extern GType g_tls_interaction_get_type(void);
extern GTlsInteractionResult
g_tls_interaction_invoke_ask_password(GTlsInteraction *
                                       interaction,
                                       GTlsPassword * password,
                                       Gancellable * cancellable,
                                       GError * *error);

```

17.12.134 glib-2.0/gio/gtlspassword.h

```

#define G_TYPE_TLS_PASSWORD (g_tls_password_get_type ())
#define G_TLS_PASSWORD_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_TYPE_TLS_PASSWORD, GTlsPasswordClass))
#define G_IS_TLS_PASSWORD_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_TLS_PASSWORD))
#define G_TLS_PASSWORD(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_TLS_PASSWORD, GTlsPassword))
#define G_IS_TLS_PASSWORD(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_TLS_PASSWORD))
#define G_TLS_PASSWORD_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_TLS_PASSWORD, GTlsPasswordClass))

typedef struct _GTlsPasswordClass {
    GObjectClass parent_class;
    const gchar *(*get_value) (GTlsPassword * password, gsize *
length);
    void (*set_value) (GTlsPassword * password, gchar * value,
gssize length, GDestroyNotify destroy);
    const gchar *(*get_default_warning) (GTlsPassword * password);
    gpointer padding[4];
} GTlsPasswordClass;

typedef struct _GTlsPasswordPrivate GTlsPasswordPrivate;
struct _GTlsPassword {
    GObject parent_instance;
    GTlsPasswordPrivate *priv;
};

extern const char *g_tls_password_get_description(GTlsPassword *
password);
extern GTlsPasswordFlags g_tls_password_get_flags(GTlsPassword *
password);
extern GType g_tls_password_get_type(void);
extern const unsigned char *g_tls_password_get_value(GTlsPassword
*
                                                password,
                                                gsize * length);
extern const char *g_tls_password_get_warning(GTlsPassword *
password);
extern GTlsPassword *g_tls_password_new(GTlsPasswordFlags flags,

```

```

        const gchar * description);
extern void g_tls_password_set_description(GTlsPassword * password,
        const gchar * description);
extern void g_tls_password_set_flags(GTlsPassword * password,
        GTlsPasswordFlags flags);
extern void g_tls_password_set_value(GTlsPassword * password,
        const gchar * value, gssize length);
extern void g_tls_password_set_value_full(GTlsPassword * password,
        gchar * value, gssize length,
        GDestroyNotify destroy);
extern void g_tls_password_set_warning(GTlsPassword * password,
        const gchar * warning);

```

17.12.135 glib-2.0/gio/gtlsserverconnection.h

```

#define G_TYPE_TLS_SERVER_CONNECTION
(g_tls_server_connection_get_type ())
#define G_TLS_SERVER_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_CAST ((inst), G_TYPE_TLS_SERVER_CONNECTION,
GTlsServerConnection))
#define G_IS_TLS_SERVER_CONNECTION(inst)
(G_TYPE_CHECK_INSTANCE_TYPE ((inst), G_TYPE_TLS_SERVER_CONNECTION))
#define G_TLS_SERVER_CONNECTION_GET_INTERFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst),
G_TYPE_TLS_SERVER_CONNECTION, GTlsServerConnectionInterface))

typedef struct _GTlsServerConnectionInterface {
    GTypeInterface g_iface;
} GTlsServerConnectionInterface;
extern GType g_tls_server_connection_get_type(void);
extern GIOStream *g_tls_server_connection_new(GIOStream *
base_io_stream,
        GTlsCertificate *
        certificate,
        GError * *error);

```

17.12.136 glib-2.0/gio/gvfs.h

```

#define G_VFS_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_VFS,
GVfsClass))
#define G_IS_VFS_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k),
G_TYPE_VFS))
#define G_VFS(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_VFS, GVfs))
#define G_IS_VFS(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_VFS))
#define G_VFS_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o),
G_TYPE_VFS, GVfsClass))
#define G_TYPE_VFS (g_vfs_get_type ())
#define G_VFS_EXTENSION_POINT_NAME "gio-vfs"

typedef struct _GVfsClass {
    GObjectClass parent_class;
    gboolean(*is_active) (GVfs * vfs);
    GFile *(*get_file_for_path) (GVfs * vfs, const char *path);
    GFile *(*get_file_for_uri) (GVfs * vfs, const char *uri);
    const gchar *const *(*get_supported_uri_schemes) (GVfs * vfs);
    GFile *(*parse_name) (GVfs * vfs, const char *parse_name);
    void (*local_file_add_info) (GVfs * vfs, const char *filename,
        guint64 device,
        GFileAttributeMatcher *
attribute_matcher,
        GFileInfo * info,

```

```

        Gancellable * cancellable,
        gpointer * extra_data,
        GDestroyNotify * free_extra_data);
void (*add_writable_namespaces) (GVfs * vfs,
        GFileAttributeInfoList * list);
gboolean(*local_file_set_attributes) (GVfs * vfs,
        const char *filename,
        GFileInfo * info,
        GFileQueryInfoFlags flags,
        Gancellable * cancellable,
        GError * *error);
void (*local_file_removed) (GVfs * vfs, const char *filename);
void (*local_file_moved) (GVfs * vfs, const char *source,
        const char *dest);
void (*_g_reserved1) (void);
void (*_g_reserved2) (void);
void (*_g_reserved3) (void);
void (*_g_reserved4) (void);
void (*_g_reserved5) (void);
void (*_g_reserved6) (void);
void (*_g_reserved7) (void);
} GVfsClass;
struct _GVfs {
    GObject parent_instance;
};
extern GVfs *g_vfs_get_default(void);
extern GFile *g_vfs_get_file_for_path(GVfs * vfs, const char *path);
extern GFile *g_vfs_get_file_for_uri(GVfs * vfs, const char *uri);
extern GVfs *g_vfs_get_local(void);
extern const char *const *g_vfs_get_supported_uri_schemes(GVfs *
    vfs);
extern GType g_vfs_get_type(void);
extern gboolean g_vfs_is_active(GVfs * vfs);
extern GFile *g_vfs_parse_name(GVfs * vfs, const char *parse_name);

```

17.12.137 glib-2.0/gio/gvolume.h

```

#define G_VOLUME(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
    G_TYPE_VOLUME, GVolum))
#define G_IS_VOLUME(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
    G_TYPE_VOLUME))
#define G_VOLUME_GET_IFACE(obj) (G_TYPE_INSTANCE_GET_INTERFACE
    ((obj), G_TYPE_VOLUME, GVolumIface))
#define G_TYPE_VOLUME (g_volume_get_type ())
#define G_VOLUME_IDENTIFIER_KIND_CLASS "class"
#define G_VOLUME_IDENTIFIER_KIND_HAL_UDI "hal-udi"
#define G_VOLUME_IDENTIFIER_KIND_LABEL "label"
#define G_VOLUME_IDENTIFIER_KIND_NFS_MOUNT "nfs-mount"
#define G_VOLUME_IDENTIFIER_KIND_UNIX_DEVICE "unix-device"
#define G_VOLUME_IDENTIFIER_KIND_UUID "uuid"

typedef struct _GVolumIface {
    GTypeInterface g_iface;
    void (*changed) (GVolum * volume);
    void (*removed) (GVolum * volume);
    char *(*get_name) (GVolum * volume);
    GIcon *(*get_icon) (GVolum * volume);
    char *(*get_uuid) (GVolum * volume);
    GDrive *(*get_drive) (GVolum * volume);
    GMount *(*get_mount) (GVolum * volume);
    gboolean(*can_mount) (GVolum * volume);
    gboolean(*can_eject) (GVolum * volume);
    void (*mount_fn) (GVolum * volume, GMountMountFlags flags,
        GMountOperation * mount_operation,
        Gancellable * cancellable,

```



```

        GAsyncReadyCallback callback, gpointer
user_data);
    gboolean(*mount_finish) (GVolume * volume, GAsyncResult *
result,
        GError * *error);
    void (*eject) (GVolume * volume, GMountUnmountFlags flags,
        GCancellable * cancellable,
        GAsyncReadyCallback callback, gpointer user_data);
    gboolean(*eject_finish) (GVolume * volume, GAsyncResult *
result,
        GError * *error);
    char *(*get_identifier) (GVolume * volume, const char *kind);
    char **(*enumerate_identifiers) (GVolume * volume);
    gboolean(*should_automount) (GVolume * volume);
    GFile *(*get_activation_root) (GVolume * volume);
    void (*eject_with_operation) (GVolume * volume,
        GMountUnmountFlags flags,
        GMountOperation * mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
    gboolean(*eject_with_operation_finish) (GVolume * volume,
        GAsyncResult * result,
        GError * *error);

    const char *(*get_sort_key) (GVolume * volume);
} GVolumeIface;
extern gboolean g_volume_can_eject(GVolume * volume);
extern gboolean g_volume_can_mount(GVolume * volume);
extern void g_volume_eject(GVolume * volume, GMountUnmountFlags
flags,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_volume_eject_finish(GVolume * volume,
        GAsyncResult * result,
        GError * *error);
extern void g_volume_eject_with_operation(GVolume * volume,
        GMountUnmountFlags flags,
        GMountOperation *
mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_volume_eject_with_operation_finish(GVolume *
volume,
        GAsyncResult * result,
        GError * *error);
extern void g_volume_enumerate_identifiers(GVolume * volume);
extern GFile *g_volume_get_activation_root(GVolume * volume);
extern GDrive *g_volume_get_drive(GVolume * volume);
extern GIcon *g_volume_get_icon(GVolume * volume);
extern char *g_volume_get_identifier(GVolume * volume, const char
*kind);
extern GMount *g_volume_get_mount(GVolume * volume);
extern char *g_volume_get_name(GVolume * volume);
extern const char *g_volume_get_sort_key(GVolume * volume);
extern GType g_volume_get_type(void);
extern char *g_volume_get_uuid(GVolume * volume);
extern void g_volume_mount(GVolume * volume, GMountMountFlags flags,
        GMountOperation * mount_operation,
        GCancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern gboolean g_volume_mount_finish(GVolume * volume,
        GAsyncResult * result,
        GError * *error);

```

```
extern gboolean g_volume_should_automount(GVolume * volume);
```

17.12.138 glib-2.0/gio/gvolumemonitor.h

```
#define G_VOLUME_MONITOR_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k), G_TYPE_VOLUME_MONITOR, GVolumeMonitorClass))
#define G_IS_VOLUME_MONITOR_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE((k), G_TYPE_VOLUME_MONITOR))
#define G_VOLUME_MONITOR(o) (G_TYPE_CHECK_INSTANCE_CAST((o), G_TYPE_VOLUME_MONITOR, GVolumeMonitor))
#define G_IS_VOLUME_MONITOR(o) (G_TYPE_CHECK_INSTANCE_TYPE((o), G_TYPE_VOLUME_MONITOR))
#define G_VOLUME_MONITOR_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_VOLUME_MONITOR, GVolumeMonitorClass))
#define G_TYPE_VOLUME_MONITOR (g_volume_monitor_get_type())
#define G_VOLUME_MONITOR_EXTENSION_POINT_NAME "gio-volume-monitor"

typedef struct _GVolumeMonitorClass {
    GObjectClass parent_class;
    void (*volume_added) (GVolumeMonitor * volume_monitor,
                          GVolume * volume);
    void (*volume_removed) (GVolumeMonitor * volume_monitor,
                           GVolume * volume);
    void (*volume_changed) (GVolumeMonitor * volume_monitor,
                           GVolume * volume);
    void (*mount_added) (GVolumeMonitor * volume_monitor, GMount * mount);
    void (*mount_removed) (GVolumeMonitor * volume_monitor,
                          GMount * mount);
    void (*mount_pre_unmount) (GVolumeMonitor * volume_monitor,
                              GMount * mount);
    void (*mount_changed) (GVolumeMonitor * volume_monitor,
                          GMount * mount);
    void (*drive_connected) (GVolumeMonitor * volume_monitor,
                             GDrive * drive);
    void (*drive_disconnected) (GVolumeMonitor * volume_monitor,
                                GDrive * drive);
    void (*drive_changed) (GVolumeMonitor * volume_monitor,
                          GDrive * drive);
    gboolean(*is_supported) (void);
    GList *(*get_connected_drives) (GVolumeMonitor * volume_monitor);
    GList *(*get_volumes) (GVolumeMonitor * volume_monitor);
    GList *(*get_mounts) (GVolumeMonitor * volume_monitor);
    GVolume *(*get_volume_for_uuid) (GVolumeMonitor * volume_monitor,
                                     const char *uuid);
    GMount *(*get_mount_for_uuid) (GVolumeMonitor * volume_monitor,
                                  const char *uuid);
    GVolume *(*adopt_orphan_mount) (GMount * mount,
                                    GVolumeMonitor * volume_monitor);
    void (*drive_eject_button) (GVolumeMonitor * volume_monitor,
                               GDrive * drive);
    void (*drive_stop_button) (GVolumeMonitor * volume_monitor,
                              GDrive * drive);
    void (*_g_reserved1) (void);
    void (*_g_reserved2) (void);
    void (*_g_reserved3) (void);
    void (*_g_reserved4) (void);
    void (*_g_reserved5) (void);
    void (*_g_reserved6) (void);
} GVolumeMonitorClass;
struct _GVolumeMonitor {
```

```

    GObject parent_instance;
    gpointer priv;
};
extern GVolumer *g_volumer_monitor_adopt_orphan_mount(GMount *
mount);
extern GVolumerMonitor *g_volumer_monitor_get(void);
extern GList *g_volumer_monitor_get_connected_drives(GVolumerMonitor
*
volume_monitor);
extern GMount *g_volumer_monitor_get_mount_for_uuid(GVolumerMonitor
*
volume_monitor,
const char *uuid);
extern GList *g_volumer_monitor_get_mounts(GVolumerMonitor *
volume_monitor);
extern GType g_volumer_monitor_get_type(void);
extern GVolumer
*g_volumer_monitor_get_volumer_for_uuid(GVolumerMonitor *
volume_monitor,
const char *uuid);
extern GList *g_volumer_monitor_get_volumes(GVolumerMonitor *
volume_monitor);

```

17.12.139 glib-2.0/gio/gzlibcompressor.h

```

#define G_ZLIB_COMPRESSOR_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_ZLIB_COMPRESSOR_CLASS, G_TYPE_ZLIB_COMPRESSOR,
GZlibCompressorClass))
#define G_IS_ZLIB_COMPRESSOR_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_ZLIB_COMPRESSOR))
#define G_ZLIB_COMPRESSOR(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_ZLIB_COMPRESSOR, GZlibCompressor))
#define G_IS_ZLIB_COMPRESSOR(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o),
G_TYPE_ZLIB_COMPRESSOR))
#define G_ZLIB_COMPRESSOR_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS
((o), G_TYPE_ZLIB_COMPRESSOR, GZlibCompressorClass))
#define G_TYPE_ZLIB_COMPRESSOR (g_zlib_compressor_get_type ())

typedef struct _GZlibCompressorClass {
    GObjectClass parent_class;
} GZlibCompressorClass;
extern GFileInfo *g_zlib_compressor_get_file_info(GZlibCompressor
*
compressor);
extern GType g_zlib_compressor_get_type(void);
extern GZlibCompressor
*g_zlib_compressor_new(GZlibCompressorFormat format,
int level);
extern void g_zlib_compressor_set_file_info(GZlibCompressor *
compressor,
GFileInfo * file_info);

```

17.12.140 glib-2.0/gio/gzlibdecompressor.h

```

#define G_ZLIB_DECOMPRESSOR_CLASS(k) (G_TYPE_CHECK_CLASS_CAST((k),
G_ZLIB_DECOMPRESSOR_CLASS, G_TYPE_ZLIB_DECOMPRESSOR,
GZlibDecompressorClass))
#define G_IS_ZLIB_DECOMPRESSOR_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE
((k), G_TYPE_ZLIB_DECOMPRESSOR))
#define G_ZLIB_DECOMPRESSOR(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
G_TYPE_ZLIB_DECOMPRESSOR, GZlibDecompressor))
#define G_IS_ZLIB_DECOMPRESSOR(o) (G_TYPE_CHECK_INSTANCE_TYPE
((o), G_TYPE_ZLIB_DECOMPRESSOR))

```

```

#define G_ZLIB_DECOMPRESSOR_GET_CLASS(o)
(G_TYPE_INSTANCE_GET_CLASS((o), G_TYPE_ZLIB_DECOMPRESSOR,
GZlibDecompressorClass))
#define G_TYPE_ZLIB_DECOMPRESSOR
(g_zlib_decompressor_get_type())

typedef struct _GZlibDecompressorClass {
    GObjectClass parent_class;
} GZlibDecompressorClass;
extern GFileInfo
*g_zlib_decompressor_get_file_info(GZlibDecompressor *
decompressor);
extern GType g_zlib_decompressor_get_type(void);
extern GZlibDecompressor
*g_zlib_decompressor_new(GZlibCompressorFormat
format);

```

17.13 Interface Definitions for libgio-2.0

The interfaces defined on the following pages are included in libgio-2.0 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 17.11 shall behave as described in the referenced base document.

17.14 Interfaces for libatk-1.0

Table 17-101 defines the library name and shared object name for the libatk-1.0 library

Table 17-101 libatk-1.0 Definition

Library:	libatk-1.0
SONAME:	libatk-1.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[ATK 2.2.0] ATK 2.2.0 Reference Manual
[Gobject 2.32] Gobject 2.32 Reference Manual
[LSB] This Specification

17.14.1 GTK Accessibility Toolkit

17.14.1.1 Interfaces for GTK Accessibility Toolkit

An LSB conforming implementation shall provide the generic functions for GTK Accessibility Toolkit specified in Table 17-102, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-102 libatk-1.0 - GTK Accessibility Toolkit Function Interfaces

atk_action_do_action [ATK 2.2.0]	atk_action_get_description [ATK 2.2.0]
atk_action_get_keybinding [ATK 2.2.0]	atk_action_get_localized_name [ATK 2.2.0]
atk_action_get_n_actions [ATK 2.2.0]	atk_action_get_name [ATK 2.2.0]

atk_action_get_type [Gobject 2.32]	atk_action_set_description [ATK 2.2.0]
atk_add_focus_tracker [ATK 2.2.0]	atk_add_global_event_listener [ATK 2.2.0]
atk_add_key_event_listener [ATK 2.2.0]	atk_attribute_set_free [ATK 2.2.0]
atk_component_add_focus_handler [ATK 2.2.0]	atk_component_contains [ATK 2.2.0]
atk_component_get_alpha [ATK 2.2.0]	atk_component_get_extents [ATK 2.2.0]
atk_component_get_layer [ATK 2.2.0]	atk_component_get_mdi_zorder [ATK 2.2.0]
atk_component_get_position [ATK 2.2.0]	atk_component_get_size [ATK 2.2.0]
atk_component_get_type [Gobject 2.32]	atk_component_grab_focus [ATK 2.2.0]
atk_component_ref_accessible_at_point [ATK 2.2.0]	atk_component_remove_focus_handler [ATK 2.2.0]
atk_component_set_extents [ATK 2.2.0]	atk_component_set_position [ATK 2.2.0]
atk_component_set_size [ATK 2.2.0]	atk_coord_type_get_type [Gobject 2.32]
atk_document_get_attribute_value [ATK 2.2.0]	atk_document_get_attributes [ATK 2.2.0]
atk_document_get_document [ATK 2.2.0]	atk_document_get_document_type [ATK 2.2.0]
atk_document_get_locale [ATK 2.2.0]	atk_document_get_type [Gobject 2.32]
atk_document_set_attribute_value [ATK 2.2.0]	atk_editable_text_copy_text [ATK 2.2.0]
atk_editable_text_cut_text [ATK 2.2.0]	atk_editable_text_delete_text [ATK 2.2.0]
atk_editable_text_get_type [Gobject 2.32]	atk_editable_text_insert_text [ATK 2.2.0]
atk_editable_text_paste_text [ATK 2.2.0]	atk_editable_text_set_run_attributes [ATK 2.2.0]
atk_editable_text_set_text_contents [ATK 2.2.0]	atk_focus_tracker_init [ATK 2.2.0]
atk_focus_tracker_notify [ATK 2.2.0]	atk_get_default_registry [ATK 2.2.0]
atk_get_focus_object [ATK 2.2.0]	atk_get_root [ATK 2.2.0]
atk_get_toolkit_name [ATK 2.2.0]	atk_get_toolkit_version [ATK 2.2.0]

atk_get_version [ATK 2.2.0]	atk_gobject_accessible_for_object [ATK 2.2.0]
atk_gobject_accessible_get_object [ATK 2.2.0]	atk_gobject_accessible_get_type [Gobject 2.32]
atk_hyperlink_get_end_index [ATK 2.2.0]	atk_hyperlink_get_n_anchors [ATK 2.2.0]
atk_hyperlink_get_object [ATK 2.2.0]	atk_hyperlink_get_start_index [ATK 2.2.0]
atk_hyperlink_get_type [Gobject 2.32]	atk_hyperlink_get_uri [ATK 2.2.0]
atk_hyperlink_impl_get_hyperlink [ATK 2.2.0]	atk_hyperlink_impl_get_type [Gobject 2.32]
atk_hyperlink_is_inline [ATK 2.2.0]	atk_hyperlink_is_selected_link [ATK 2.2.0]
atk_hyperlink_is_valid [ATK 2.2.0]	atk_hyperlink_state_flags_get_type [Gobject 2.32]
atk_hypertext_get_link [ATK 2.2.0]	atk_hypertext_get_link_index [ATK 2.2.0]
atk_hypertext_get_n_links [ATK 2.2.0]	atk_hypertext_get_type [Gobject 2.32]
atk_image_get_image_description [ATK 2.2.0]	atk_image_get_image_locale [ATK 2.2.0]
atk_image_get_image_position [ATK 2.2.0]	atk_image_get_image_size [ATK 2.2.0]
atk_image_get_type [Gobject 2.32]	atk_image_set_image_description [ATK 2.2.0]
atk_implementor_get_type [Gobject 2.32]	atk_implementor_ref_accessible [ATK 2.2.0]
atk_key_event_type_get_type [Gobject 2.32]	atk_layer_get_type [Gobject 2.32]
atk_misc_get_instance [LSB]	atk_misc_get_type [Gobject 2.32]
atk_misc_threads_enter [LSB]	atk_misc_threads_leave [LSB]
atk_no_op_object_factory_get_type [Gobject 2.32]	atk_no_op_object_factory_new [ATK 2.2.0]
atk_no_op_object_get_type [Gobject 2.32]	atk_no_op_object_new [ATK 2.2.0]
atk_object_add_relationship [ATK 2.2.0]	atk_object_connect_property_change_handler [ATK 2.2.0]
atk_object_factory_create_accessible [ATK 2.2.0]	atk_object_factory_get_accessible_type [ATK 2.2.0]
atk_object_factory_get_type [Gobject 2.32]	atk_object_factory_invalidate [ATK 2.2.0]

atk_object_get_attributes [ATK 2.2.0]	atk_object_get_description [ATK 2.2.0]
atk_object_get_index_in_parent [ATK 2.2.0]	atk_object_get_n_accessible_children [ATK 2.2.0]
atk_object_get_name [ATK 2.2.0]	atk_object_get_parent [ATK 2.2.0]
atk_object_get_role [ATK 2.2.0]	atk_object_get_type [Gobject 2.32]
atk_object_initialize [ATK 2.2.0]	atk_object_notify_state_change [ATK 2.2.0]
atk_object_ref_accessible_child [ATK 2.2.0]	atk_object_ref_relation_set [ATK 2.2.0]
atk_object_ref_state_set [ATK 2.2.0]	atk_object_remove_property_change_handler [ATK 2.2.0]
atk_object_remove_relationship [ATK 2.2.0]	atk_object_set_description [ATK 2.2.0]
atk_object_set_name [ATK 2.2.0]	atk_object_set_parent [ATK 2.2.0]
atk_object_set_role [ATK 2.2.0]	atk_plug_get_id [ATK 2.2.0]
atk_plug_get_type [Gobject 2.32]	atk_plug_new [ATK 2.2.0]
atk_rectangle_get_type [Gobject 2.32]	atk_registry_get_factory [ATK 2.2.0]
atk_registry_get_factory_type [ATK 2.2.0]	atk_registry_get_type [Gobject 2.32]
atk_registry_set_factory_type [ATK 2.2.0]	atk_relation_add_target [ATK 2.2.0]
atk_relation_get_relation_type [ATK 2.2.0]	atk_relation_get_target [ATK 2.2.0]
atk_relation_get_type [Gobject 2.32]	atk_relation_new [ATK 2.2.0]
atk_relation_remove_target [ATK 2.2.0]	atk_relation_set_add [ATK 2.2.0]
atk_relation_set_add_relation_by_type [ATK 2.2.0]	atk_relation_set_contains [ATK 2.2.0]
atk_relation_set_get_n_relations [ATK 2.2.0]	atk_relation_set_get_relation [ATK 2.2.0]
atk_relation_set_get_relation_by_type [ATK 2.2.0]	atk_relation_set_get_type [Gobject 2.32]
atk_relation_set_new [ATK 2.2.0]	atk_relation_set_remove [ATK 2.2.0]
atk_relation_type_for_name [ATK 2.2.0]	atk_relation_type_get_name [ATK 2.2.0]
atk_relation_type_get_type [Gobject 2.32]	atk_relation_type_register [ATK 2.2.0]
atk_remove_focus_tracker [ATK 2.2.0]	atk_remove_global_event_listener [ATK 2.2.0]

atk_remove_key_event_listener [ATK 2.2.0]	atk_role_for_name [ATK 2.2.0]
atk_role_get_localized_name [ATK 2.2.0]	atk_role_get_name [ATK 2.2.0]
atk_role_get_type [Gobject 2.32]	atk_role_register [ATK 2.2.0]
atk_selection_add_selection [ATK 2.2.0]	atk_selection_clear_selection [ATK 2.2.0]
atk_selection_get_selection_count [ATK 2.2.0]	atk_selection_get_type [Gobject 2.32]
atk_selection_is_child_selected [ATK 2.2.0]	atk_selection_ref_selection [ATK 2.2.0]
atk_selection_remove_selection [ATK 2.2.0]	atk_selection_select_all_selection [ATK 2.2.0]
atk_socket_embed [ATK 2.2.0]	atk_socket_get_type [Gobject 2.32]
atk_socket_is_occupied [ATK 2.2.0]	atk_socket_new [ATK 2.2.0]
atk_state_set_add_state [ATK 2.2.0]	atk_state_set_add_states [ATK 2.2.0]
atk_state_set_and_sets [ATK 2.2.0]	atk_state_set_clear_states [ATK 2.2.0]
atk_state_set_contains_state [ATK 2.2.0]	atk_state_set_contains_states [ATK 2.2.0]
atk_state_set_get_type [Gobject 2.32]	atk_state_set_is_empty [ATK 2.2.0]
atk_state_set_new [ATK 2.2.0]	atk_state_set_or_sets [ATK 2.2.0]
atk_state_set_remove_state [ATK 2.2.0]	atk_state_set_xor_sets [ATK 2.2.0]
atk_state_type_for_name [ATK 2.2.0]	atk_state_type_get_name [ATK 2.2.0]
atk_state_type_get_type [Gobject 2.32]	atk_state_type_register [ATK 2.2.0]
atk_streamable_content_get_mime_type [ATK 2.2.0]	atk_streamable_content_get_n_mime_types [ATK 2.2.0]
atk_streamable_content_get_stream [ATK 2.2.0]	atk_streamable_content_get_type [Gobject 2.32]
atk_streamable_content_get_uri [ATK 2.2.0]	atk_table_add_column_selection [ATK 2.2.0]
atk_table_add_row_selection [ATK 2.2.0]	atk_table_get_caption [ATK 2.2.0]
atk_table_get_column_at_index [ATK 2.2.0]	atk_table_get_column_description [ATK 2.2.0]
atk_table_get_column_extents_at [ATK 2.2.0]	atk_table_get_column_header [ATK 2.2.0]
atk_table_get_index_at [ATK 2.2.0]	atk_table_get_n_columns [ATK 2.2.0]

atk_table_get_n_rows [ATK 2.2.0]	atk_table_get_row_at_index [ATK 2.2.0]
atk_table_get_row_description [ATK 2.2.0]	atk_table_get_row_extent_at [ATK 2.2.0]
atk_table_get_row_header [ATK 2.2.0]	atk_table_get_selected_columns [ATK 2.2.0]
atk_table_get_selected_rows [ATK 2.2.0]	atk_table_get_summary [ATK 2.2.0]
atk_table_get_type [Gobject 2.32]	atk_table_is_column_selected [ATK 2.2.0]
atk_table_is_row_selected [ATK 2.2.0]	atk_table_is_selected [ATK 2.2.0]
atk_table_ref_at [ATK 2.2.0]	atk_table_remove_column_selection [ATK 2.2.0]
atk_table_remove_row_selection [ATK 2.2.0]	atk_table_set_caption [ATK 2.2.0]
atk_table_set_column_description [ATK 2.2.0]	atk_table_set_column_header [ATK 2.2.0]
atk_table_set_row_description [ATK 2.2.0]	atk_table_set_row_header [ATK 2.2.0]
atk_table_set_summary [ATK 2.2.0]	atk_text_add_selection [ATK 2.2.0]
atk_text_attribute_for_name [ATK 2.2.0]	atk_text_attribute_get_name [ATK 2.2.0]
atk_text_attribute_get_type [Gobject 2.32]	atk_text_attribute_get_value [ATK 2.2.0]
atk_text_attribute_register [ATK 2.2.0]	atk_text_boundary_get_type [Gobject 2.32]
atk_text_clip_type_get_type [Gobject 2.32]	atk_text_free_ranges [ATK 2.2.0]
atk_text_get_bounded_ranges [ATK 2.2.0]	atk_text_get_caret_offset [ATK 2.2.0]
atk_text_get_character_at_offset [ATK 2.2.0]	atk_text_get_character_count [ATK 2.2.0]
atk_text_get_character_extents [ATK 2.2.0]	atk_text_get_default_attributes [ATK 2.2.0]
atk_text_get_n_selections [ATK 2.2.0]	atk_text_get_offset_at_point [ATK 2.2.0]
atk_text_get_range_extents [ATK 2.2.0]	atk_text_get_run_attributes [ATK 2.2.0]
atk_text_get_selection [ATK 2.2.0]	atk_text_get_text [ATK 2.2.0]
atk_text_get_text_after_offset [ATK 2.2.0]	atk_text_get_text_at_offset [ATK 2.2.0]

atk_text_get_text_before_offset [ATK 2.2.0]	atk_text_get_type [Gobject 2.32]
atk_text_range_get_type [Gobject 2.32]	atk_text_remove_selection [ATK 2.2.0]
atk_text_set_caret_offset [ATK 2.2.0]	atk_text_set_selection [ATK 2.2.0]
atk_util_get_type [Gobject 2.32]	atk_value_get_current_value [ATK 2.2.0]
atk_value_get_maximum_value [ATK 2.2.0]	atk_value_get_minimum_increment [ATK 2.2.0]
atk_value_get_minimum_value [ATK 2.2.0]	atk_value_get_type [Gobject 2.32]
atk_value_set_current_value [ATK 2.2.0]	atk_window_get_type [Gobject 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for GTK Accessibility Toolkit specified in Table 17-103, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-103 libatk-1.0 - GTK Accessibility Toolkit Deprecated Function Interfaces

atk_hyberlink_is_selected_link [ATK 2.2.0]	

17.15 Data Definitions for libatk-1.0

This section defines global identifiers and their values that are associated with interfaces contained in libatk-1.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.15.1 atk-1.0/atk/atk-enum-types.h

```
#define ATK_TYPE_HYPERLINK_STATE_FLAGS \
    (atk_hyberlink_state_flags_get_type())
#define ATK_TYPE_COORD_TYPE (atk_coord_type_get_type())
#define ATK_TYPE_KEY_EVENT_TYPE (atk_key_event_type_get_type())
#define ATK_TYPE_LAYER (atk_layer_get_type())
```

```

#define ATK_TYPE_RELATION_TYPE (atk_relation_type_get_type())
#define ATK_TYPE_ROLE (atk_role_get_type())
#define ATK_TYPE_STATE_TYPE (atk_state_type_get_type())
#define ATK_TYPE_TEXT_ATTRIBUTE (atk_text_attribute_get_type())
#define ATK_TYPE_TEXT_BOUNDARY (atk_text_boundary_get_type())
#define ATK_TYPE_TEXT_CLIP_TYPE (atk_text_clip_type_get_type())

```

```

extern GType atk_coord_type_get_type(void);
extern GType atk_hyperlink_state_flags_get_type(void);
extern GType atk_key_event_type_get_type(void);
extern GType atk_layer_get_type(void);
extern GType atk_relation_type_get_type(void);
extern GType atk_role_get_type(void);
extern GType atk_state_type_get_type(void);
extern GType atk_text_attribute_get_type(void);
extern GType atk_text_boundary_get_type(void);
extern GType atk_text_clip_type_get_type(void);

```

17.15.2 atk-1.0/atk/atk.h

```

#define _TYPEDEF ATK_ACTION_
#define _TYPEDEF ATK_COMPONENT_
#define _TYPEDEF ATK_DOCUMENT_
#define _TYPEDEF ATK_EDITABLE_TEXT_
#define _TYPEDEF ATK_HYPertext_
#define _TYPEDEF ATK_IMAGE_
#define _TYPEDEF ATK_SELECTION_
#define _TYPEDEF ATK_STREAMABLE_CONTENT_
#define _TYPEDEF ATK_TABLE_
#define _TYPEDEF ATK_TEXT_
#define _TYPEDEF ATK_UTIL_
#define _TYPEDEF ATK_VALUE_
#define ATK_TYPE_NO_OP_OBJECT_FACTORY \
    (atk_no_op_object_factory_get_type ())
#define ATK_GOBJECT_ACCESSIBLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    ATK_TYPE_GOBJECT_ACCESSIBLE, \
    AtkGObjectAccessibleClass))
#define ATK_HYPERLINK_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_HYPERLINK, \
    AtkHyperlinkClass))
#define ATK_NO_OP_OBJECT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_NO_OP_OBJECT, \
    AtkNoOpObjectClass))
#define ATK_NO_OP_OBJECT_FACTORY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    ATK_TYPE_NO_OP_OBJECT_FACTORY, \
    AtkNoOpObjectFactoryClass))
#define ATK_OBJECT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_OBJECT, \
    AtkObjectClass))
#define ATK_OBJECT_FACTORY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_OBJECT_FACTORY, \
    AtkObjectFactoryClass))
#define ATK_REGISTRY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_REGISTRY, \
    AtkRegistryClass))
#define ATK_RELATION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_RELATION, \
    AtkRelationClass))
#define ATK_RELATION_SET_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_RELATION_SET, \
    AtkRelationSetClass))
#define ATK_STATE_SET_CLASS(klass) \

```

```

        (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_STATE_SET, \
        AtkStateSetClass))
#define ATK_UTIL_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), ATK_TYPE_UTIL, \
        AtkUtilClass))
#define ATK_IS_GOBJECT_ACCESSIBLE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), \
        ATK_TYPE_GOBJECT_ACCESSIBLE))
#define ATK_IS_HYPERLINK_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_HYPERLINK))
#define ATK_IS_NO_OP_OBJECT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_NO_OP_OBJECT))
#define ATK_IS_NO_OP_OBJECT_FACTORY_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), \
        ATK_TYPE_NO_OP_OBJECT_FACTORY))
#define ATK_IS_OBJECT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_OBJECT))
#define ATK_IS_OBJECT_FACTORY_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_OBJECT_FACTORY))
#define ATK_IS_REGISTRY_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_REGISTRY))
#define ATK_IS_RELATION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_RELATION))
#define ATK_IS_RELATION_SET_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_RELATION_SET))
#define ATK_IS_STATE_SET_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_STATE_SET))
#define ATK_IS_UTIL_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), ATK_TYPE_UTIL))
#define ATK_GOBJECT_ACCESSIBLE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
        ATK_TYPE_GOBJECT_ACCESSIBLE, \
        AtkGObjectAccessible))
#define ATK_HYPERLINK(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_HYPERLINK, \
        AtkHyperlink))
#define ATK_NO_OP_OBJECT(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_NO_OP_OBJECT, \
        \
        AtkNoOpObject))
#define ATK_NO_OP_OBJECT_FACTORY(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
        ATK_TYPE_NO_OP_OBJECT_FACTORY, \
        AtkNoOpObjectFactory))
#define ATK_OBJECT(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_OBJECT, \
        AtkObject))
#define ATK_OBJECT_FACTORY(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_OBJECT_FACTORY, \
        \
        AtkObjectFactory))
#define ATK_REGISTRY(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_REGISTRY, \
        AtkRegistry))
#define ATK_RELATION(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_RELATION, \
        AtkRelation))
#define ATK_RELATION_SET(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_RELATION_SET, \
        \
        AtkRelationSet))
#define ATK_STATE_SET(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_STATE_SET, \
        AtkStateSet))
#define ATK_IS_GOBJECT_ACCESSIBLE(obj) \

```

```

        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_GOBJECT_ACCESSIBLE))
#define ATK_IS_HYPERLINK(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_HYPERLINK))
#define ATK_IS_NO_OP_OBJECT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_NO_OP_OBJECT))
#define ATK_IS_NO_OP_OBJECT_FACTORY(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_NO_OP_OBJECT_FACTORY))
#define ATK_IS_OBJECT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_OBJECT))
#define ATK_IS_OBJECT_FACTORY(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_OBJECT_FACTORY))
#define ATK_IS_REGISTRY(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_REGISTRY))
#define ATK_IS_RELATION(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_RELATION))
#define ATK_IS_RELATION_SET(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_RELATION_SET))
#define ATK_IS_STATE_SET(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_STATE_SET))
#define ATK_NO_OP_OBJECT_FACTORY_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj),
ATK_TYPE_NO_OP_OBJECT_FACTORY, \
        AtkNoOpObjectFactoryClass))
#define ATK_GOBJECT_ACCESSIBLE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj),
ATK_TYPE_GOBJECT_ACCESSIBLE, \
        AtkGObjectAccessibleClass))
#define ATK_HYPERLINK_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_HYPERLINK, \
        AtkHyperlinkClass))
#define ATK_NO_OP_OBJECT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_NO_OP_OBJECT, \
        AtkNoOpObjectClass))
#define ATK_OBJECT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_OBJECT,
        AtkObjectClass))
#define ATK_OBJECT_FACTORY_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_OBJECT_FACTORY,
        \
        AtkObjectFactoryClass))
#define ATK_REGISTRY_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_REGISTRY, \
        AtkRegistryClass))
#define ATK_RELATION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_RELATION, \
        AtkRelationClass))
#define ATK_RELATION_SET_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_RELATION_SET, \
        AtkRelationSetClass))
#define ATK_STATE_SET_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_STATE_SET, \
        AtkStateSetClass))
#define ATK_UTIL_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), ATK_TYPE_UTIL,
        AtkUtilClass))
#define ATK_ACTION_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_ACTION, \
        AtkActionIface))
#define ATK_COMPONENT_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_COMPONENT,
        \
        AtkComponentIface))
#define ATK_DOCUMENT_GET_IFACE(obj) \

```

```

        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_DOCUMENT, \
        AtkDocumentIface))
#define ATK_EDITABLE_TEXT_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), \
        ATK_TYPE_EDITABLE_TEXT, \
        AtkEditableTextIface))
#define ATK_HYPertext_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_HYPertext, \
        \
        AtkHypertextIface))
#define ATK_IMAGE_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_IMAGE, \
        AtkImageIface))
#define ATK_IMPLEMENTOR_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_IMPLEMENTOR, \
        \
        AtkImplementorIface))
#define ATK_SELECTION_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_SELECTION, \
        \
        AtkSelectionIface))
#define ATK_STREAMABLE_CONTENT_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), \
        ATK_TYPE_STREAMABLE_CONTENT, \
        AtkStreamableContentIface))
#define ATK_TABLE_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_TABLE, \
        AtkTableIface))
#define ATK_TEXT_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_TEXT, \
        AtkTextIface))
#define ATK_VALUE_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_VALUE, \
        AtkValueIface))
#define ATK_ACTION(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_ACTION, \
        AtkAction)
#define ATK_COMPONENT(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_COMPONENT, \
        AtkComponent)
#define ATK_DOCUMENT(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_DOCUMENT, \
        AtkDocument)
#define ATK_EDITABLE_TEXT(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_EDITABLE_TEXT, \
        \
        AtkEditableText)
#define ATK_HYPertext(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_HYPertext, \
        AtkHypertext)
#define ATK_IMAGE(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_IMAGE, AtkImage)
#define ATK_IMPLEMENTOR(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_IMPLEMENTOR, \
        AtkImplementor)
#define ATK_SELECTION(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_SELECTION, \
        AtkSelection)
#define ATK_STREAMABLE_CONTENT(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), \
        ATK_TYPE_STREAMABLE_CONTENT, \
        AtkStreamableContent)
#define ATK_TABLE(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_TABLE, AtkTable)
#define ATK_TEXT(obj) \
        G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_TEXT, AtkText)

```

```

#define ATK_UTIL(obj) \
    G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_UTIL, AtkUtil)
#define ATK_VALUE(obj) \
    G_TYPE_CHECK_INSTANCE_CAST ((obj), ATK_TYPE_VALUE, AtkValue)
#define ATK_IS_ACTION(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_ACTION)
#define ATK_IS_COMPONENT(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_COMPONENT)
#define ATK_IS_DOCUMENT(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_DOCUMENT)
#define ATK_IS_EDITABLE_TEXT(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_EDITABLE_TEXT)
#define ATK_IS_HYPERTEXT(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_HYPERTEXT)
#define ATK_IS_IMAGE(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_IMAGE)
#define ATK_IS_IMPLEMENTOR(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_IMPLEMENTOR)
#define ATK_IS_SELECTION(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_SELECTION)
#define ATK_IS_STREAMABLE_CONTENT(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_STREAMABLE_CONTENT)
#define ATK_IS_TABLE(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_TABLE)
#define ATK_IS_VALUE(obj) \
    G_TYPE_CHECK_INSTANCE_TYPE ((obj), ATK_TYPE_VALUE)
#define ATK_DEFINE_TYPE_EXTENDED_END() \
    /* following custom code */ \
    } \
    g_once_init_leave (&g_define_type_id__volatile,
g_define_type_id); \
    } \
    return g_define_type_id__volatile; \
} /* closes type_name##_get_type() */
#define
_ATK_DEFINE_TYPE_EXTENDED_BEGIN(TypeName,type_name,TYPE,flags) \
\
static void type_name##_init (TypeName *self);
\
static void type_name##_class_init (TypeName##Class
*klass); \
static gpointer type_name##_parent_class = NULL; \
static void type_name##_class_intern_init (gpointer klass) \
{ \
    type_name##_parent_class = g_type_class_peek_parent (klass); \
    type_name##_class_init ((TypeName##Class*) klass); \
} \
\
GType \
type_name##_get_type (void) \
{ \
    static volatile gsize g_define_type_id__volatile = 0; \
    if (g_once_init_enter (&g_define_type_id__volatile)) \
    { \
        AtkObjectFactory *factory; \
        GType derived_type; \
        GTypeQuery query; \
        GType derived_atk_type; \
        GType g_define_type_id; \
\
        /* Figure out the size of the class and instance we are
        deriving from */ \
        derived_type = g_type_parent (TYPE); \
        factory = atk_registry_get_factory (atk_get_default_registry
()), \

```



```

        derived_type); \
    derived_atk_type = atk_object_factory_get_accessible_type
(factory); \
    g_type_query (derived_atk_type, &query); \
\
    g_define_type_id = \
        g_type_register_static_simple (derived_atk_type, \
            g_intern_static_string
        (#TypeName), \
            query.class_size, \
            (GClassInitFunc)
type_name##_class_intern_init, \
            query.instance_size, \
            (GInstanceInitFunc)
type_name##_init, \
            (GTypeFlags) flags); \
    {
        /* custom code follows */
#define ATK_TYPE_ACTION (atk_action_get_type ())
#define ATK_TYPE_COMPONENT (atk_component_get_type ())
#define ATK_TYPE_DOCUMENT (atk_document_get_type ())
#define ATK_TYPE_EDITABLE_TEXT (atk_editable_text_get_type ())
#define ATK_TYPE_GOBJECT_ACCESSIBLE
(atk_gobject_accessible_get_type ())
#define ATK_TYPE_HYPERLINK (atk_hyperlink_get_type ())
#define ATK_TYPE_HYPERLINK_IMPL (atk_hyperlink_impl_get_type ())
#define ATK_TYPE_HYPertext (atk_hypertext_get_type ())
#define ATK_TYPE_IMAGE (atk_image_get_type ())
#define ATK_TYPE_IMPLEMENTOR (atk_implementation_get_type ())
#define ATK_TYPE_MISC (atk_misc_get_type ())
#define ATK_TYPE_NO_OP_OBJECT (atk_no_op_object_get_type ())
#define ATK_TYPE_OBJECT_FACTORY (atk_object_factory_get_type ())
#define ATK_TYPE_OBJECT (atk_object_get_type ())
#define ATK_TYPE_RECTANGLE (atk_rectangle_get_type ())
#define ATK_TYPE_REGISTRY (atk_registry_get_type ())
#define ATK_TYPE_RELATION (atk_relation_get_type ())
#define ATK_TYPE_RELATION_SET (atk_relation_set_get_type ())
#define ATK_TYPE_SELECTION (atk_selection_get_type ())
#define ATK_TYPE_STATE_SET (atk_state_set_get_type ())
#define ATK_TYPE_STREAMABLE_CONTENT
(atk_streamable_content_get_type ())
#define ATK_TYPE_TABLE (atk_table_get_type ())
#define ATK_TYPE_TEXT (atk_text_get_type ())
#define ATK_TYPE_UTIL (atk_util_get_type ())
#define ATK_TYPE_VALUE (atk_value_get_type ())
#define ATK_MISC_CLASS(klass) (G_TYPE_CHECK_CLASS_CAST ((klass),
ATK_TYPE_MISC, AtkMiscClass))
#define ATK_IS_MISC_CLASS(klass) (G_TYPE_CHECK_CLASS_TYPE
((klass), ATK_TYPE_MISC))
#define ATK_MISC_GET_CLASS(obj) (G_TYPE_INSTANCE_GET_CLASS ((obj),
ATK_TYPE_MISC, AtkMiscClass))
#define ATK_DEFINE_TYPE(TN,t_n,T_P) ATK_DEFINE_TYPE_EXTENDED
(TN, t_n, T_P, 0, {})
#define ATK_DEFINE_ABSTRACT_TYPE(TN,t_n,T_P)
ATK_DEFINE_TYPE_EXTENDED (TN, t_n, T_P, G_TYPE_FLAG_ABSTRACT, {})
#define ATK_VAR extern
#define ATK_HYPERLINK_IMPL(obj) G_TYPE_CHECK_INSTANCE_CAST ((obj),
ATK_TYPE_HYPERLINK_IMPL, AtkHyperlinkImpl)
#define ATK_MISC(obj) G_TYPE_CHECK_INSTANCE_CAST ((obj),
ATK_TYPE_MISC, AtkMisc)
#define ATK_IS_HYPERLINK_IMPL(obj) G_TYPE_CHECK_INSTANCE_TYPE
((obj), ATK_TYPE_HYPERLINK_IMPL)
#define ATK_IS_MISC(obj) G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_MISC)
#define ATK_IS_TEXT(obj) G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_TEXT)

```



```

#define ATK_IS_UTIL(obj)          G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_UTIL)
#define                          ATK_HYPERLINK_IMPL_GET_IFACE(obj)
G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_HYPERLINK_IMPL,
AtkHyperlinkImplIface)
#define                          ATK_DEFINE_TYPE_WITH_CODE(TN,t_n,T_P,C_)
_ATK_DEFINE_TYPE_EXTENDED_BEGIN (TN, t_n, T_P, 0) {_C_;}
_ATK_DEFINE_TYPE_EXTENDED_END()
#define                          ATK_DEFINE_ABSTRACT_TYPE_WITH_CODE(TN,t_n,T_P,C_)
_ATK_DEFINE_TYPE_EXTENDED_BEGIN (TN, t_n, T_P,
G_TYPE_FLAG_ABSTRACT) {_C_;} _ATK_DEFINE_TYPE_EXTENDED_END()
#define                          ATK_DEFINE_TYPE_EXTENDED(TN,t_n,T_P,f_,C_)
_ATK_DEFINE_TYPE_EXTENDED_BEGIN (TN, t_n, T_P, f_) {_C_;}
_ATK_DEFINE_TYPE_EXTENDED_END()

typedef struct _AtkText AtkText;
typedef struct _AtkObject {
    GObject parent;
    gchar *description;
    gchar *name;
    AtkObject *accessible_parent;
    AtkRole role;
    AtkRelationSet *relation_set;
    AtkLayer layer;
} AtkObject;
typedef enum {
    ATK_ROLE_INVALID = 0,
    ATK_ROLE_ACCEL_LABEL = 1,
    ATK_ROLE_ALERT = 2,
    ATK_ROLE_ANIMATION = 3,
    ATK_ROLE_ARROW = 4,
    ATK_ROLE_CALENDAR = 5,
    ATK_ROLE_CANVAS = 6,
    ATK_ROLE_CHECK_BOX = 7,
    ATK_ROLE_CHECK_MENU_ITEM = 8,
    ATK_ROLE_COLOR_CHOOSER = 9,
    ATK_ROLE_COLUMN_HEADER = 10,
    ATK_ROLE_COMBO_BOX = 11,
    ATK_ROLE_DATE_EDITOR = 12,
    ATK_ROLE_DESKTOP_ICON = 13,
    ATK_ROLE_DESKTOP_FRAME = 14,
    ATK_ROLE_DIAL = 15,
    ATK_ROLE_DIALOG = 16,
    ATK_ROLE_DIRECTORY_PANE = 17,
    ATK_ROLE_DRAWING_AREA = 18,
    ATK_ROLE_FILE_CHOOSER = 19,
    ATK_ROLE_FILLER = 20,
    ATK_ROLE_FONT_CHOOSER = 21,
    ATK_ROLE_FRAME = 22,
    ATK_ROLE_GLASS_PANE = 23,
    ATK_ROLE_HTML_CONTAINER = 24,
    ATK_ROLE_ICON = 25,
    ATK_ROLE_IMAGE = 26,
    ATK_ROLE_INTERNAL_FRAME = 27,
    ATK_ROLE_LABEL = 28,
    ATK_ROLE_LAYERED_PANE = 29,
    ATK_ROLE_LIST = 30,
    ATK_ROLE_LIST_ITEM = 31,
    ATK_ROLE_MENU = 32,
    ATK_ROLE_MENU_BAR = 33,
    ATK_ROLE_MENU_ITEM = 34,
    ATK_ROLE_OPTION_PANE = 35,
    ATK_ROLE_PAGE_TAB = 36,
    ATK_ROLE_PAGE_TAB_LIST = 37,
    ATK_ROLE_PANEL = 38,
    ATK_ROLE_PASSWORD_TEXT = 39,

```

```

    ATK_ROLE_POPUP_MENU = 40,
    ATK_ROLE_PROGRESS_BAR = 41,
    ATK_ROLE_PUSH_BUTTON = 42,
    ATK_ROLE_RADIO_BUTTON = 43,
    ATK_ROLE_RADIO_MENU_ITEM = 44,
    ATK_ROLE_ROOT_PANE = 45,
    ATK_ROLE_ROW_HEADER = 46,
    ATK_ROLE_SCROLL_BAR = 47,
    ATK_ROLE_SCROLL_PANE = 48,
    ATK_ROLE_SEPARATOR = 49,
    ATK_ROLE_SLIDER = 50,
    ATK_ROLE_SPLIT_PANE = 51,
    ATK_ROLE_SPIN_BUTTON = 52,
    ATK_ROLE_STATUSBAR = 53,
    ATK_ROLE_TABLE = 54,
    ATK_ROLE_TABLE_CELL = 55,
    ATK_ROLE_TABLE_COLUMN_HEADER = 56,
    ATK_ROLE_TABLE_ROW_HEADER = 57,
    ATK_ROLE_TEAR_OFF_MENU_ITEM = 58,
    ATK_ROLE_TERMINAL = 59,
    ATK_ROLE_TEXT = 60,
    ATK_ROLE_TOGGLE_BUTTON = 61,
    ATK_ROLE_TOOL_BAR = 62,
    ATK_ROLE_TOOL_TIP = 63,
    ATK_ROLE_TREE = 64,
    ATK_ROLE_TREE_TABLE = 65,
    ATK_ROLE_UNKNOWN = 66,
    ATK_ROLE_VIEWPORT = 67,
    ATK_ROLE_WINDOW = 68,
    ATK_ROLE_HEADER = 69,
    ATK_ROLE_FOOTER = 70,
    ATK_ROLE_PARAGRAPH = 71,
    ATK_ROLE_RULER = 72,
    ATK_ROLE_APPLICATION = 73,
    ATK_ROLE_AUTOCOMPLETE = 74,
    ATK_ROLE_EDITBAR = 75,
    ATK_ROLE_ENTRY = 77,
    ATK_ROLE_EMBEDDED = 76,
    ATK_ROLE_CHART = 78,
    ATK_ROLE_CAPTION = 79,
    ATK_ROLE_DOCUMENT_FRAME = 80,
    ATK_ROLE_HEADING = 81,
    ATK_ROLE_PAGE = 82,
    ATK_ROLE_SECTION = 83,
    ATK_ROLE_REDUNDANT_OBJECT = 84,
    ATK_ROLE_FORM = 85,
    ATK_ROLE_LINK = 86,
    ATK_ROLE_INPUT_METHOD_WINDOW = 87,
    ATK_ROLE_TABLE_ROW = 88,
    ATK_ROLE_TREE_ITEM = 89,
    ATK_ROLE_DOCUMENT_SPREADSHEET = 90,
    ATK_ROLE_DOCUMENT_PRESENTATION = 91,
    ATK_ROLE_DOCUMENT_TEXT = 92,
    ATK_ROLE_DOCUMENT_WEB = 93,
    ATK_ROLE_DOCUMENT_EMAIL = 94,
    ATK_ROLE_COMMENT = 95,
    ATK_ROLE_LIST_BOX = 96,
    ATK_ROLE_GROUPING = 97,
    ATK_ROLE_IMAGE_MAP = 98,
    ATK_ROLE_NOTIFICATION = 99,
    ATK_ROLE_INFO_BAR = 100,
    ATK_ROLE_LAST_DEFINED = 101
} AtkRole;

typedef struct _AtkRelationSet {
    GObject parent;
    GPtrArray *relations;

```

```

} AtkRelationSet;
typedef enum {
    ATK_LAYER_INVALID = 0,
    ATK_LAYER_BACKGROUND = 1,
    ATK_LAYER_CANVAS = 2,
    ATK_LAYER_WIDGET = 3,
    ATK_LAYER_MDI = 4,
    ATK_LAYER_POPUP = 5,
    ATK_LAYER_OVERLAY = 6,
    ATK_LAYER_WINDOW = 7
} AtkLayer;
typedef struct _AtkPropertyValues {
    const gchar *property_name;
    GValue old_value;
    GValue new_value;
} AtkPropertyValues;
typedef void (*AtkPropertyChangeHandler) (AtkObject *,
                                           AtkPropertyValues *);

typedef struct _AtkImage AtkImage;
typedef enum {
    ATK_XY_SCREEN = 0,
    ATK_XY_WINDOW = 1
} AtkCoordType;
typedef enum {
    ATK_RELATION_NULL = 0,
    ATK_RELATION_CONTROLLED_BY = 1,
    ATK_RELATION_CONTROLLER_FOR = 2,
    ATK_RELATION_LABEL_FOR = 3,
    ATK_RELATION_LABELLED_BY = 4,
    ATK_RELATION_MEMBER_OF = 5,
    ATK_RELATION_NODE_CHILD_OF = 6,
    ATK_RELATION_FLOWS_TO = 7,
    ATK_RELATION_FLOWS_FROM = 8,
    ATK_RELATION_SUBWINDOW_OF = 9,
    ATK_RELATION_EMBEDS = 10,
    ATK_RELATION_EMBEDDED_BY = 11,
    ATK_RELATION_POPUP_FOR = 12,
    ATK_RELATION_PARENT_WINDOW_OF = 13,
    ATK_RELATION_DESCRIBED_BY = 14,
    ATK_RELATION_DESCRIPTION_FOR = 15,
    ATK_RELATION_NODE_PARENT_OF = 16,
    ATK_RELATION_LAST_DEFINED = 17
} AtkRelationType;
typedef struct _AtkRelation {
    GObject parent;
    GPtrArray *target;
    AtkRelationType relationship;
} AtkRelation;
typedef struct _AtkDocument AtkDocument;
typedef struct _AtkTable AtkTable;
typedef struct _AtkHyperlink {
    GObject parent;
} AtkHyperlink;
typedef struct _AtkStateSet {
    GObject parent;
} AtkStateSet;
typedef struct _AtkObjectFactory {
    GObject parent;
} AtkObjectFactory;
typedef struct _AtkRegistry {
    GObject parent;
    GHashTable *factory_type_registry;
    GHashTable *factory_singleton_cache;
} AtkRegistry;
typedef struct _AtkEditableText AtkEditableText;
typedef GSList AtkAttributeSet;

```

```

typedef struct _AtkHypertext AtkHypertext;
typedef struct _AtkSelection AtkSelection;
typedef struct _AtkAction AtkAction;
typedef struct _AtkComponent AtkComponent;
typedef struct _AtkValue AtkValue;
typedef struct _AtkStreamableContent AtkStreamableContent;
typedef void (*AtkEventListenerInit) (void);
typedef enum {
    ATK_TEXT_ATTR_INVALID = 0,
    ATK_TEXT_ATTR_LEFT_MARGIN = 1,
    ATK_TEXT_ATTR_RIGHT_MARGIN = 2,
    ATK_TEXT_ATTR_INDENT = 3,
    ATK_TEXT_ATTR_INVISIBLE = 4,
    ATK_TEXT_ATTR_EDITABLE = 5,
    ATK_TEXT_ATTR_PIXELS_ABOVE_LINES = 6,
    ATK_TEXT_ATTR_PIXELS_BELOW_LINES = 7,
    ATK_TEXT_ATTR_PIXELS_INSIDE_WRAP = 8,
    ATK_TEXT_ATTR_BG_FULL_HEIGHT = 9,
    ATK_TEXT_ATTR_RISE = 10,
    ATK_TEXT_ATTR_UNDERLINE = 11,
    ATK_TEXT_ATTR_STRIKETHROUGH = 12,
    ATK_TEXT_ATTR_SIZE = 13,
    ATK_TEXT_ATTR_SCALE = 14,
    ATK_TEXT_ATTR_WEIGHT = 15,
    ATK_TEXT_ATTR_LANGUAGE = 16,
    ATK_TEXT_ATTR_FAMILY_NAME = 17,
    ATK_TEXT_ATTR_BG_COLOR = 18,
    ATK_TEXT_ATTR_FG_COLOR = 19,
    ATK_TEXT_ATTR_BG_STIPPLE = 20,
    ATK_TEXT_ATTR_FG_STIPPLE = 21,
    ATK_TEXT_ATTR_WRAP_MODE = 22,
    ATK_TEXT_ATTR_DIRECTION = 23,
    ATK_TEXT_ATTR_JUSTIFICATION = 24,
    ATK_TEXT_ATTR_STRETCH = 25,
    ATK_TEXT_ATTR_VARIANT = 26,
    ATK_TEXT_ATTR_STYLE = 27,
    ATK_TEXT_ATTR_LAST_DEFINED = 28
} AtkTextAttribute;
typedef enum {
    ATK_TEXT_BOUNDARY_CHAR = 0,
    ATK_TEXT_BOUNDARY_WORD_START = 1,
    ATK_TEXT_BOUNDARY_WORD_END = 2,
    ATK_TEXT_BOUNDARY_SENTENCE_START = 3,
    ATK_TEXT_BOUNDARY_SENTENCE_END = 4,
    ATK_TEXT_BOUNDARY_LINE_START = 5,
    ATK_TEXT_BOUNDARY_LINE_END = 6
} AtkTextBoundary;
typedef enum {
    ATK_STATE_INVALID = 0,
    ATK_STATE_ACTIVE = 1,
    ATK_STATE_ARMED = 2,
    ATK_STATE_BUSY = 3,
    ATK_STATE_CHECKED = 4,
    ATK_STATE_DEFUNCT = 5,
    ATK_STATE_EDITABLE = 6,
    ATK_STATE_ENABLED = 7,
    ATK_STATE_EXPANDABLE = 8,
    ATK_STATE_EXPANDED = 9,
    ATK_STATE_FOCUSABLE = 10,
    ATK_STATE_FOCUSED = 11,
    ATK_STATE_HORIZONTAL = 12,
    ATK_STATE_ICONIFIED = 13,
    ATK_STATE_MODAL = 14,
    ATK_STATE_MULTI_LINE = 15,
    ATK_STATE_MULTISELECTABLE = 16,
    ATK_STATE_OPAQUE = 17,

```

```

    ATK_STATE_PRESSED = 18,
    ATK_STATE_RESIZABLE = 19,
    ATK_STATE_SELECTABLE = 20,
    ATK_STATE_SELECTED = 21,
    ATK_STATE_SENSITIVE = 22,
    ATK_STATE_SHOWING = 23,
    ATK_STATE_SINGLE_LINE = 24,
    ATK_STATE_STALE = 25,
    ATK_STATE_TRANSIENT = 26,
    ATK_STATE_VERTICAL = 27,
    ATK_STATE_VISIBLE = 28,
    ATK_STATE_MANAGES_DESCENDANTS = 29,
    ATK_STATE_INDETERMINATE = 30,
    ATK_STATE_TRUNCATED = 31,
    ATK_STATE_REQUIRED = 32,
    ATK_STATE_INVALID_ENTRY = 33,
    ATK_STATE_SUPPORTS_AUTOCOMPLETION = 34,
    ATK_STATE_SELECTABLE_TEXT = 35,
    ATK_STATE_DEFAULT = 36,
    ATK_STATE_ANIMATED = 37,
    ATK_STATE_VISITED = 38,
    ATK_STATE_LAST_DEFINED = 39
} AtkStateType;
typedef struct _AtkGObjectAccessible {
    AtkObject parent;
} AtkGObjectAccessible;
typedef guint64 AtkState;
typedef void (*AtkFocusHandler) (AtkObject*, gboolean);
typedef struct _AtkKeyEventStruct {
    gint type;
    guint state;
    guint keyval;
    gint length;
    gchar *string;
    guint16 keycode;
    guint32 timestamp;
} AtkKeyEventStruct;
typedef gint (*AtkKeySnoopFunc) (AtkKeyEventStruct *, gpointer);
typedef struct _AtkTextRectangle {
    gint x;
    gint y;
    gint width;
    gint height;
} AtkTextRectangle;
typedef struct _AtkTextRange {
    AtkTextRectangle bounds;
    gint start_offset;
    gint end_offset;
    gchar *content;
} AtkTextRange;
typedef enum {
    ATK_TEXT_CLIP_NONE = 0,
    ATK_TEXT_CLIP_MIN = 1,
    ATK_TEXT_CLIP_MAX = 2,
    ATK_TEXT_CLIP_BOTH = 3
} AtkTextClipType;
typedef struct _AtkImplementor AtkImplementor;
typedef void (*AtkEventListener) (AtkObject *);
typedef gboolean (*AtkFunction) (gpointer);
typedef struct _AtkImplementorIface {
    GTypeInterface parent;
    AtkObject *(*ref_accessible) (AtkImplementor *);
} AtkImplementorIface;
typedef struct _AtkRegistryClass {
    GObjectClass parent_class;
} AtkRegistryClass;

```

```

typedef struct _AtkHypertextIface {
    GTypeInterface parent;
    AtkHyperlink *(*get_link) (AtkHypertext *, gint);
    gint(*get_n_links) (AtkHypertext *);
    gint(*get_link_index) (AtkHypertext *, gint);
    void(*link_selected) (AtkHypertext *, gint);
    AtkFunction pad1;
    AtkFunction pad2;
    AtkFunction pad3;
} AtkHypertextIface;
typedef struct _AtkStreamableContentIface {
    GTypeInterface parent;
    gint(*get_n_mime_types) (AtkStreamableContent *);
    const gchar *(*get_mime_type) (AtkStreamableContent *, gint);
    GIOChannel *(*get_stream) (AtkStreamableContent *, const gchar
*);
    const gchar *(*get_uri) (AtkStreamableContent *, const gchar*);
    AtkFunction pad1;
    AtkFunction pad2;
    AtkFunction pad3;
} AtkStreamableContentIface;
typedef struct _AtkRectangle {
    gint x;
    gint y;
    gint width;
    gint height;
} AtkRectangle;
typedef struct _AtkStateSetClass {
    GObjectClass parent;
} AtkStateSetClass;
typedef struct _AtkAttribute {
    gchar *name;
    gchar *value;
} AtkAttribute;
typedef struct _AtkObjectClass {
    GObjectClass parent;
    const gchar *(*get_name) (AtkObject *);
    const gchar *(*get_description) (AtkObject *);
    AtkObject *(*get_parent) (AtkObject *);
    gint(*get_n_children) (AtkObject *);
    AtkObject *(*ref_child) (AtkObject *, gint);
    gint(*get_index_in_parent) (AtkObject *);
    AtkRelationSet *(*ref_relation_set) (AtkObject *);
    AtkRole(*get_role) (AtkObject *);
    AtkLayer(*get_layer) (AtkObject *);
    gint(*get_mdi_zorder) (AtkObject *);
    AtkStateSet *(*ref_state_set) (AtkObject *);
    void(*set_name) (AtkObject *, const gchar *);
    void(*set_description) (AtkObject *, const gchar *);
    void(*set_parent) (AtkObject *, AtkObject *);
    void(*set_role) (AtkObject *, AtkRole);
    guint(*connect_property_change_handler) (AtkObject *,
AtkPropertyChangeHandler *);
    void(*remove_property_change_handler) (AtkObject *, guint);
    void(*initialize) (AtkObject *, gpointer);
    void(*children_changed) (AtkObject *, guint, gpointer);
    void(*focus_event) (AtkObject *, gboolean);
    void(*property_change) (AtkObject *, AtkPropertyValues *);
    void(*state_change) (AtkObject *, const gchar *, gboolean);
    void(*visible_data_changed) (AtkObject *);
    void(*active_descendant_changed) (AtkObject *, gpointer *);
    AtkAttributeSet *(*get_attributes) (AtkObject *);
    AtkFunction pad1;
    AtkFunction pad2;
} AtkObjectClass;
typedef struct _AtkGObjectAccessibleClass {

```

```

    AtkObjectClass parent_class;
    AtkFunction pad1;
    AtkFunction pad2;
} AtkGObjectAccessibleClass;
typedef struct _AtkObjectFactoryClass {
    GObjectClass parent_class;
    AtkObject *(*create_accessible) (GObject *);
    void (*invalidate) (AtkObjectFactory *);
    GType(*get_accessible_type) (void);
    AtkFunction pad1;
    AtkFunction pad2;
} AtkObjectFactoryClass;
typedef struct _AtkNoOpObjectFactoryClass {
    AtkObjectFactoryClass parent_class;
} AtkNoOpObjectFactoryClass;
typedef struct _AtkActionIface {
    GTypeInterface parent;
    gboolean(*do_action) (AtkAction *, gint);
    gint(*get_n_actions) (AtkAction *);
    const gchar *(*get_description) (AtkAction *, gint);
    const gchar *(*get_name) (AtkAction *, gint);
    const gchar *(*get_keybinding) (AtkAction *, gint);
    gboolean(*set_description) (AtkAction *, gint, const gchar *);
    const gchar *(*get_localized_name) (AtkAction *, gint);
    AtkFunction pad2;
} AtkActionIface;
typedef struct _AtkTableIface {
    GTypeInterface parent;
    AtkObject *(*ref_at) (AtkTable *, gint, gint);
    gint(*get_index_at) (AtkTable *, gint, gint);
    gint(*get_column_at_index) (AtkTable *, gint);
    gint(*get_row_at_index) (AtkTable *, gint);
    gint(*get_n_columns) (AtkTable *);
    gint(*get_n_rows) (AtkTable *);
    gint(*get_column_extent_at) (AtkTable *, gint, gint);
    gint(*get_row_extent_at) (AtkTable *, gint, gint);
    AtkObject *(*get_caption) (AtkTable *);
    const gchar *(*get_column_description) (AtkTable *, gint);
    AtkObject *(*get_column_header) (AtkTable *, gint);
    const gchar *(*get_row_description) (AtkTable *, gint);
    AtkObject *(*get_row_header) (AtkTable *, gint);
    AtkObject *(*get_summary) (AtkTable *);
    void (*set_caption) (AtkTable *, AtkObject *);
    void (*set_column_description) (AtkTable *, gint, const gchar
*);
    void (*set_column_header) (AtkTable *, gint, AtkObject *);
    void (*set_row_description) (AtkTable *, gint, const gchar *);
    void (*set_row_header) (AtkTable *, gint, AtkObject *);
    void (*set_summary) (AtkTable *, AtkObject *);
    gint(*get_selected_columns) (AtkTable *, gint *);
    gint(*get_selected_rows) (AtkTable *, gint *);
    gboolean(*is_column_selected) (AtkTable *, gint);
    gboolean(*is_row_selected) (AtkTable *, gint);
    gboolean(*is_selected) (AtkTable *, gint, gint);
    gboolean(*add_row_selection) (AtkTable *, gint);
    gboolean(*remove_row_selection) (AtkTable *, gint);
    gboolean(*add_column_selection) (AtkTable *, gint);
    gboolean(*remove_column_selection) (AtkTable *, gint);
    void (*row_inserted) (AtkTable *, gint, gint);
    void (*column_inserted) (AtkTable *, gint, gint);
    void (*row_deleted) (AtkTable *, gint, gint);
    void (*column_deleted) (AtkTable *, gint, gint);
    void (*row_reordered) (AtkTable *);
    void (*column_reordered) (AtkTable *);
    void (*model_changed) (AtkTable *);
    AtkFunction pad1;

```

```

    AtkFunction pad2;
    AtkFunction pad3;
    AtkFunction pad4;
} AtkTableIface;
typedef struct _AtkValueIface {
    GTypeInterface parent;
    void (*get_current_value) (AtkValue *, GValue *);
    void (*get_maximum_value) (AtkValue *, GValue *);
    void (*get_minimum_value) (AtkValue *, GValue *);
    gboolean (*set_current_value) (AtkValue *, const GValue *);
} AtkValueIface;
typedef struct _AtkRelationClass {
    GObjectClass parent;
} AtkRelationClass;
typedef struct _AtkTextIface {
    GTypeInterface parent;
    gchar *(*get_text) (AtkText *, gint, gint);
    gchar *(*get_text_after_offset) (AtkText *, gint,
    AtkTextBoundary,
    gint *, gint *);
    gchar *(*get_text_at_offset) (AtkText *, gint, AtkTextBoundary,
    gint *,
    gint *);
    gunichar(*get_character_at_offset) (AtkText *, gint);
    gchar *(*get_text_before_offset) (AtkText *, gint,
    AtkTextBoundary,
    gint *, gint *);
    gint(*get_caret_offset) (AtkText *);
    AtkAttributeSet *(*get_run_attributes) (AtkText *, gint, gint
    *,
    gint *);
    AtkAttributeSet *(*get_default_attributes) (AtkText *);
    void (*get_character_extents) (AtkText *, gint, gint *, gint *,
    gint *,
    gint *, AtkCoordType);
    gint(*get_character_count) (AtkText *);
    gint(*get_offset_at_point) (AtkText *, gint, gint,
    AtkCoordType);
    gint(*get_n_selections) (AtkText *);
    gchar *(*get_selection) (AtkText *, gint, gint *, gint *);
    gboolean(*add_selection) (AtkText *, gint, gint);
    gboolean(*remove_selection) (AtkText *, gint);
    gboolean(*set_selection) (AtkText *, gint, gint, gint);
    gboolean(*set_caret_offset) (AtkText *, gint);
    void (*text_changed) (AtkText *, gint, gint);
    void (*text_caret_moved) (AtkText *, gint);
    void (*text_selection_changed) (AtkText *);
    void (*text_attributes_changed) (AtkText *);
    void (*get_range_extents) (AtkText *, gint, gint, AtkCoordType,
    AtkTextRectangle *);
    AtkTextRange **(*get_bounded_ranges) (AtkText *,
    AtkTextRectangle *,
    AtkCoordType, AtkTextClipType,
    AtkTextClipType);

    AtkFunction pad4;
} AtkTextIface;
typedef struct _AtkRelationSetClass {
    GObjectClass parent;
    AtkFunction pad1;
    AtkFunction pad2;
} AtkRelationSetClass;
typedef struct _AtkSelectionIface {
    GTypeInterface parent;
    gboolean(*add_selection) (AtkSelection *, gint);
    gboolean(*clear_selection) (AtkSelection *);
    AtkObject *(*ref_selection) (AtkSelection *, gint);

```



```

    gint(*get_selection_count) (AtkSelection *);
    gboolean(*is_child_selected) (AtkSelection *, gint);
    gboolean(*remove_selection) (AtkSelection *, gint);
    gboolean(*select_all_selection) (AtkSelection *);
    void (*selection_changed) (AtkSelection *);
    AtkFunction pad1;
    AtkFunction pad2;
} AtkSelectionIface;
typedef struct _AtkUtilClass {
    GObjectClass parent;
    guint(*add_global_event_listener) (GSignalEmissionHook,
                                       const gchar *);
    void (*remove_global_event_listener) (guint);
    guint(*add_key_event_listener) (AtkKeySnoopFunc, gpointer);
    void (*remove_key_event_listener) (guint);
    AtkObject *(*get_root) (void);
    const gchar *(*get_toolkit_name) (void);
    const gchar *(*get_toolkit_version) (void);
} AtkUtilClass;
typedef struct _AtkNoOpObjectClass {
    AtkObjectClass parent_class;
} AtkNoOpObjectClass;
typedef struct _AtkImageIface {
    GTypeInterface parent;
    void (*get_image_position) (AtkImage *, gint *, gint *,
                                AtkCoordType);
    const gchar *(*get_image_description) (AtkImage *, AtkImage *);
    void (*get_image_size) (AtkImage *, gint *, gint *);
    gboolean(*set_image_description) (AtkImage *, const gchar *);
    const gchar *(*get_image_locale) (AtkImage *, AtkImage *);
    AtkFunction pad1;
} AtkImageIface;
typedef struct _AtkComponentIface {
    GTypeInterface parent;
    guint(*add_focus_handler) (AtkComponent *, AtkFocusHandler);
    gboolean(*contains) (AtkComponent *, gint, gint, AtkCoordType);
    AtkObject *(*ref_accessible_at_point) (AtkComponent *, gint,
                                           gint,
                                           AtkCoordType);
    void (*get_extents) (AtkComponent *, gint *, gint *, gint *,
                        gint *,
                        AtkCoordType);
    void (*get_position) (AtkComponent *, gint *, gint *,
                        AtkCoordType);
    void (*get_size) (AtkComponent *, gint *, gint *);
    gboolean(*grab_focus) (AtkComponent *);
    void (*remove_focus_handler) (AtkComponent *, guint);
    gboolean(*set_extents) (AtkComponent *, gint, gint, gint, gint,
                           AtkCoordType);
    gboolean(*set_position) (AtkComponent *, gint, gint,
                           AtkCoordType);
    gboolean(*set_size) (AtkComponent *, gint, gint);
    AtkLayer(*get_layer) (AtkComponent *);
    gint(*get_mdi_zorder) (AtkComponent *);
    void (*bounds_changed) (AtkComponent *, AtkRectangle *);
    gdouble(*get_alpha) (AtkComponent *);
} AtkComponentIface;
typedef struct _AtkEditableTextIface {
    GTypeInterface parent_interface;
    gboolean(*set_run_attributes) (AtkEditableText *,
                                   AtkAttributeSet *,
                                   gint, gint);
    void (*set_text_contents) (AtkEditableText *, const gchar *);
    void (*insert_text) (AtkEditableText *, const gchar *, gint,
                        gint *);
    void (*copy_text) (AtkEditableText *, gint, gint);

```

```

void (*cut_text) (AtkEditableText *, gint, gint);
void (*delete_text) (AtkEditableText *, gint, gint);
void (*paste_text) (AtkEditableText *, gint);
AtkFunction pad1;
AtkFunction pad2;
} AtkEditableTextIface;
typedef struct _AtkDocumentIface {
    GTypeInterface parent;
    const gchar *(*get_document_type) (AtkDocument *);
    gpointer(*get_document) (AtkDocument *);
    const gchar *(*get_document_locale) (AtkDocument *);
    AtkAttributeSet *(*get_document_attributes) (AtkDocument *);
    const gchar *(*get_document_attribute_value) (AtkDocument *,
                                                const gchar *);
    gboolean(*set_document_attribute) (AtkDocument *, const gchar
*,
                                   const gchar *);

    AtkFunction pad1;
    AtkFunction pad2;
    AtkFunction pad3;
    AtkFunction pad4;
} AtkDocumentIface;
typedef struct _AtkHyperlinkClass {
    GObjectClass parent;
    gchar *(*get_uri) (AtkHyperlink *, gint);
    AtkObject *(*get_object) (AtkHyperlink *, gint);
    gint(*get_end_index) (AtkHyperlink *);
    gint(*get_start_index) (AtkHyperlink *);
    gboolean(*is_valid) (AtkHyperlink *);
    gint(*get_n_anchors) (AtkHyperlink *);
    guint(*link_state) (AtkHyperlink *);
    gboolean(*is_selected_link) (AtkHyperlink *);
    void (*link_activated) (AtkHyperlink *);
    AtkFunction pad1;
} AtkHyperlinkClass;
typedef enum {
    ATK_KEY_EVENT_PRESS,
    ATK_KEY_EVENT_RELEASE,
    ATK_KEY_EVENT_LAST_DEFINED
} AtkKeyEventTypes;
typedef struct _AtkUtil {
    GObject parent;
} AtkUtil;
typedef enum {
    ATK_HYPERLINK_IS_INLINE = 1 << 0
} AtkHyperlinkStateFlags;
typedef struct _AtkNoOpObjectFactory {
    AtkObjectFactory parent;
} AtkNoOpObjectFactory;
typedef struct _AtkNoOpObject {
    AtkObject parent;
} AtkNoOpObject;
typedef struct _AtkMisc {
    GObject parent;
} AtkMisc;
typedef struct _AtkMiscClass {
    GObjectClass parent;
    void (*threads_enter) (AtkMisc *);
    void (*threads_leave) (AtkMisc *);
    gpointer vfuncs[32];
} AtkMiscClass;
typedef struct _AtkHyperlinkImpl AtkHyperlinkImpl;
typedef struct _AtkHyperlinkImplIface {
    GTypeInterface parent;
    AtkHyperlink *(*get_hyperlink) (AtkHyperlinkImpl *);
    AtkFunction pad1;

```

```

} AtkHyperlinkImpliface;
extern gboolean atk_action_do_action(AtkAction * action, gint i);
extern const gchar *atk_action_get_description(AtkAction * action,
gint i);
extern const gchar *atk_action_get_keybinding(AtkAction * action,
gint i);
extern const gchar *atk_action_get_localized_name(AtkAction *
action,
gint i);
extern gint atk_action_get_n_actions(AtkAction * action);
extern const gchar *atk_action_get_name(AtkAction * action, gint
i);
extern GType atk_action_get_type(void);
extern gboolean atk_action_set_description(AtkAction * action, gint
i,
const gchar * desc);
extern guint atk_add_focus_tracker(AtkEventListener focus_tracker);
extern guint atk_add_global_event_listener(GSignalEmissionHook
listener,
const gchar * event_type);
extern guint atk_add_key_event_listener(AtkKeySnoopFunc listener,
gpointer data);
extern void atk_attribute_set_free(AtkAttributeSet * attrib_set);
extern guint atk_component_add_focus_handler(AtkComponent *
component,
AtkFocusHandler handler);
extern gboolean atk_component_contains(AtkComponent * component,
gint x,
gint y, AtkCoordType coord_type);
extern gdouble atk_component_get_alpha(AtkComponent * component);
extern void atk_component_get_extents(AtkComponent * component,
gint * x,
gint * y, gint * width,
gint * height,
AtkCoordType coord_type);
extern AtkLayer atk_component_get_layer(AtkComponent * component);
extern gint atk_component_get_mdi_zorder(AtkComponent * component);
extern void atk_component_get_position(AtkComponent * component,
gint * x,
gint * y, AtkCoordType coord_type);
extern void atk_component_get_size(AtkComponent * component, gint
* width,
gint * height);
extern GType atk_component_get_type(void);
extern gboolean atk_component_grab_focus(AtkComponent * component);
extern AtkObject
*atk_component_ref_accessible_at_point(AtkComponent *
component, gint x,
gint y,
AtkCoordType
coord_type);
extern void atk_component_remove_focus_handler(AtkComponent *
component,
guint handler_id);
extern gboolean atk_component_set_extents(AtkComponent * component,
gint x,
gint y, gint width, gint height,
AtkCoordType coord_type);
extern gboolean atk_component_set_position(AtkComponent *
component,
gint x, gint y,
AtkCoordType coord_type);
extern gboolean atk_component_set_size(AtkComponent * component,
gint width, gint height);
extern const char *atk_document_get_attribute_value(AtkDocument *
document,

```

```

        const char
            *attribute_name);
extern AtkAttributeSet *atk_document_get_attributes(AtkDocument *
    document);
extern gpointer atk_document_get_document(AtkDocument * document);
extern const gchar *atk_document_get_document_type(AtkDocument *
    document);
extern const char *atk_document_get_locale(AtkDocument * document);
extern GType atk_document_get_type(void);
extern gboolean atk_document_set_attribute_value(AtkDocument *
    document,
        const gchar *
            attribute_name,
        const gchar *
            attribute_value);
extern void atk_editable_text_copy_text(AtkEditableText * text,
    gint start_pos, gint end_pos);
extern void atk_editable_text_cut_text(AtkEditableText * text,
    gint start_pos, gint end_pos);
extern void atk_editable_text_delete_text(AtkEditableText * text,
    gint start_pos, gint end_pos);
extern GType atk_editable_text_get_type(void);
extern void atk_editable_text_insert_text(AtkEditableText * text,
    const gchar * string,
    gint length, gint * position);
extern void atk_editable_text_paste_text(AtkEditableText * text,
    gint position);
extern
    gboolean
atk_editable_text_set_run_attributes(AtkEditableText *
    text,
    AtkAttributeSet *
        attrib_set,
    gint start_offset,
    gint end_offset);
extern void atk_editable_text_set_text_contents(AtkEditableText *
    text,
        const gchar * string);
extern void atk_focus_tracker_init(AtkEventListenerInit init);
extern void atk_focus_tracker_notify(AtkObject * object);
extern AtkRegistry *atk_get_default_registry(void);
extern AtkObject *atk_get_focus_object(void);
extern AtkObject *atk_get_root(void);
extern const gchar *atk_get_toolkit_name(void);
extern const gchar *atk_get_toolkit_version(void);
extern const char *atk_get_version(void);
extern AtkObject *atk_gobject_accessible_for_object(GObject * obj);
extern
    GObject
*atk_gobject_accessible_get_object(AtkGObjectAccessible *
    obj);
extern GType atk_gobject_accessible_get_type(void);
extern gint atk_hyperlink_get_end_index(AtkHyperlink * link_);
extern gint atk_hyperlink_get_n_anchors(AtkHyperlink * link_);
extern AtkObject *atk_hyperlink_get_object(AtkHyperlink * link_,
    gint i);
extern gint atk_hyperlink_get_start_index(AtkHyperlink * link_);
extern GType atk_hyperlink_get_type(void);
extern gchar *atk_hyperlink_get_uri(AtkHyperlink * link_, gint i);
extern
    AtkHyperlink
*atk_hyperlink_impl_get_hyperlink(AtkHyperlinkImpl *
    obj);
extern GType atk_hyperlink_impl_get_type(void);
extern gboolean atk_hyperlink_is_inline(AtkHyperlink * link_);
extern gboolean atk_hyperlink_is_selected_link(AtkHyperlink *
    link_);
extern gboolean atk_hyperlink_is_valid(AtkHyperlink * link_);

```

```

extern AtkHyperlink *atk_hypertext_get_link(AtkHypertext *
hypertext,
                                gint link_index);
extern gint atk_hypertext_get_link_index(AtkHypertext * hypertext,
                                gint char_index);
extern gint atk_hypertext_get_n_links(AtkHypertext * hypertext);
extern GType atk_hypertext_get_type(void);
extern const gchar *atk_image_get_image_description(AtkImage *
image);
extern const char *atk_image_get_image_locale(AtkImage * image);
extern void atk_image_get_image_position(AtkImage * image, gint *
x,
                                gint * y,
                                AtkCoordType coord_type);
extern void atk_image_get_image_size(AtkImage * image, int *width,
int *height);
extern GType atk_image_get_type(void);
extern gboolean atk_image_set_image_description(AtkImage * image,
const gchar * description);
extern GType atk_implementor_get_type(void);
extern AtkObject *atk_implementor_ref_accessible(AtkImplementor *
implementor);
extern const AtkMisc *atk_misc_get_instance(void);
extern GType atk_misc_get_type(void);
extern void atk_misc_threads_enter(AtkMisc * misc);
extern void atk_misc_threads_leave(AtkMisc * misc);
extern GType atk_no_op_object_factory_get_type(void);
extern AtkObjectFactory *atk_no_op_object_factory_new(void);
extern GType atk_no_op_object_get_type(void);
extern AtkObject *atk_no_op_object_new(AtkObject *);
extern gboolean atk_object_add_relationship(AtkObject * object,
AtkRelationType relationship,
AtkObject * target);
extern guint atk_object_connect_property_change_handler(AtkObject
*
                                accessible,

AtkPropertyChangeHandler
                                * handler);
extern AtkObject
*atk_object_factory_create_accessible(AtkObjectFactory *
factory,
GObject * obj);
extern GType
atk_object_factory_get_accessible_type(AtkObjectFactory *
factory);
extern GType atk_object_factory_get_type(void);
extern void atk_object_factory_invalidate(AtkObjectFactory *
factory);
extern AtkAttributeSet *atk_object_get_attributes(AtkObject *
accessible);
extern const gchar *atk_object_get_description(AtkObject *
accessible);
extern gint atk_object_get_index_in_parent(AtkObject * accessible);
extern gint atk_object_get_n_accessible_children(AtkObject *
accessible);
extern const gchar *atk_object_get_name(AtkObject * accessible);
extern AtkObject *atk_object_get_parent(AtkObject * accessible);
extern AtkRole atk_object_get_role(AtkObject * accessible);
extern GType atk_object_get_type(void);
extern void atk_object_initialize(AtkObject * accessible, gpointer
data);
extern void atk_object_notify_state_change(AtkObject * accessible,
AtkState state, gboolean
value);

```

```

extern AtkObject *atk_object_ref_accessible_child(AtkObject *
accessible,
gint i);
extern AtkRelationSet *atk_object_ref_relation_set(AtkObject *
accessible);
extern AtkStateSet *atk_object_ref_state_set(AtkObject *
accessible);
extern void atk_object_remove_property_change_handler(AtkObject *
accessible,
guint handler_id);
extern gboolean atk_object_remove_relationship(AtkObject * object,
AtkRelationType
relationship,
AtkObject * target);
extern void atk_object_set_description(AtkObject * accessible,
const gchar * description);
extern void atk_object_set_name(AtkObject * accessible,
const gchar * name);
extern void atk_object_set_parent(AtkObject * accessible,
AtkObject * parent);
extern void atk_object_set_role(AtkObject * accessible, AtkRole
role);
extern GType atk_rectangle_get_type(void);
extern AtkObjectFactory *atk_registry_get_factory(AtkRegistry *
registry,
GType type);
extern GType atk_registry_get_factory_type(AtkRegistry * registry,
GType type);
extern GType atk_registry_get_type(void);
extern void atk_registry_set_factory_type(AtkRegistry * registry,
GType type, GType factory_type);
extern void atk_relation_add_target(AtkRelation * relation,
AtkObject * target);
extern AtkRelationType atk_relation_get_relation_type(AtkRelation
*
relation);
extern GPtrArray *atk_relation_get_target(AtkRelation * relation);
extern GType atk_relation_get_type(void);
extern AtkRelation *atk_relation_new(AtkObject * *targets, gint
n_targets,
AtkRelationType relationship);
extern gboolean atk_relation_remove_target(AtkRelation * relation,
AtkObject * target);
extern void atk_relation_set_add(AtkRelationSet * set,
AtkRelation * relation);
extern void atk_relation_set_add_relation_by_type(AtkRelationSet *
set,
AtkRelationType
relationship,
AtkObject * target);
extern gboolean atk_relation_set_contains(AtkRelationSet * set,
AtkRelationType relationship);
extern gint atk_relation_set_get_n_relations(AtkRelationSet * set);
extern AtkRelation *atk_relation_set_get_relation(AtkRelationSet *
set,
gint i);
extern AtkRelation
*atk_relation_set_get_relation_by_type(AtkRelationSet *
set,
AtkRelationType
relationship);
extern GType atk_relation_set_get_type(void);
extern AtkRelationSet *atk_relation_set_new(void);
extern void atk_relation_set_remove(AtkRelationSet * set,
AtkRelation * relation);

```

```

extern AtkRelationType atk_relation_type_for_name(const gchar *
name);
extern const gchar *atk_relation_type_get_name(AtkRelationType
type);
extern AtkRelationType atk_relation_type_register(const gchar *
name);
extern void atk_remove_focus_tracker(guint tracker_id);
extern void atk_remove_global_event_listener(guint listener_id);
extern void atk_remove_key_event_listener(guint listener_id);
extern AtkRole atk_role_for_name(const gchar * name);
extern const gchar *atk_role_get_localized_name(AtkRole role);
extern const gchar *atk_role_get_name(AtkRole role);
extern AtkRole atk_role_register(const gchar * name);
extern gboolean atk_selection_add_selection(AtkSelection *
selection,
gint i);
extern gboolean atk_selection_clear_selection(AtkSelection *
selection);
extern gint atk_selection_get_selection_count(AtkSelection *
selection);
extern GType atk_selection_get_type(void);
extern gboolean atk_selection_is_child_selected(AtkSelection *
selection,
gint i);
extern AtkObject *atk_selection_ref_selection(AtkSelection *
selection,
gint i);
extern gboolean atk_selection_remove_selection(AtkSelection *
selection,
gint i);
extern gboolean atk_selection_select_all_selection(AtkSelection *
selection);
extern gboolean atk_state_set_add_state(AtkStateSet * set,
AtkStateType type);
extern void atk_state_set_add_states(AtkStateSet * set,
AtkStateType * types, gint n_types);
extern AtkStateSet *atk_state_set_and_sets(AtkStateSet * set,
AtkStateSet * compare_set);
extern void atk_state_set_clear_states(AtkStateSet * set);
extern gboolean atk_state_set_contains_state(AtkStateSet * set,
AtkStateType type);
extern gboolean atk_state_set_contains_states(AtkStateSet * set,
AtkStateType * types,
gint n_types);
extern GType atk_state_set_get_type(void);
extern gboolean atk_state_set_is_empty(AtkStateSet * set);
extern AtkStateSet *atk_state_set_new(void);
extern AtkStateSet *atk_state_set_or_sets(AtkStateSet * set,
AtkStateSet * compare_set);
extern gboolean atk_state_set_remove_state(AtkStateSet * set,
AtkStateType type);
extern AtkStateSet *atk_state_set_xor_sets(AtkStateSet * set,
AtkStateSet * compare_set);
extern AtkStateType atk_state_type_for_name(const gchar * name);
extern const gchar *atk_state_type_get_name(AtkStateType type);
extern AtkStateType atk_state_type_register(const gchar * name);
extern const gchar
*atk_streamable_content_get_mime_type(AtkStreamableContent *
streamable, gint i);
extern
gint
atk_streamable_content_get_n_mime_types(AtkStreamableContent *
streamable);
extern
GIOChannel
*atk_streamable_content_get_stream(AtkStreamableContent *
streamable,
const gchar *

```

```

        mime_type);
extern GType atk_streamable_content_get_type(void);
extern      const      char
*atk_streamable_content_get_uri(AtkStreamableContent *
                                streamable,
                                const char *mime_type);
extern gboolean atk_table_add_column_selection(AtkTable * table,
                                                gint column);
extern gboolean atk_table_add_row_selection(AtkTable * table, gint
row);
extern AtkObject *atk_table_get_caption(AtkTable * table);
extern gint atk_table_get_column_at_index(AtkTable * table, gint
index_);
extern const gchar *atk_table_get_column_description(AtkTable *
table,
                                                gint column);
extern gint atk_table_get_column_extents_at(AtkTable * table, gint
row,
                                                gint column);
extern AtkObject *atk_table_get_column_header(AtkTable * table,
                                                gint column);
extern gint atk_table_get_index_at(AtkTable * table, gint row,
gint column);
extern gint atk_table_get_n_columns(AtkTable * table);
extern gint atk_table_get_n_rows(AtkTable * table);
extern gint atk_table_get_row_at_index(AtkTable * table, gint
index_);
extern const gchar *atk_table_get_row_description(AtkTable * table,
gint row);
extern gint atk_table_get_row_extents_at(AtkTable * table, gint row,
gint column);
extern AtkObject *atk_table_get_row_header(AtkTable * table, gint
row);
extern gint atk_table_get_selected_columns(AtkTable * table,
gint * *selected);
extern gint atk_table_get_selected_rows(AtkTable * table,
gint * *selected);
extern AtkObject *atk_table_get_summary(AtkTable * table);
extern GType atk_table_get_type(void);
extern gboolean atk_table_is_column_selected(AtkTable * table,
gint column);
extern gboolean atk_table_is_row_selected(AtkTable * table, gint
row);
extern gboolean atk_table_is_selected(AtkTable * table, gint row,
gint column);
extern AtkObject *atk_table_ref_at(AtkTable * table, gint row,
gint column);
extern gboolean atk_table_remove_column_selection(AtkTable * table,
gint column);
extern gboolean atk_table_remove_row_selection(AtkTable * table,
gint row);
extern void atk_table_set_caption(AtkTable * table, AtkObject *
caption);
extern void atk_table_set_column_description(AtkTable * table, gint
column,
                                                const gchar * description);
extern void atk_table_set_column_header(AtkTable * table, gint
column,
                                                AtkObject * header);
extern void atk_table_set_row_description(AtkTable * table, gint
row,
                                                const gchar * description);
extern void atk_table_set_row_header(AtkTable * table, gint row,
AtkObject * header);
extern void atk_table_set_summary(AtkTable * table,
AtkObject * accessible);

```



```

extern gboolean  atk_text_add_selection(AtkText * text, gint
start_offset,
                                gint end_offset);
extern AtkTextAttribute atk_text_attribute_for_name(const gchar *
name);
extern const gchar *atk_text_attribute_get_name(AtkTextAttribute
attr);
extern const gchar *atk_text_attribute_get_value(AtkTextAttribute
attr,
                                gint index_);
extern AtkTextAttribute atk_text_attribute_register(const gchar *
name);
extern void atk_text_free_ranges(AtkTextRange * *ranges);
extern AtkTextRange **atk_text_get_bounded_ranges(AtkText * text,
AtkTextRectangle * rect,
AtkCoordType coord_type,
AtkTextClipType
x_clip_type,
AtkTextClipType
y_clip_type);
extern gint atk_text_get_caret_offset(AtkText * text);
extern gunichar atk_text_get_character_at_offset(AtkText * text,
gint offset);
extern gint atk_text_get_character_count(AtkText * text);
extern void atk_text_get_character_extents(AtkText * text, gint
offset,
                                gint * x, gint * y,
                                gint * width, gint * height,
                                AtkCoordType coords);
extern AtkAttributeSet *atk_text_get_default_attributes(AtkText *
text);
extern gint atk_text_get_n_selections(AtkText * text);
extern gint atk_text_get_offset_at_point(AtkText * text, gint x,
gint y,
                                AtkCoordType coords);
extern void atk_text_get_range_extents(AtkText * text, gint
start_offset,
                                gint end_offset,
                                AtkCoordType coord_type,
                                AtkTextRectangle * rect);
extern AtkAttributeSet *atk_text_get_run_attributes(AtkText * text,
gint offset,
gint * start_offset,
gint * end_offset);
extern gchar *atk_text_get_selection(AtkText * text, gint
selection_num,
                                gint * start_offset,
                                gint * end_offset);
extern gchar *atk_text_get_text(AtkText * text, gint start_offset,
gint end_offset);
extern gchar *atk_text_get_text_after_offset(AtkText * text, gint
offset,
                                AtkTextBoundary
boundary_type,
                                gint * start_offset,
                                gint * end_offset);
extern gchar *atk_text_get_text_at_offset(AtkText * text, gint
offset,
                                AtkTextBoundary boundary_type,
                                gint * start_offset,
                                gint * end_offset);
extern gchar *atk_text_get_text_before_offset(AtkText * text, gint
offset,
                                AtkTextBoundary
boundary_type,
                                gint * start_offset,

```

```

                                gint * end_offset);
extern GType atk_text_get_type(void);
extern GType atk_text_range_get_type(void);
extern gboolean atk_text_remove_selection(AtkText * text,
                                           gint selection_num);
extern gboolean atk_text_set_caret_offset(AtkText * text, gint
offset);
extern gboolean atk_text_set_selection(AtkText * text, gint
selection_num,
                                gint start_offset, gint
end_offset);
extern GType atk_util_get_type(void);
extern void atk_value_get_current_value(AtkValue * obj, GValue *
value);
extern void atk_value_get_maximum_value(AtkValue * obj, GValue *
value);
extern void atk_value_get_minimum_increment(AtkValue * obj, GValue *
value);
extern void atk_value_get_minimum_value(AtkValue * obj, GValue *
value);
extern GType atk_value_get_type(void);
extern gboolean atk_value_set_current_value(AtkValue * obj,
const GValue * value);

```

17.15.3 atk-1.0/atk/atkplug.h

```

#define ATK_TYPE_PLUG (atk_plug_get_type())
#define ATK_PLUG_CLASS(klass) (G_TYPE_CHECK_CLASS_CAST ((klass),
ATK_TYPE_PLUG, AtkPlugClass))
#define ATK_IS_PLUG_CLASS(klass) (G_TYPE_CHECK_CLASS_TYPE
((klass), ATK_TYPE_PLUG))
#define ATK_PLUG(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
ATK_TYPE_PLUG, AtkPlug))
#define ATK_IS_PLUG(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_PLUG))
#define ATK_PLUG_GET_CLASS(obj) (G_TYPE_INSTANCE_GET_CLASS ((obj),
ATK_TYPE_PLUG, AtkPlugClass))

typedef struct _AtkPlug {
    AtkObject parent;
} AtkPlug;
typedef struct _AtkPlugClass {
    AtkObjectClass parent_class;
    gchar *(*get_object_id) (AtkPlug * plug);
} AtkPlugClass;
extern gchar *atk_plug_get_id(AtkPlug * plug);
extern GType atk_plug_get_type(void);
extern AtkObject *atk_plug_new(void);

```

17.15.4 atk-1.0/atk/atksocket.h

```

#define ATK_TYPE_SOCKET (atk_socket_get_type ())
#define ATK_SOCKET_CLASS(klass) (G_TYPE_CHECK_CLASS_CAST ((klass),
ATK_TYPE_SOCKET, AtkSocketClass))
#define ATK_IS_SOCKET_CLASS(klass) (G_TYPE_CHECK_CLASS_TYPE
((klass), ATK_TYPE_SOCKET))
#define ATK_SOCKET(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
ATK_TYPE_SOCKET, AtkSocket))
#define ATK_IS_SOCKET(obj) (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
ATK_TYPE_SOCKET))
#define ATK_SOCKET_GET_CLASS(obj) (G_TYPE_INSTANCE_GET_CLASS
((obj), ATK_TYPE_SOCKET, AtkSocketClass))

```

```

typedef struct _AtkSocket {
    AtkObject parent;
    gchar *embedded_plug_id;
} AtkSocket;
typedef struct _AtkSocketClass {
    AtkObjectClass parent_class;
    void (*embed) (AtkSocket * obj, gchar * plug_id);
} AtkSocketClass;
extern void atk_socket_embed(AtkSocket * obj, gchar * plug_id);
extern GType atk_socket_get_type(void);
extern gboolean atk_socket_is_occupied(AtkSocket * obj);
extern AtkObject *atk_socket_new(void);

```

17.15.5 atk-1.0/atk/atkwindow.h

```

#define ATK_TYPE_WINDOW (atk_window_get_type ())
#define ATK_WINDOW_GET_IFACE(obj) \
(G_TYPE_INSTANCE_GET_INTERFACE ((obj), ATK_TYPE_WINDOW, \
AtkWindowIface))
#define ATK_WINDOW(obj) G_TYPE_CHECK_INSTANCE_CAST ((obj), \
ATK_TYPE_WINDOW, AtkWindow)
#define ATK_IS_WINDOW(obj) G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
ATK_TYPE_WINDOW)

typedef struct _AtkWindow AtkWindow;
typedef struct _AtkWindowIface {
    GTypeInterface parent;
    gpointer _padding_dummy[16];
} AtkWindowIface;
extern GType atk_window_get_type(void);

```

17.16 Interface Definitions for libatk-1.0

The interfaces defined on the following pages are included in libatk-1.0 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 17.14 shall behave as described in the referenced base document.

17.17 Interfaces for libpango-1.0

Table 17-104 defines the library name and shared object name for the libpango-1.0 library

Table 17-104 libpango-1.0 Definition

Library:	libpango-1.0
SONAME:	libpango-1.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gobject 2.32] Gobject 2.32 Reference Manual

[Pango 1.30] Pango 1.30.1 Reference Manual

17.17.1 GTK Internationalized Text Layout and Rendering library

17.17.1.1 Interfaces for GTK Internationalized Text Layout and Rendering library

An LSB conforming implementation shall provide the generic functions for GTK Internationalized Text Layout and Rendering library specified in Table 17-105, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-105 libpango-1.0 - GTK Internationalized Text Layout and Rendering library Function Interfaces

pango_alignment_get_type [Gobject 2.32]	pango_attr_background_new [Pango 1.30]
pango_attr_fallback_new [Pango 1.30]	pango_attr_family_new [Pango 1.30]
pango_attr_font_desc_new [Pango 1.30]	pango_attr_foreground_new [Pango 1.30]
pango_attr_gravity_hint_new [Pango 1.30]	pango_attr_gravity_new [Pango 1.30]
pango_attr_iterator_copy [Pango 1.30]	pango_attr_iterator_destroy [Pango 1.30]
pango_attr_iterator_get [Pango 1.30]	pango_attr_iterator_get_attrs [Pango 1.30]
pango_attr_iterator_get_font [Pango 1.30]	pango_attr_iterator_next [Pango 1.30]
pango_attr_iterator_range [Pango 1.30]	pango_attr_language_new [Pango 1.30]
pango_attr_letter_spacing_new [Pango 1.30]	pango_attr_list_change [Pango 1.30]
pango_attr_list_copy [Pango 1.30]	pango_attr_list_filter [Pango 1.30]
pango_attr_list_get_iterator [Pango 1.30]	pango_attr_list_get_type [Gobject 2.32]
pango_attr_list_insert [Pango 1.30]	pango_attr_list_insert_before [Pango 1.30]
pango_attr_list_new [Pango 1.30]	pango_attr_list_ref [Pango 1.30]
pango_attr_list_splice [Pango 1.30]	pango_attr_list_unref [Pango 1.30]
pango_attr_rise_new [Pango 1.30]	pango_attr_scale_new [Pango 1.30]
pango_attr_shape_new [Pango 1.30]	pango_attr_shape_new_with_data [Pango 1.30]
pango_attr_size_new [Pango 1.30]	pango_attr_size_new_absolute [Pango 1.30]

pango_attr_stretch_new [Pango 1.30]	pango_attr_strikethrough_color_new [Pango 1.30]
pango_attr_strikethrough_new [Pango 1.30]	pango_attr_style_new [Pango 1.30]
pango_attr_type_get_name [Pango 1.30]	pango_attr_type_get_type [Gobject 2.32]
pango_attr_type_register [Pango 1.30]	pango_attr_underline_color_new [Pango 1.30]
pango_attr_underline_new [Pango 1.30]	pango_attr_variant_new [Pango 1.30]
pango_attr_weight_new [Pango 1.30]	pango_attribute_copy [Pango 1.30]
pango_attribute_destroy [Pango 1.30]	pango_attribute_equal [Pango 1.30]
pango_attribute_init [Pango 1.30]	pango_bidi_type_for_unichar [Pango 1.30]
pango_bidi_type_get_type [Gobject 2.32]	pango_break [Pango 1.30]
pango_color_copy [Pango 1.30]	pango_color_free [Pango 1.30]
pango_color_get_type [Gobject 2.32]	pango_color_parse [Pango 1.30]
pango_color_to_string [Pango 1.30]	pango_context_get_base_dir [Pango 1.30]
pango_context_get_base_gravity [Pango 1.30]	pango_context_get_font_description [Pango 1.30]
pango_context_get_font_map [Pango 1.30]	pango_context_get_gravity [Pango 1.30]
pango_context_get_gravity_hint [Pango 1.30]	pango_context_get_language [Pango 1.30]
pango_context_get_matrix [Pango 1.30]	pango_context_get_metrics [Pango 1.30]
pango_context_get_type [Gobject 2.32]	pango_context_list_families [Pango 1.30]
pango_context_load_font [Pango 1.30]	pango_context_load_fontset [Pango 1.30]
pango_context_new [Pango 1.30]	pango_context_set_base_dir [Pango 1.30]
pango_context_set_base_gravity [Pango 1.30]	pango_context_set_font_description [Pango 1.30]
pango_context_set_font_map [Pango 1.30]	pango_context_set_gravity_hint [Pango 1.30]
pango_context_set_language [Pango 1.30]	pango_context_set_matrix [Pango 1.30]

pango_coverage_copy [Pango 1.30]	pango_coverage_from_bytes [Pango 1.30]
pango_coverage_get [Pango 1.30]	pango_coverage_level_get_type [Gobject 2.32]
pango_coverage_max [Pango 1.30]	pango_coverage_new [Pango 1.30]
pango_coverage_ref [Pango 1.30]	pango_coverage_set [Pango 1.30]
pango_coverage_to_bytes [Pango 1.30]	pango_coverage_unref [Pango 1.30]
pango_direction_get_type [Gobject 2.32]	pango_ellipsize_mode_get_type [Gobject 2.32]
pango_extents_to_pixels [Pango 1.30]	pango_find_base_dir [Pango 1.30]
pango_find_paragraph_boundary [Pango 1.30]	pango_font_describe [Pango 1.30]
pango_font_describe_with_absolute_size [Pango 1.30]	pango_font_description_better_match [Pango 1.30]
pango_font_description_copy [Pango 1.30]	pango_font_description_copy_static [Pango 1.30]
pango_font_description_equal [Pango 1.30]	pango_font_description_free [Pango 1.30]
pango_font_description_from_string [Pango 1.30]	pango_font_description_get_family [Pango 1.30]
pango_font_description_get_gravity [Pango 1.30]	pango_font_description_get_set_fields [Pango 1.30]
pango_font_description_get_size [Pango 1.30]	pango_font_description_get_size_is_absolute [Pango 1.30]
pango_font_description_get_stretch [Pango 1.30]	pango_font_description_get_style [Pango 1.30]
pango_font_description_get_type [Gobject 2.32]	pango_font_description_get_variant [Pango 1.30]
pango_font_description_get_weight [Pango 1.30]	pango_font_description_hash [Pango 1.30]
pango_font_description_merge [Pango 1.30]	pango_font_description_merge_static [Pango 1.30]
pango_font_description_new [Pango 1.30]	pango_font_description_set_absolute_size [Pango 1.30]
pango_font_description_set_family [Pango 1.30]	pango_font_description_set_family_static [Pango 1.30]
pango_font_description_set_gravity [Pango 1.30]	pango_font_description_set_size [Pango 1.30]
pango_font_description_set_stretch [Pango 1.30]	pango_font_description_set_style [Pango 1.30]

pango_font_description_set_variant [Pango 1.30]	pango_font_description_set_weight [Pango 1.30]
pango_font_description_to_filename [Pango 1.30]	pango_font_description_to_string [Pango 1.30]
pango_font_description_unset_fields [Pango 1.30]	pango_font_descriptions_free [Pango 1.30]
pango_font_face_describe [Pango 1.30]	pango_font_face_get_face_name [Pango 1.30]
pango_font_face_get_type [Gobject 2.32]	pango_font_face_is_synthesized [Pango 1.30]
pango_font_face_list_sizes [Pango 1.30]	pango_font_family_get_name [Pango 1.30]
pango_font_family_get_type [Gobject 2.32]	pango_font_family_is_monospace [Pango 1.30]
pango_font_family_list_faces [Pango 1.30]	pango_font_find_shaper [Pango 1.30]
pango_font_get_coverage [Pango 1.30]	pango_font_get_font_map [Pango 1.30]
pango_font_get_glyph_extents [Pango 1.30]	pango_font_get_metrics [Pango 1.30]
pango_font_get_type [Gobject 2.32]	pango_font_map_create_context [Pango 1.30]
pango_font_map_get_type [Gobject 2.32]	pango_font_map_list_families [Pango 1.30]
pango_font_map_load_font [Pango 1.30]	pango_font_map_load_fontset [Pango 1.30]
pango_font_mask_get_type [Gobject 2.32]	pango_font_metrics_get_approximate_char_width [Pango 1.30]
pango_font_metrics_get_approximate_digit_width [Pango 1.30]	pango_font_metrics_get_ascent [Pango 1.30]
pango_font_metrics_get_descent [Pango 1.30]	pango_font_metrics_get_strikethrough_position [Pango 1.30]
pango_font_metrics_get_strikethrough_thickness [Pango 1.30]	pango_font_metrics_get_type [Gobject 2.32]
pango_font_metrics_get_underline_position [Pango 1.30]	pango_font_metrics_get_underline_thickness [Pango 1.30]
pango_font_metrics_ref [Pango 1.30]	pango_font_metrics_unref [Pango 1.30]
pango_fontset_foreach [Pango 1.30]	pango_fontset_get_font [Pango 1.30]
pango_fontset_get_metrics [Pango 1.30]	pango_fontset_get_type [Gobject 2.32]

pango_get_log_attrs [Pango 1.30]	pango_glyph_item_apply_attrs [Pango 1.30]
pango_glyph_item_copy [Pango 1.30]	pango_glyph_item_free [Pango 1.30]
pango_glyph_item_get_logical_widths [Pango 1.30]	pango_glyph_item_get_type [Gobject 2.32]
pango_glyph_item_iter_copy [Pango 1.30]	pango_glyph_item_iter_free [Pango 1.30]
pango_glyph_item_iter_get_type [Gobject 2.32]	pango_glyph_item_iter_init_end [Pango 1.30]
pango_glyph_item_iter_init_start [Pango 1.30]	pango_glyph_item_iter_next_cluster [Pango 1.30]
pango_glyph_item_iter_prev_cluster [Pango 1.30]	pango_glyph_item_letter_space [Pango 1.30]
pango_glyph_item_split [Pango 1.30]	pango_glyph_string_copy [Pango 1.30]
pango_glyph_string_extents [Pango 1.30]	pango_glyph_string_extents_range [Pango 1.30]
pango_glyph_string_free [Pango 1.30]	pango_glyph_string_get_logical_widths [Pango 1.30]
pango_glyph_string_get_type [Gobject 2.32]	pango_glyph_string_get_width [Pango 1.30]
pango_glyph_string_index_to_x [Pango 1.30]	pango_glyph_string_new [Pango 1.30]
pango_glyph_string_set_size [Pango 1.30]	pango_glyph_string_x_to_index [Pango 1.30]
pango_gravity_get_for_matrix [Pango 1.30]	pango_gravity_get_for_script [Pango 1.30]
pango_gravity_get_for_script_and_width [Pango 1.30]	pango_gravity_get_type [Gobject 2.32]
pango_gravity_hint_get_type [Gobject 2.32]	pango_gravity_to_rotation [Pango 1.30]
pango_is_zero_width [Pango 1.30]	pango_item_copy [Pango 1.30]
pango_item_free [Pango 1.30]	pango_item_get_type [Gobject 2.32]
pango_item_new [Pango 1.30]	pango_item_split [Pango 1.30]
pango_itemize [Pango 1.30]	pango_itemize_with_base_dir [Pango 1.30]
pango_language_from_string [Pango 1.30]	pango_language_get_default [Pango 1.30]
pango_language_get_sample_string [Pango 1.30]	pango_language_get_scripts [Pango 1.30]

pango_language_get_type [Gobject 2.32]	pango_language_includes_script [Pango 1.30]
pango_language_matches [Pango 1.30]	pango_language_to_string [Pango 1.30]
pango_layout_context_changed [Pango 1.30]	pango_layout_copy [Pango 1.30]
pango_layout_get_alignment [Pango 1.30]	pango_layout_get_attributes [Pango 1.30]
pango_layout_get_auto_dir [Pango 1.30]	pango_layout_get_baseline [Pango 1.30]
pango_layout_get_character_count [Pango 1.30]	pango_layout_get_context [Pango 1.30]
pango_layout_get_cursor_pos [Pango 1.30]	pango_layout_get_ellipsize [Pango 1.30]
pango_layout_get_extents [Pango 1.30]	pango_layout_get_font_description [Pango 1.30]
pango_layout_get_height [Pango 1.30]	pango_layout_get_indent [Pango 1.30]
pango_layout_get_iter [Pango 1.30]	pango_layout_get_justify [Pango 1.30]
pango_layout_get_line [Pango 1.30]	pango_layout_get_line_count [Pango 1.30]
pango_layout_get_line_readonly [Pango 1.30]	pango_layout_get_lines [Pango 1.30]
pango_layout_get_lines_readonly [Pango 1.30]	pango_layout_get_log_attrs [Pango 1.30]
pango_layout_get_log_attrs_readonly [Pango 1.30]	pango_layout_get_pixel_extents [Pango 1.30]
pango_layout_get_pixel_size [Pango 1.30]	pango_layout_get_single_paragraph_mode [Pango 1.30]
pango_layout_get_size [Pango 1.30]	pango_layout_get_spacing [Pango 1.30]
pango_layout_get_tabs [Pango 1.30]	pango_layout_get_text [Pango 1.30]
pango_layout_get_type [Gobject 2.32]	pango_layout_get_unknown_glyphs_count [Pango 1.30]
pango_layout_get_width [Pango 1.30]	pango_layout_get_wrap [Pango 1.30]
pango_layout_index_to_line_x [Pango 1.30]	pango_layout_index_to_pos [Pango 1.30]
pango_layout_is_ellipsized [Pango 1.30]	pango_layout_is_wrapped [Pango 1.30]

pango_layout_iter_at_last_line [Pango 1.30]	pango_layout_iter_copy [Pango 1.30]
pango_layout_iter_free [Pango 1.30]	pango_layout_iter_get_baseline [Pango 1.30]
pango_layout_iter_get_char_extents [Pango 1.30]	pango_layout_iter_get_cluster_extents [Pango 1.30]
pango_layout_iter_get_index [Pango 1.30]	pango_layout_iter_get_layout [Pango 1.30]
pango_layout_iter_get_layout_extents [Pango 1.30]	pango_layout_iter_get_line [Pango 1.30]
pango_layout_iter_get_line_extents [Pango 1.30]	pango_layout_iter_get_line_readonly [Pango 1.30]
pango_layout_iter_get_line_yrange [Pango 1.30]	pango_layout_iter_get_run [Pango 1.30]
pango_layout_iter_get_run_extents [Pango 1.30]	pango_layout_iter_get_run_readonly [Pango 1.30]
pango_layout_iter_get_type [Gobject 2.32]	pango_layout_iter_next_char [Pango 1.30]
pango_layout_iter_next_cluster [Pango 1.30]	pango_layout_iter_next_line [Pango 1.30]
pango_layout_iter_next_run [Pango 1.30]	pango_layout_line_get_extents [Pango 1.30]
pango_layout_line_get_pixel_extents [Pango 1.30]	pango_layout_line_get_type [Gobject 2.32]
pango_layout_line_get_x_ranges [Pango 1.30]	pango_layout_line_index_to_x [Pango 1.30]
pango_layout_line_ref [Pango 1.30]	pango_layout_line_unref [Pango 1.30]
pango_layout_line_x_to_index [Pango 1.30]	pango_layout_move_cursor_visually [Pango 1.30]
pango_layout_new [Pango 1.30]	pango_layout_set_alignment [Pango 1.30]
pango_layout_set_attributes [Pango 1.30]	pango_layout_set_auto_dir [Pango 1.30]
pango_layout_set_ellipsize [Pango 1.30]	pango_layout_set_font_description [Pango 1.30]
pango_layout_set_height [Pango 1.30]	pango_layout_set_indent [Pango 1.30]
pango_layout_set_justify [Pango 1.30]	pango_layout_set_markup [Pango 1.30]
pango_layout_set_markup_with_accel [Pango 1.30]	pango_layout_set_single_paragraph_mode [Pango 1.30]

pango_layout_set_spacing [Pango 1.30]	pango_layout_set_tabs [Pango 1.30]
pango_layout_set_text [Pango 1.30]	pango_layout_set_width [Pango 1.30]
pango_layout_set_wrap [Pango 1.30]	pango_layout_xy_to_index [Pango 1.30]
pango_log2vis_get_embedding_levels [Pango 1.30]	pango_matrix_concat [Pango 1.30]
pango_matrix_copy [Pango 1.30]	pango_matrix_free [Pango 1.30]
pango_matrix_get_font_scale_factor [Pango 1.30]	pango_matrix_get_type [Gobject 2.32]
pango_matrix_rotate [Pango 1.30]	pango_matrix_scale [Pango 1.30]
pango_matrix_transform_distance [Pango 1.30]	pango_matrix_transform_pixel_rectangle [Pango 1.30]
pango_matrix_transform_point [Pango 1.30]	pango_matrix_transform_rectangle [Pango 1.30]
pango_matrix_translate [Pango 1.30]	pango_parse_enum [Pango 1.30]
pango_parse_markup [Pango 1.30]	pango_parse_stretch [Pango 1.30]
pango_parse_style [Pango 1.30]	pango_parse_variant [Pango 1.30]
pango_parse_weight [Pango 1.30]	pango_quantize_line_geometry [Pango 1.30]
pango_read_line [Pango 1.30]	pango_render_part_get_type [Gobject 2.32]
pango_renderer_activate [Pango 1.30]	pango_renderer_deactivate [Pango 1.30]
pango_renderer_draw_error_underline [Pango 1.30]	pango_renderer_draw_glyph [Pango 1.30]
pango_renderer_draw_glyph_item [Pango 1.30]	pango_renderer_draw_glyphs [Pango 1.30]
pango_renderer_draw_layout [Pango 1.30]	pango_renderer_draw_layout_line [Pango 1.30]
pango_renderer_draw_rectangle [Pango 1.30]	pango_renderer_draw_trapezoid [Pango 1.30]
pango_renderer_get_color [Pango 1.30]	pango_renderer_get_layout [Pango 1.30]
pango_renderer_get_layout_line [Pango 1.30]	pango_renderer_get_matrix [Pango 1.30]
pango_renderer_get_type [Gobject 2.32]	pango_renderer_part_changed [Pango 1.30]
pango_renderer_set_color [Pango 1.30]	pango_renderer_set_matrix [Pango 1.30]

pango_reorder_items [Pango 1.30]	pango_scan_int [Pango 1.30]
pango_scan_string [Pango 1.30]	pango_scan_word [Pango 1.30]
pango_script_for_unichar [Pango 1.30]	pango_script_get_sample_language [Pango 1.30]
pango_script_get_type [Gobject 2.32]	pango_script_iter_free [Pango 1.30]
pango_script_iter_get_range [Pango 1.30]	pango_script_iter_new [Pango 1.30]
pango_script_iter_next [Pango 1.30]	pango_shape [Pango 1.30]
pango_skip_space [Pango 1.30]	pango_split_file_list [Pango 1.30]
pango_stretch_get_type [Gobject 2.32]	pango_style_get_type [Gobject 2.32]
pango_tab_align_get_type [Gobject 2.32]	pango_tab_array_copy [Pango 1.30]
pango_tab_array_free [Pango 1.30]	pango_tab_array_get_positions_in_pixels [Pango 1.30]
pango_tab_array_get_size [Pango 1.30]	pango_tab_array_get_tab [Pango 1.30]
pango_tab_array_get_tabs [Pango 1.30]	pango_tab_array_get_type [Gobject 2.32]
pango_tab_array_new [Pango 1.30]	pango_tab_array_new_with_positions [Pango 1.30]
pango_tab_array_resize [Pango 1.30]	pango_tab_array_set_tab [Pango 1.30]
pango_trim_string [Pango 1.30]	pango_underline_get_type [Gobject 2.32]
pango_unichar_direction [Pango 1.30]	pango_units_from_double [Pango 1.30]
pango_units_to_double [Pango 1.30]	pango_variant_get_type [Gobject 2.32]
pango_version [Pango 1.30]	pango_version_check [Pango 1.30]
pango_version_string [Pango 1.30]	pango_weight_get_type [Gobject 2.32]
pango_wrap_mode_get_type [Gobject 2.32]	

17.18 Data Definitions for libpango-1.0

This section defines global identifiers and their values that are associated with interfaces contained in libpango-1.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.18.1 pango-1.0/pango/pango-bidi-type.h

```
typedef enum {
    PANGO_BIDI_TYPE_L,
    PANGO_BIDI_TYPE_LRE,
    PANGO_BIDI_TYPE_LRO,
    PANGO_BIDI_TYPE_R,
    PANGO_BIDI_TYPE_AL,
    PANGO_BIDI_TYPE_RLE,
    PANGO_BIDI_TYPE_RLO,
    PANGO_BIDI_TYPE_PDF,
    PANGO_BIDI_TYPE_EN,
    PANGO_BIDI_TYPE_ES,
    PANGO_BIDI_TYPE_ET,
    PANGO_BIDI_TYPE_AN,
    PANGO_BIDI_TYPE_CS,
    PANGO_BIDI_TYPE_NSM,
    PANGO_BIDI_TYPE_BN,
    PANGO_BIDI_TYPE_B,
    PANGO_BIDI_TYPE_S,
    PANGO_BIDI_TYPE_WS,
    PANGO_BIDI_TYPE_ON
} PangoBidiType;
extern PangoBidiType pango_bidi_type_for_unichar(gunichar ch);
```

17.18.2 pango-1.0/pango/pango-utils.h

```
#define PANGO_VERSION_ENCODE(major,minor,micro) ( ((major) * 10000)
+ ((minor) * 100) + ((micro) * 1))
#define PANGO_VERSION_CHECK(major,minor,micro) (PANGO_VERSION >=
PANGO_VERSION_ENCODE(major,minor,micro))
#define PANGO_VERSION PANGO_VERSION_ENCODE( PANGO_VERSION_MAJOR,
PANGO_VERSION_MINOR, PANGO_VERSION_MICRO)

extern gboolean pango_is_zero_width(gunichar ch);
extern const char *pango_language_get_sample_string(PangoLanguage
*
language);
extern guint8 *pango_log2vis_get_embedding_levels(const gchar *
text,
int length,
PangoDirection *
pbase_dir);
extern gboolean pango_parse_enum(GType type, const char *str, int
*value,
gboolean warn, char **possible_values);
extern gboolean pango_parse_stretch(const char *str,
PangoStretch * stretch, gboolean
warn);
extern gboolean pango_parse_style(const char *str, PangoStyle *
style,
gboolean warn);
```

```

extern gboolean pango_parse_variant(const char *str,
                                   PangoVariant * variant, gboolean
warn);
extern gboolean pango_parse_weight(const char *str, PangoWeight *
weight,
                                   gboolean warn);
extern void pango_quantize_line_geometry(int *thickness, int
*position);
extern gint pango_read_line(FILE * stream, GString * str);
extern gboolean pango_scan_int(const char **pos, int *out);
extern gboolean pango_scan_string(const char **pos, GString * out);
extern gboolean pango_scan_word(const char **pos, GString * out);
extern gboolean pango_skip_space(const char **pos);
extern char **pango_split_file_list(const char *str);
extern char *pango_trim_string(const char *str);
extern int pango_version(void);
extern const char *pango_version_check(int required_major,
                                       int required_minor,
                                       int required_micro);
extern const char *pango_version_string(void);

```

17.18.3 pango-1.0/pango/pango.h

```

#define PANGO_CONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), PANGO_TYPE_CONTEXT, \
    PangoContextClass))
#define PANGO_LAYOUT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), PANGO_TYPE_LAYOUT, \
    PangoLayoutClass))
#define PANGO_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), PANGO_TYPE_RENDERER, \
    PangoRendererClass))
#define PANGO_IS_CONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), PANGO_TYPE_CONTEXT))
#define PANGO_IS_LAYOUT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), PANGO_TYPE_LAYOUT))
#define PANGO_IS_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), PANGO_TYPE_RENDERER))
#define PANGO_CONTEXT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_CONTEXT, \
    \
    PangoContext))
#define PANGO_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_FONT, \
    PangoFont))
#define PANGO_FONTSET(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_FONTSET, \
    \
    PangoFontset))
#define PANGO_FONT_FACE(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_FONT_FACE, \
    \
    PangoFontFace))
#define PANGO_FONT_FAMILY(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_FONT_FAMILY, \
    PangoFontFamily))
#define PANGO_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_FONT_MAP, \
    \
    PangoFontMap))
#define PANGO_LAYOUT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_LAYOUT, \
    PangoLayout))
#define PANGO_RENDERER(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_RENDERER, \
    PangoRenderer))

```

```

        (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_RENDERER,
\
        PangoRenderer))
#define PANGO_IS_CONTEXT(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_CONTEXT))
#define PANGO_IS_FONT(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_FONT))
#define PANGO_IS_FONTSET(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_FONTSET))
#define PANGO_IS_FONT_FACE(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object),
PANGO_TYPE_FONT_FACE))
#define PANGO_IS_FONT_FAMILY(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object),
PANGO_TYPE_FONT_FAMILY))
#define PANGO_IS_FONT_MAP(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_FONT_MAP))
#define PANGO_IS_LAYOUT(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_LAYOUT))
#define PANGO_IS_RENDERER(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_RENDERER))
#define PANGO_CONTEXT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), PANGO_TYPE_CONTEXT, \
        PangoContextClass))
#define PANGO_LAYOUT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), PANGO_TYPE_LAYOUT, \
        PangoLayoutClass))
#define PANGO_RENDERER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), PANGO_TYPE_RENDERER, \
        PangoRendererClass))
#define PANGO_UNITS_ROUND(d) ((d) + (PANGO_SCALE >> 1)) &
~(PANGO_SCALE - 1))
#define PANGO_PIXELS_CEIL(d) (((int)(d) + 1023) >> 10)
#define PANGO_PIXELS(d) (((int)(d) + 512) >> 10)
#define PANGO_PIXELS_FLOOR(d) (((int)(d)) >> 10)
#define pango_language_to_string(language) ((const char
*)language)
#define PANGO_SCALE_XX_SMALL ((double)0.5787037037037)
#define PANGO_SCALE_X_SMALL ((double)0.64444444444444)
#define PANGO_SCALE_SMALL ((double)0.83333333333333)
#define PANGO_SCALE_MEDIUM ((double)1.0)
#define PANGO_SCALE_LARGE ((double)1.2)
#define PANGO_SCALE_X_LARGE ((double)1.43999999999999)
#define PANGO_SCALE_XX_LARGE ((double)1.728)
#define PANGO_GRAVITY_IS_VERTICAL(gravity) ((gravity) ==
PANGO_GRAVITY_EAST || (gravity) == PANGO_GRAVITY_WEST)
#if defined(PANGO_ENABLE_ENGINE) || defined(PANGO_ENABLE_BACKEND)
#define PANGO_GET_UNKNOWN_GLYPH(wc)
((PangoGlyph)(wc) | PANGO_GLYPH_UNKNOWN_FLAG)
#endif
#define PANGO_GLYPH_EMPTY ((PangoGlyph)0xFFFFFFFF)
#if defined(PANGO_ENABLE_ENGINE) || defined(PANGO_ENABLE_BACKEND)
#define PANGO_GLYPH_UNKNOWN_FLAG ((PangoGlyph)0x10000000)
#endif
#define PANGO_GLYPH_INVALID_INPUT ((PangoGlyph)0xFFFFFFFF)
#define PANGO_RBEARING(rect) ((rect).x + (rect).width)
#define PANGO_LBEARING(rect) ((rect).x)
#define PANGO_DESCENT(rect) ((rect).y + (rect).height)
#define PANGO_ASCENT(rect) (-(rect).y)
#define PANGO_ANALYSIS_FLAG_CENTERED_BASELINE (1 << 0)
#define PANGO_TYPE_ALIGNMENT (pango_alignment_get_type())
#define PANGO_TYPE_ATTR_TYPE (pango_attr_type_get_type())
#define PANGO_TYPE_BIDI_TYPE (pango_bidi_type_get_type())
#define PANGO_TYPE_CONTEXT (pango_context_get_type())
#define PANGO_TYPE_COVERAGE_LEVEL
(pango_coverage_level_get_type())

```



```

#define PANGO_TYPE_DIRECTION      (pango_direction_get_type())
#define PANGO_TYPE_ELLIPSIZE_MODE (pango_ellipsize_mode_get_type())
#define PANGO_TYPE_FONTSET        (pango_fontset_get_type ())
#define PANGO_TYPE_FONT_DESCRIPTION (pango_font_description_get_type ())
#define PANGO_TYPE_FONT_FACE      (pango_font_face_get_type ())
#define PANGO_TYPE_FONT_FAMILY    (pango_font_family_get_type ())
#define PANGO_TYPE_FONT            (pango_font_get_type ())
#define PANGO_TYPE_FONT_MAP        (pango_font_map_get_type ())
#define PANGO_TYPE_FONT_MASK       (pango_font_mask_get_type())
#define PANGO_TYPE_FONT_METRICS    (pango_font_metrics_get_type ())
#define PANGO_TYPE_GLYPH_ITEM      (pango_glyph_item_get_type ())
#define PANGO_TYPE_GLYPH_ITEM_ITER (pango_glyph_item_iter_get_type ())
#define PANGO_TYPE_GLYPH_STRING    (pango_glyph_string_get_type ())
#define PANGO_TYPE_GRAVITY          (pango_gravity_get_type())
#define PANGO_TYPE_GRAVITY_HINT    (pango_gravity_hint_get_type())
#define PANGO_TYPE_ITEM             (pango_item_get_type ())
#define PANGO_TYPE_LANGUAGE         (pango_language_get_type ())
#define PANGO_TYPE_LAYOUT           (pango_layout_get_type ())
#define PANGO_TYPE_LAYOUT_ITER      (pango_layout_iter_get_type ())
#define PANGO_TYPE_LAYOUT_LINE      (pango_layout_line_get_type ())
#define PANGO_TYPE_MATRIX           (pango_matrix_get_type ())
#define PANGO_TYPE_RENDERER         (pango_renderer_get_type())
#define PANGO_TYPE_RENDER_PART      (pango_render_part_get_type())
#define PANGO_TYPE_SCRIPT           (pango_script_get_type())
#define PANGO_TYPE_STRETCH          (pango_stretch_get_type())
#define PANGO_TYPE_STYLE            (pango_style_get_type())
#define PANGO_TYPE_TAB_ALIGN        (pango_tab_align_get_type())
#define PANGO_TYPE_TAB_ARRAY        (pango_tab_array_get_type ())
#define PANGO_TYPE_UNDERLINE        (pango_underline_get_type())
#define PANGO_TYPE_VARIANT          (pango_variant_get_type())
#define PANGO_TYPE_WEIGHT           (pango_weight_get_type())
#define PANGO_TYPE_WRAP_MODE        (pango_wrap_mode_get_type())
#define PANGO_ATTR_INDEX_FROM_TEXT_BEGINNING 0
#define PANGO_VERSION_MAJOR         1
#define PANGO_VERSION_MICRO         1
#define PANGO_VERSION_STRING        "1.30.1"
#define PANGO_SCALE                  1024
#define PANGO_VERSION_MINOR         30
#define PANGO_ATTR_INDEX_TO_TEXT_END      G_MAXUINT
#define PANGO_TYPE_ATTR_LIST           pango_attr_list_get_type ()
#define PANGO_TYPE_COLOR               pango_color_get_type ()
#define PANGO_MATRIX_INIT              { 1., 0., 0., 1., 0., 0. }

typedef struct _PangoFontFace PangoFontFace;
typedef enum {
    PANGO_WRAP_WORD = 0,
    PANGO_WRAP_CHAR = 1,
    PANGO_WRAP_WORD_CHAR = 2
} PangoWrapMode;
typedef struct _PangoLayout PangoLayout;
typedef struct _PangoScriptIter PangoScriptIter;
typedef enum {
    PANGO_SCRIPT_INVALID_CODE = -1,
    PANGO_SCRIPT_COMMON = 0,
    PANGO_SCRIPT_INHERITED = 1,
    PANGO_SCRIPT_ARABIC = 2,
    PANGO_SCRIPT_ARMENIAN = 3,
    PANGO_SCRIPT_BENGALI = 4,
    PANGO_SCRIPT_BOPOMOFO = 5,
    PANGO_SCRIPT_CHEROKEE = 6,
    PANGO_SCRIPT_COPTIC = 7,
    PANGO_SCRIPT_CYRILLIC = 8,
    PANGO_SCRIPT_DESERET = 9,

```



```

PANGO_SCRIPT_DEVANAGARI = 10,
PANGO_SCRIPT_ETHIOPIA = 11,
PANGO_SCRIPT_GEORGIAN = 12,
PANGO_SCRIPT_GOTHIC = 13,
PANGO_SCRIPT_GREEK = 14,
PANGO_SCRIPT_GUJARATI = 15,
PANGO_SCRIPT_GURMUKHI = 16,
PANGO_SCRIPT_HAN = 17,
PANGO_SCRIPT_HANGUL = 18,
PANGO_SCRIPT_HEBREW = 19,
PANGO_SCRIPT_HIRAGANA = 20,
PANGO_SCRIPT_KANNADA = 21,
PANGO_SCRIPT_KATAKANA = 22,
PANGO_SCRIPT_KHMER = 23,
PANGO_SCRIPT_LAO = 24,
PANGO_SCRIPT_LATIN = 25,
PANGO_SCRIPT_MALAYALAM = 26,
PANGO_SCRIPT_MONGOLIAN = 27,
PANGO_SCRIPT_MYANMAR = 28,
PANGO_SCRIPT_OGHAM = 29,
PANGO_SCRIPT_OLD_ITALIC = 30,
PANGO_SCRIPT_ORIYA = 31,
PANGO_SCRIPT_RUNIC = 32,
PANGO_SCRIPT_SINHALA = 33,
PANGO_SCRIPT_SYRIAC = 34,
PANGO_SCRIPT_TAMIL = 35,
PANGO_SCRIPT_TELUGU = 36,
PANGO_SCRIPT_THAANA = 37,
PANGO_SCRIPT_THAI = 38,
PANGO_SCRIPT_TIBETAN = 39,
PANGO_SCRIPT_CANADIAN_ABORIGINAL = 40,
PANGO_SCRIPT_YI = 41,
PANGO_SCRIPT_TAGALOG = 42,
PANGO_SCRIPT_HANUNOO = 43,
PANGO_SCRIPT_BUHID = 44,
PANGO_SCRIPT_TAGBANWA = 45,
PANGO_SCRIPT_BRAILLE = 46,
PANGO_SCRIPT_CYPRIOT = 47,
PANGO_SCRIPT_LIMBU = 48,
PANGO_SCRIPT_ORMANYA = 49,
PANGO_SCRIPT_SHAVIAN = 50,
PANGO_SCRIPT_LINEAR_B = 51,
PANGO_SCRIPT_TAI_LE = 52,
PANGO_SCRIPT_UGARITIC = 53
} PangoScript;
typedef struct _PangoFont PangoFont;
typedef struct _PangoContext PangoContext;
typedef struct _PangoFontDescription PangoFontDescription;
typedef enum {
    PANGO_ATTR_INVALID = 0,
    PANGO_ATTR_LANGUAGE = 1,
    PANGO_ATTR_FAMILY = 2,
    PANGO_ATTR_STYLE = 3,
    PANGO_ATTR_WEIGHT = 4,
    PANGO_ATTR_VARIANT = 5,
    PANGO_ATTR_STRETCH = 6,
    PANGO_ATTR_SIZE = 7,
    PANGO_ATTR_FONT_DESC = 8,
    PANGO_ATTR_FOREGROUND = 9,
    PANGO_ATTR_BACKGROUND = 10,
    PANGO_ATTR_UNDERLINE = 11,
    PANGO_ATTR_STRIKETHROUGH = 12,
    PANGO_ATTR_RISE = 13,
    PANGO_ATTR_SHAPE = 14,
    PANGO_ATTR_SCALE = 15,
    PANGO_ATTR_FALLBACK = 16,

```

```

        PANGO_ATTR_LETTER_SPACING = 17,
        PANGO_ATTR_UNDERLINE_COLOR = 18,
        PANGO_ATTR_STRIKETHROUGH_COLOR = 19,
        PANGO_ATTR_ABSOLUTE_SIZE = 20
    } PangoAttrType;
typedef struct _PangoAttribute {
    const PangoAttrClass *klass;
    guint start_index;
    guint end_index;
} PangoAttribute;
typedef struct _PangoAttrClass {
    PangoAttrType type;
    PangoAttribute *(*copy) (const PangoAttribute *);
    void (*destroy) (PangoAttribute *);
    gboolean(*equal) (const PangoAttribute *, const PangoAttribute
*);
} PangoAttrClass;
typedef struct _PangoLanguage PangoLanguage;
typedef struct _PangoLogAttr {
    guint is_line_break:1;
    guint is_mandatory_break:1;
    guint is_char_break:1;
    guint is_white:1;
    guint is_cursor_position:1;
    guint is_word_start:1;
    guint is_word_end:1;
    guint is_sentence_boundary:1;
    guint is_sentence_start:1;
    guint is_sentence_end:1;
    guint backspace_deletes_character:1;
    unsigned int is_expandable_space:1;
    unsigned int is_word_boundary:1;
} PangoLogAttr;
typedef struct _PangoColor {
    guint16 red;
    guint16 green;
    guint16 blue;
} PangoColor;
typedef struct _PangoMatrix {
    double xx;
    double xy;
    double yx;
    double yy;
    double x0;
    double y0;
} PangoMatrix;
typedef struct _PangoEngineShape PangoEngineShape;
typedef struct _PangoEngineLang PangoEngineLang;
typedef struct _PangoAnalysis {
    PangoEngineShape *shape_engine;
    PangoEngineLang *lang_engine;
    PangoFont *font;
    guint8 level;
    PangoLanguage *language;
    GSList *extra_attrs;
} PangoAnalysis;
typedef struct _PangoItem {
    gint offset;
    gint length;
    gint num_chars;
    PangoAnalysis analysis;
} PangoItem;
typedef guint32 PangoGlyph;
typedef gint32 PangoGlyphUnit;
typedef struct _PangoGlyphGeometry {
    PangoGlyphUnit width;

```

```

    PangoGlyphUnit x_offset;
    PangoGlyphUnit y_offset;
} PangoGlyphGeometry;
typedef struct _PangoGlyphVisAttr {
    guint is_cluster_start:1;
} PangoGlyphVisAttr;
typedef struct _PangoGlyphInfo {
    PangoGlyph glyph;
    PangoGlyphGeometry geometry;
    PangoGlyphVisAttr attr;
} PangoGlyphInfo;
typedef struct _PangoGlyphString {
    gint num_glyphs;
    PangoGlyphInfo *glyphs;
    gint *log_clusters;
    gint space;
} PangoGlyphString;
typedef struct _PangoGlyphItem {
    PangoItem *item;
    PangoGlyphString *glyphs;
} PangoGlyphItem;
typedef PangoGlyphItem PangoLayoutRun;
typedef struct _PangoLayoutIter PangoLayoutIter;
typedef enum {
    PANGO_UNDERLINE_NONE = 0,
    PANGO_UNDERLINE_SINGLE = 1,
    PANGO_UNDERLINE_DOUBLE = 2,
    PANGO_UNDERLINE_LOW = 3,
    PANGO_UNDERLINE_ERROR = 4
} PangoUnderline;
typedef struct _PangoRendererPrivate PangoRendererPrivate;
typedef struct _PangoRenderer {
    GObject parent_instance;
    PangoUnderline underline;
    gboolean strikethrough;
    int active_count;
    PangoMatrix *matrix;
    PangoRendererPrivate *priv;
} PangoRenderer;
typedef enum {
    PANGO_RENDER_PART_FOREGROUND = 0,
    PANGO_RENDER_PART_BACKGROUND = 1,
    PANGO_RENDER_PART_UNDERLINE = 2,
    PANGO_RENDER_PART_STRIKETHROUGH = 3
} PangoRenderPart;
typedef struct _PangoAttrList PangoAttrList;
typedef struct _PangoLayoutLine {
    PangoLayout *layout;
    gint start_index;
    gint length;
    GSList *runs;
    guint is_paragraph_start:1;
    guint resolved_dir:3;
} PangoLayoutLine;
typedef enum {
    PANGO_STRETCH_ULTRA_CONDENSED = 0,
    PANGO_STRETCH_EXTRA_CONDENSED = 1,
    PANGO_STRETCH_CONDENSED = 2,
    PANGO_STRETCH_SEMI_CONDENSED = 3,
    PANGO_STRETCH_NORMAL = 4,
    PANGO_STRETCH_SEMI_EXPANDED = 5,
    PANGO_STRETCH_EXPANDED = 6,
    PANGO_STRETCH_EXTRA_EXPANDED = 7,
    PANGO_STRETCH_ULTRA_EXPANDED = 8
} PangoStretch;
typedef struct _PangoRectangle {

```

```

    int x;
    int y;
    int width;
    int height;
} PangoRectangle;
typedef struct _PangoFontFamily PangoFontFamily;
typedef struct _PangoFontMetrics PangoFontMetrics;
typedef struct _PangoTabArray PangoTabArray;
typedef enum {
    PANGO_TAB_LEFT = 0
} PangoTabAlign;
typedef enum {
    PANGO_ALIGN_LEFT = 0,
    PANGO_ALIGN_CENTER = 1,
    PANGO_ALIGN_RIGHT = 2
} PangoAlignment;
typedef struct _PangoAttrIterator PangoAttrIterator;
typedef enum {
    PANGO_FONT_MASK_FAMILY = 1,
    PANGO_FONT_MASK_STYLE = 2,
    PANGO_FONT_MASK_VARIANT = 4,
    PANGO_FONT_MASK_WEIGHT = 8,
    PANGO_FONT_MASK_STRETCH = 16,
    PANGO_FONT_MASK_SIZE = 32
} PangoFontMask;
typedef enum {
    PANGO_DIRECTION_LTR = 0,
    PANGO_DIRECTION_RTL = 1,
    PANGO_DIRECTION_TTB_LTR = 2,
    PANGO_DIRECTION_TTB_RTL = 3,
    PANGO_DIRECTION_WEAK_LTR = 4,
    PANGO_DIRECTION_WEAK_RTL = 5,
    PANGO_DIRECTION_NEUTRAL = 6
} PangoDirection;
typedef enum {
    PANGO_ELLIPSIZE_NONE = 0,
    PANGO_ELLIPSIZE_START = 1,
    PANGO_ELLIPSIZE_MIDDLE = 2,
    PANGO_ELLIPSIZE_END = 3
} PangoEllipsizeMode;
typedef struct _PangoCoverage PangoCoverage;
typedef enum {
    PANGO_STYLE_NORMAL = 0,
    PANGO_STYLE_OBLIQUE = 1,
    PANGO_STYLE_ITALIC = 2
} PangoStyle;
typedef enum {
    PANGO_COVERAGE_NONE = 0,
    PANGO_COVERAGE_FALLBACK = 1,
    PANGO_COVERAGE_APPROXIMATE = 2,
    PANGO_COVERAGE_EXACT = 3
} PangoCoverageLevel;
typedef struct _PangoFontMap PangoFontMap;
typedef gboolean(*PangoAttrFilterFunc) (PangoAttribute *,
gpointer);
typedef struct _PangoFontset PangoFontset;
typedef enum {
    PANGO_WEIGHT_ULTRALIGHT = 200,
    PANGO_WEIGHT_LIGHT = 300,
    PANGO_WEIGHT_NORMAL = 400,
    PANGO_WEIGHT_SEMIBOLD = 600,
    PANGO_WEIGHT_BOLD = 700,
    PANGO_WEIGHT_ULTRABOLD = 800,
    PANGO_WEIGHT_HEAVY = 900
} PangoWeight;

```

```

typedef    gboolean(*PangoFontsetForeachFunc)    (PangoFontset    *,
PangoFont *,
                                                    gpointer);

typedef enum {
    PANGO_VARIANT_NORMAL = 0,
    PANGO_VARIANT_SMALL_CAPS = 1
} PangoVariant;
typedef gpointer(*PangoAttrDataCopyFunc) (gconstpointer);
typedef struct _PangoAttrShape {
    PangoAttribute attr;
    PangoRectangle ink_rect;
    PangoRectangle logical_rect;
    gpointer data;
    PangoAttrDataCopyFunc copy_func;
    GDestroyNotify destroy_func;
} PangoAttrShape;
typedef struct _PangoContextClass PangoContextClass;
typedef struct _PangoAttrString {
    PangoAttribute attr;
    char *value;
} PangoAttrString;
typedef struct _PangoAttrColor {
    PangoAttribute attr;
    PangoColor color;
} PangoAttrColor;
typedef struct _PangoAttrFontDesc {
    PangoAttribute attr;
    PangoFontDescription *desc;
} PangoAttrFontDesc;
typedef struct _PangoAttrFloat {
    PangoAttribute attr;
    double value;
} PangoAttrFloat;
typedef struct _PangoRendererClass {
    GObjectClass parent_class;
    void    (*draw_glyphs)    (PangoRenderer    *,    PangoFont    *,
PangoGlyphString *,
                                int, int);
    void (*draw_rectangle) (PangoRenderer *, PangoRenderPart, int,
int,
                                int, int);
    void (*draw_error_underline) (PangoRenderer *, int, int, int,
int);
    void (*draw_shape) (PangoRenderer *, PangoAttrShape *, int, int);
    void (*draw_trapezoid) (PangoRenderer *, PangoRenderPart,
double,
                                double, double, double, double, double);
    void (*draw_glyph) (PangoRenderer *, PangoFont *, PangoGlyph,
double,
                                double);
    void (*part_changed) (PangoRenderer *, PangoRenderPart);
    void (*begin) (PangoRenderer *);
    void (*end) (PangoRenderer *);
    void (*prepare_run) (PangoRenderer *, PangoLayoutRun *);
    void (*draw_glyph_item) (void);
    void (*_pango_reserved2) (void);
    void (*_pango_reserved3) (void);
    void (*_pango_reserved4) (void);
} PangoRendererClass;
typedef struct _PangoAttrLanguage {
    PangoAttribute attr;
    PangoLanguage *value;
} PangoAttrLanguage;
typedef struct _PangoAttrInt {
    PangoAttribute attr;
    int value;

```

```

} PangoAttrInt;
typedef struct _PangoAttrSize {
    PangoAttribute attr;
    int size;
    guint absolute:1;
} PangoAttrSize;
typedef struct _PangoLayoutClass PangoLayoutClass;
typedef enum {
    PANGO_GRAVITY_SOUTH,
    PANGO_GRAVITY_EAST,
    PANGO_GRAVITY_NORTH,
    PANGO_GRAVITY_WEST,
    PANGO_GRAVITY_AUTO
} PangoGravity;
typedef enum {
    PANGO_GRAVITY_HINT_NATURAL,
    PANGO_GRAVITY_HINT_STRONG,
    PANGO_GRAVITY_HINT_LINE
} PangoGravityHint;
typedef struct _PangoGlyphItemIter {
    PangoGlyphItem *glyph_item;
    const gchar *text;
    int start_glyph;
    int start_index;
    int start_char;
    int end_glyph;
    int end_index;
    int end_char;
} PangoGlyphItemIter;
extern GType pango_alignment_get_type(void);
extern PangoAttribute *pango_attr_background_new(guint16 red,
                                                  guint16 green,
                                                  guint16 blue);
extern PangoAttribute *pango_attr_fallback_new(gboolean
enable_fallback);
extern PangoAttribute *pango_attr_family_new(const char *family);
extern PangoAttribute *pango_attr_font_desc_new(const
PangoFontDescription
* desc);
extern PangoAttribute *pango_attr_foreground_new(guint16 red,
                                                  guint16 green,
                                                  guint16 blue);
extern PangoAttribute *pango_attr_gravity_hint_new(PangoGravityHint hint);
extern PangoAttribute *pango_attr_gravity_new(PangoGravity
gravity);
extern PangoAttrIterator *pango_attr_iterator_copy(PangoAttrIterator *
iterator);
extern void pango_attr_iterator_destroy(PangoAttrIterator *
iterator);
extern PangoAttribute *pango_attr_iterator_get(PangoAttrIterator *
iterator,
PangoAttrType type);
extern GSList *pango_attr_iterator_get_attrs(PangoAttrIterator *
iterator);
extern void pango_attr_iterator_get_font(PangoAttrIterator *
iterator,
PangoFontDescription * desc,
PangoLanguage * *language,
GSList * *extra_attrs);
extern gboolean pango_attr_iterator_next(PangoAttrIterator *
iterator);
extern void pango_attr_iterator_range(PangoAttrIterator * iterator,
gint * start, gint * end);

```

```

extern PangoAttribute *pango_attr_language_new(PangoLanguage *
language);
extern PangoAttribute *pango_attr_letter_spacing_new(int
letter_spacing);
extern void pango_attr_list_change(PangoAttrList * list,
PangoAttribute * attr);
extern PangoAttrList *pango_attr_list_copy(PangoAttrList * list);
extern PangoAttrList *pango_attr_list_filter(PangoAttrList * list,
PangoAttrFilterFunc func,
gpointer data);
extern PangoAttrIterator
*pango_attr_list_get_iterator(PangoAttrList *
list);
extern GType pango_attr_list_get_type(void);
extern void pango_attr_list_insert(PangoAttrList * list,
PangoAttribute * attr);
extern void pango_attr_list_insert_before(PangoAttrList * list,
PangoAttribute * attr);
extern PangoAttrList *pango_attr_list_new(void);
extern PangoAttrList *pango_attr_list_ref(PangoAttrList * list);
extern void pango_attr_list_splice(PangoAttrList * list,
PangoAttrList * other, gint pos,
gint len);
extern void pango_attr_list_unref(PangoAttrList * list);
extern PangoAttribute *pango_attr_rise_new(int rise);
extern PangoAttribute *pango_attr_scale_new(double scale_factor);
extern PangoAttribute *pango_attr_shape_new(const PangoRectangle *
ink_rect,
const PangoRectangle *
logical_rect);
extern PangoAttribute *pango_attr_shape_new_with_data(const
PangoRectangle
* ink_rect,
const PangoRectangle
* logical_rect,
gpointer data,
PangoAttrDataCopyFunc
copy_func,
GDestroyNotify
destroy_func);
extern PangoAttribute *pango_attr_size_new(int size);
extern PangoAttribute *pango_attr_size_new_absolute(int size);
extern PangoAttribute *pango_attr_stretch_new(PangoStretch
stretch);
extern PangoAttribute *pango_attr_strikethrough_color_new(guint16
red,
guint16 green,
guint16 blue);
extern PangoAttribute *pango_attr_strikethrough_new(gboolean
strikethrough);
extern PangoAttribute *pango_attr_style_new(PangoStyle style);
extern const char *pango_attr_type_get_name(PangoAttrType type);
extern GType pango_attr_type_get_type(void);
extern PangoAttrType pango_attr_type_register(const gchar * name);
extern PangoAttribute *pango_attr_underline_color_new(guint16 red,
guint16 green,
guint16 blue);
extern PangoAttribute *pango_attr_underline_new(PangoUnderline
underline);
extern PangoAttribute *pango_attr_variant_new(PangoVariant
variant);
extern PangoAttribute *pango_attr_weight_new(PangoWeight weight);
extern PangoAttribute *pango_attr_copy(const PangoAttribute *
attr);
extern void pango_attr_destroy(PangoAttribute * attr);

```

```

extern gboolean pango_attribute_equal(const PangoAttribute * attr1,
                                     const PangoAttribute * attr2);
extern void pango_attribute_init(PangoAttribute * attr,
                                const PangoAttrClass * klass);
extern GType pango_bidi_type_get_type(void);
extern void pango_break(const gchar * text, gint length,
                       PangoAnalysis * analysis, PangoLogAttr * attrs,
                       int attrs_len);
extern PangoColor *pango_color_copy(const PangoColor * src);
extern void pango_color_free(PangoColor * color);
extern GType pango_color_get_type(void);
extern gboolean pango_color_parse(PangoColor * color, const char
*spec);
extern gchar *pango_color_to_string(const PangoColor * color);
extern PangoDirection pango_context_get_base_dir(PangoContext *
context);
extern PangoGravity pango_context_get_base_gravity(PangoContext *
context);
extern PangoFontDescription
    *pango_context_get_font_description(PangoContext * context);
extern PangoFontMap *pango_context_get_font_map(PangoContext *
context);
extern PangoGravity pango_context_get_gravity(PangoContext *
context);
extern PangoGravityHint
    pango_context_get_gravity_hint(PangoContext *
context);
extern PangoLanguage *pango_context_get_language(PangoContext *
context);
extern const PangoMatrix *pango_context_get_matrix(PangoContext *
context);
extern PangoFontMetrics *pango_context_get_metrics(PangoContext *
context,
                                                    const
                                                    PangoFontDescription *
                                                    desc,
                                                    PangoLanguage *
                                                    language);
extern GType pango_context_get_type(void);
extern void pango_context_list_families(PangoContext * context,
                                       PangoFontFamily * **families,
                                       int *n_families);
extern PangoFont *pango_context_load_font(PangoContext * context,
                                          const PangoFontDescription *
                                          desc);
extern PangoFontset *pango_context_load_fontset(PangoContext *
context,
                                                const PangoFontDescription
                                                * desc,
                                                PangoLanguage * language);
extern PangoContext *pango_context_new(void);
extern void pango_context_set_base_dir(PangoContext * context,
                                       PangoDirection direction);
extern void pango_context_set_base_gravity(PangoContext * context,
                                           PangoGravity gravity);
extern void pango_context_set_font_description(PangoContext *
context,
                                              const PangoFontDescription
                                              *
                                              desc);
extern void pango_context_set_font_map(PangoContext * context,
                                       PangoFontMap * font_map);
extern void pango_context_set_gravity_hint(PangoContext * context,
                                           PangoGravityHint hint);
extern void pango_context_set_language(PangoContext * context,
                                       PangoLanguage * language);

```



```

extern void pango_context_set_matrix(PangoContext * context,
                                     const PangoMatrix * matrix);
extern PangoCoverage *pango_coverage_copy(PangoCoverage *
coverage);
extern PangoCoverage *pango_coverage_from_bytes(guchar * bytes,
                                                int n_bytes);
extern PangoCoverageLevel pango_coverage_get(PangoCoverage *
coverage,
                                             int index);
extern GType pango_coverage_level_get_type(void);
extern void pango_coverage_max(PangoCoverage * coverage,
                               PangoCoverage * other);
extern PangoCoverage *pango_coverage_new(void);
extern PangoCoverage *pango_coverage_ref(PangoCoverage * coverage);
extern void pango_coverage_set(PangoCoverage * coverage, int index,
                              PangoCoverageLevel level);
extern void pango_coverage_to_bytes(PangoCoverage * coverage,
                                    guchar * *bytes, int *n_bytes);
extern void pango_coverage_unref(PangoCoverage * coverage);
extern GType pango_direction_get_type(void);
extern GType pango_ellipsize_mode_get_type(void);
extern void pango_extents_to_pixels(PangoRectangle * inclusive,
                                    PangoRectangle * nearest);
extern PangoDirection pango_find_base_dir(const gchar * text, gint
length);
extern void pango_find_paragraph_boundary(const gchar * text, gint
length,
                                         gint *
paragraph_delimiter_index,
                                         gint * next_paragraph_start);
extern PangoFontDescription *pango_font_describe(PangoFont * font);
extern PangoFontDescription
    *pango_font_describe_with_absolute_size(PangoFont * font);
extern gboolean pango_font_description_better_match(const
PangoFontDescription *
desc,
const
PangoFontDescription *
old_match,
const
PangoFontDescription *
new_match);
extern PangoFontDescription *pango_font_description_copy(const
PangoFontDescription
* desc);
extern PangoFontDescription
    *pango_font_description_copy_static(const
PangoFontDescription
* desc);
extern gboolean pango_font_description_equal(const
PangoFontDescription *
desc1,
const PangoFontDescription *
desc2);
extern void pango_font_description_free(PangoFontDescription *
desc);
extern PangoFontDescription
    *pango_font_description_from_string(const char
*str);
extern const char *pango_font_description_get_family(const
PangoFontDescription
*
desc);
extern PangoGravity pango_font_description_get_gravity(const

```

```

PangoFontDescription
                                * desc);
extern PangoFontMask pango_font_description_get_set_fields(const
PangoFontDescription
                                * desc);
extern          gint          pango_font_description_get_size(const
PangoFontDescription *
                                desc);
extern gboolean pango_font_description_get_size_is_absolute(const
PangoFontDescription
                                * desc);
extern PangoStretch pango_font_description_get_stretch(const
PangoFontDescription
                                * desc);
extern PangoStyle pango_font_description_get_style(const
                                PangoFontDescription *
                                desc);
extern GType pango_font_description_get_type(void);
extern PangoVariant pango_font_description_get_variant(const
PangoFontDescription
                                * desc);
extern PangoWeight pango_font_description_get_weight(const
                                PangoFontDescription
                                *
                                desc);
extern          guint          pango_font_description_hash(const
PangoFontDescription *
                                desc);
extern void pango_font_description_merge(PangoFontDescription *
desc,
                                const PangoFontDescription *
                                desc_to_merge,
                                gboolean replace_existing);
extern
                                void
pango_font_description_merge_static(PangoFontDescription *
                                desc,
                                const PangoFontDescription
                                * desc_to_merge,
                                gboolean replace_existing);
extern PangoFontDescription *pango_font_description_new(void);
extern
                                void
pango_font_description_set_absolute_size(PangoFontDescription *
                                desc, double size);
extern void pango_font_description_set_family(PangoFontDescription
* desc,
                                const char *family);
extern
                                void
pango_font_description_set_family_static(PangoFontDescription *
                                desc,
                                const char *family);
extern
                                void
pango_font_description_set_gravity(PangoFontDescription * desc,
                                PangoGravity gravity);
extern void pango_font_description_set_size(PangoFontDescription *
desc,
                                gint size);
extern
                                void
pango_font_description_set_stretch(PangoFontDescription * desc,
                                PangoStretch stretch);
extern void pango_font_description_set_style(PangoFontDescription
* desc,

```

```

        PangoStyle style);
extern void pango_font_description_set_variant(PangoFontDescription * desc,
        PangoVariant variant);
extern void pango_font_description_set_weight(PangoFontDescription
* desc,
        PangoWeight weight);
extern char *pango_font_description_to_filename(const
PangoFontDescription
        * desc);
extern char *pango_font_description_to_string(const
PangoFontDescription *
        desc);
extern void pango_font_description_unset_fields(PangoFontDescription *
        desc,
        PangoFontMask to_unset);
extern void pango_font_descriptions_free(PangoFontDescription *
*descs,
        int n_descs);
extern PangoFontDescription
*pango_font_face_describe(PangoFontFace *
        face);
extern const char *pango_font_face_get_face_name(PangoFontFace *
face);
extern GType pango_font_face_get_type(void);
extern gboolean pango_font_face_is_synthesized(PangoFontFace *
face);
extern void pango_font_face_list_sizes(PangoFontFace * face, int
**sizes,
        int *n_sizes);
extern const char *pango_font_family_get_name(PangoFontFamily *
family);
extern GType pango_font_family_get_type(void);
extern gboolean pango_font_family_is_monospace(PangoFontFamily *
family);
extern void pango_font_family_list_faces(PangoFontFamily * family,
PangoFontFace * **faces,
        int *n_faces);
extern PangoEngineShape *pango_font_find_shaper(PangoFont * font,
PangoLanguage * language,
        guint32 ch);
extern PangoCoverage *pango_font_get_coverage(PangoFont * font,
PangoLanguage * language);
extern PangoFontMap *pango_font_get_font_map(PangoFont * font);
extern void pango_font_get_glyph_extents(PangoFont * font,
PangoGlyph glyph,
        PangoRectangle * ink_rect,
        PangoRectangle * logical_rect);
extern PangoFontMetrics *pango_font_get_metrics(PangoFont * font,
PangoLanguage * language);
extern GType pango_font_get_type(void);
extern PangoContext *pango_font_map_create_context(PangoFontMap *
fontmap);
extern GType pango_font_map_get_type(void);
extern void pango_font_map_list_families(PangoFontMap * fontmap,
PangoFontFamily * **families,
        int *n_families);
extern PangoFont *pango_font_map_load_font(PangoFontMap * fontmap,
PangoContext * context,
        const PangoFontDescription *
desc);
extern PangoFontset *pango_font_map_load_fontset(PangoFontMap *
fontmap,
        PangoContext * context,

```

```

                                const
PangoFontDescription
                                * desc,
                                PangoLanguage * language);
extern GType pango_font_mask_get_type(void);
extern                                             int
pango_font_metrics_get_approximate_char_width(PangoFontMetrics *
                                                metrics);
extern                                             int
pango_font_metrics_get_approximate_digit_width(PangoFontMetrics
                                                * metrics);
extern int pango_font_metrics_get_ascent(PangoFontMetrics *
metrics);
extern int pango_font_metrics_get_descent(PangoFontMetrics *
metrics);
extern                                             int
pango_font_metrics_get_strikethrough_position(PangoFontMetrics *
                                                metrics);
extern                                             int
pango_font_metrics_get_strikethrough_thickness(PangoFontMetrics
                                                * metrics);
extern GType pango_font_metrics_get_type(void);
extern                                             int
pango_font_metrics_get_underline_position(PangoFontMetrics *
                                                metrics);
extern                                             int
pango_font_metrics_get_underline_thickness(PangoFontMetrics *
                                                metrics);
extern PangoFontMetrics *pango_font_metrics_ref(PangoFontMetrics *
metrics);
extern void pango_font_metrics_unref(PangoFontMetrics * metrics);
extern void pango_fontset_foreach(PangoFontset *,
PangoFontsetForeachFunc,
                                gpointer);
extern PangoFont *pango_fontset_get_font(PangoFontset *, guint);
extern PangoFontMetrics *pango_fontset_get_metrics(PangoFontset *);
extern GType pango_fontset_get_type(void);
extern void pango_get_log_attrs(const char *text, int length, int
level,
                                PangoLanguage * language,
                                PangoLogAttr * log_attrs, int attrs_len);
extern GSList *pango_glyph_item_apply_attrs(PangoGlyphItem *
glyph_item,
                                const char *text,
                                PangoAttrList * list);
extern PangoGlyphItem *pango_glyph_item_copy(PangoGlyphItem *
orig);
extern void pango_glyph_item_free(PangoGlyphItem * glyph_item);
extern void pango_glyph_item_get_logical_widths(PangoGlyphItem *
glyph_item,
                                                const char *text,
                                                int *logical_widths);
extern GType pango_glyph_item_get_type(void);
extern                                             PangoGlyphItemIter
*pango_glyph_item_iter_copy(PangoGlyphItemIter *
                                orig);
extern void pango_glyph_item_iter_free(PangoGlyphItemIter * iter);
extern GType pango_glyph_item_iter_get_type(void);
extern gboolean pango_glyph_item_iter_init_end(PangoGlyphItemIter
* iter,
                                                PangoGlyphItem * glyph_item,
                                                const char *text);
extern                                             gboolean
pango_glyph_item_iter_init_start(PangoGlyphItemIter * iter,
                                PangoGlyphItem *
                                glyph_item,

```

```

                                const char *text);
extern                                gboolean
pango_glyph_item_iter_next_cluster(PangoGlyphItemIter *
                                iter);
extern                                gboolean
pango_glyph_item_iter_prev_cluster(PangoGlyphItemIter *
                                iter);
extern void pango_glyph_item_letter_space(PangoGlyphItem *
glyph_item,
                                const char *text,
                                PangoLogAttr * log_attrs,
                                int letter_spacing);
extern PangoGlyphItem *pango_glyph_item_split(PangoGlyphItem *
orig,
                                const char *text,
                                int split_index);
extern PangoGlyphString *pango_glyph_string_copy(PangoGlyphString *
*
                                string);
extern void pango_glyph_string_extents(PangoGlyphString * glyphs,
                                PangoFont * font,
                                PangoRectangle * ink_rect,
                                PangoRectangle * logical_rect);
extern void pango_glyph_string_extents_range(PangoGlyphString *
glyphs,
                                int start, int end,
                                PangoFont * font,
                                PangoRectangle * ink_rect,
                                PangoRectangle *
                                logical_rect);
extern void pango_glyph_string_free(PangoGlyphString * string);
extern void pango_glyph_string_get_logical_widths(PangoGlyphString *
*
                                glyphs, const char *text,
                                int length,
                                int embedding_level,
                                int *logical_widths);
extern GType pango_glyph_string_get_type(void);
extern int pango_glyph_string_get_width(PangoGlyphString * glyphs);
extern void pango_glyph_string_index_to_x(PangoGlyphString *
glyphs,
                                char *text, int length,
                                PangoAnalysis * analysis,
                                int index_, gboolean trailing,
                                int *x_pos);
extern PangoGlyphString *pango_glyph_string_new(void);
extern void pango_glyph_string_set_size(PangoGlyphString * string,
gint new_len);
extern void pango_glyph_string_x_to_index(PangoGlyphString *
glyphs,
                                char *text, int length,
                                PangoAnalysis * analysis,
                                int x_pos, int *index_,
                                gboolean * trailing);
extern PangoGravity pango_gravity_get_for_matrix(const PangoMatrix *
*
                                matrix);
extern PangoGravity pango_gravity_get_for_script(PangoScript
script,
                                PangoGravity base_gravity,
                                PangoGravityHint hint);
extern PangoGravity
pango_gravity_get_for_script_and_width(PangoScript
                                script,
                                gboolean wide,
                                PangoGravity

```

```

base_gravity,
PangoGravityHint
hint);

extern GType pango_gravity_get_type(void);
extern GType pango_gravity_hint_get_type(void);
extern double pango_gravity_to_rotation(PangoGravity gravity);
extern PangoItem *pango_item_copy(PangoItem * item);
extern void pango_item_free(PangoItem * item);
extern GType pango_item_get_type(void);
extern PangoItem *pango_item_new(void);
extern PangoItem *pango_item_split(PangoItem * orig, int
split_index,
                                int split_offset);
extern GList *pango_itemize(PangoContext * context, const char
*text,
                                int start_index, int length,
                                PangoAttrList * attrs,
                                PangoAttrIterator * cached_iter);
extern GList *pango_itemize_with_base_dir(PangoContext * context,
PangoDirection base_dir,
const char *text,
int start_index, int length,
PangoAttrList * attrs,
PangoAttrIterator *
cached_iter);
extern PangoLanguage *pango_language_from_string(const char
*language);
extern PangoLanguage *pango_language_get_default(void);
extern const PangoScript *pango_language_get_scripts(PangoLanguage
*
                                language,
                                int *num_scripts);
extern GType pango_language_get_type(void);
extern gboolean pango_language_includes_script(PangoLanguage *
language,
                                PangoScript script);
extern gboolean pango_language_matches(PangoLanguage * language,
const char *range_list);
extern const char *pango_language_to_string(PangoLanguage *
language);
extern void pango_layout_context_changed(PangoLayout * layout);
extern PangoLayout *pango_layout_copy(PangoLayout * src);
extern PangoAlignment pango_layout_get_alignment(PangoLayout *
layout);
extern PangoAttrList *pango_layout_get_attributes(PangoLayout *
layout);
extern gboolean pango_layout_get_auto_dir(PangoLayout * layout);
extern int pango_layout_get_baseline(PangoLayout * layout);
extern gint pango_layout_get_character_count(PangoLayout * layout);
extern PangoContext *pango_layout_get_context(PangoLayout *
layout);
extern void pango_layout_get_cursor_pos(PangoLayout * layout, int
index_,
                                PangoRectangle * strong_pos,
                                PangoRectangle * weak_pos);
extern PangoEllipsizeMode pango_layout_get_ellipsize(PangoLayout *
layout);
extern void pango_layout_get_extents(PangoLayout * layout,
PangoRectangle * ink_rect,
PangoRectangle * logical_rect);
extern const PangoFontDescription
*pango_layout_get_font_description(PangoLayout * layout);
extern int pango_layout_get_height(PangoLayout * layout);
extern int pango_layout_get_indent(PangoLayout * layout);
extern PangoLayoutIter *pango_layout_get_iter(PangoLayout *
layout);

```

```

extern gboolean pango_layout_get_justify(PangoLayout * layout);
extern PangoLayoutLine *pango_layout_get_line(PangoLayout * layout,
                                              int line);
extern int pango_layout_get_line_count(PangoLayout * layout);
extern PangoLayoutLine *pango_layout_get_line_readonly(PangoLayout
*
                                              layout, int line);
extern GSList *pango_layout_get_lines(PangoLayout * layout);
extern GSList *pango_layout_get_lines_readonly(PangoLayout *
layout);
extern void pango_layout_get_log_attrs(PangoLayout * layout,
                                       PangoLogAttr * *attrs,
                                       gint * n_attrs);
extern const PangoLogAttr
*pango_layout_get_log_attrs_readonly(PangoLayout
* layout,
gint *
n_attrs);
extern void pango_layout_get_pixel_extents(PangoLayout * layout,
                                           PangoRectangle * ink_rect,
                                           PangoRectangle * logical_rect);
extern void pango_layout_get_pixel_size(PangoLayout * layout, int
*width,
int *height);
extern gboolean pango_layout_get_single_paragraph_mode(PangoLayout
*
layout);
extern void pango_layout_get_size(PangoLayout * layout, int *width,
int *height);
extern int pango_layout_get_spacing(PangoLayout * layout);
extern PangoTabArray *pango_layout_get_tabs(PangoLayout * layout);
extern const char *pango_layout_get_text(PangoLayout * layout);
extern GType pango_layout_get_type(void);
extern int pango_layout_get_unknown_glyphs_count(PangoLayout *
layout);
extern int pango_layout_get_width(PangoLayout * layout);
extern PangoWrapMode pango_layout_get_wrap(PangoLayout * layout);
extern void pango_layout_index_to_line_x(PangoLayout * layout, int
index_,
gboolean trailing, int *line,
int *x_pos);
extern void pango_layout_index_to_pos(PangoLayout * layout, int
index_,
PangoRectangle * pos);
extern gboolean pango_layout_is_ellipsized(PangoLayout * layout);
extern gboolean pango_layout_is_wrapped(PangoLayout * layout);
extern gboolean pango_layout_iter_at_last_line(PangoLayoutIter *
iter);
extern PangoLayoutIter *pango_layout_iter_copy(PangoLayoutIter *
iter);
extern void pango_layout_iter_free(PangoLayoutIter * iter);
extern int pango_layout_iter_get_baseline(PangoLayoutIter * iter);
extern void pango_layout_iter_get_char_extents(PangoLayoutIter *
iter,
PangoRectangle *
logical_rect);
extern void pango_layout_iter_get_cluster_extents(PangoLayoutIter
* iter,
PangoRectangle *
ink_rect,
PangoRectangle *
logical_rect);
extern int pango_layout_iter_get_index(PangoLayoutIter * iter);
extern PangoLayout *pango_layout_iter_get_layout(PangoLayoutIter *
iter);

```

```

extern void pango_layout_iter_get_layout_extents(PangoLayoutIter *
iter,
                                                PangoRectangle * ink_rect,
                                                PangoRectangle *
                                                logical_rect);
extern PangoLayoutLine *pango_layout_iter_get_line(PangoLayoutIter
* iter);
extern void pango_layout_iter_get_line_extents(PangoLayoutIter *
iter,
                                                PangoRectangle * ink_rect,
                                                PangoRectangle *
                                                logical_rect);
extern
                                                PangoLayoutLine
*pango_layout_iter_get_line_readonly(PangoLayoutIter
* iter);
extern void pango_layout_iter_get_line_yrange(PangoLayoutIter *
iter,
                                                int *y0_, int *y1_);
extern PangoLayoutRun *pango_layout_iter_get_run(PangoLayoutIter *
iter);
extern void pango_layout_iter_get_run_extents(PangoLayoutIter *
iter,
                                                PangoRectangle * ink_rect,
                                                PangoRectangle *
                                                logical_rect);
extern
                                                PangoLayoutRun
*pango_layout_iter_get_run_readonly(PangoLayoutIter *
iter);
extern GType pango_layout_iter_get_type(void);
extern gboolean pango_layout_iter_next_char(PangoLayoutIter *
iter);
extern gboolean pango_layout_iter_next_cluster(PangoLayoutIter *
iter);
extern gboolean pango_layout_iter_next_line(PangoLayoutIter *
iter);
extern gboolean pango_layout_iter_next_run(PangoLayoutIter * iter);
extern void pango_layout_line_get_extents(PangoLayoutLine * line,
                                                PangoRectangle * ink_rect,
                                                PangoRectangle * logical_rect);
extern void pango_layout_line_get_pixel_extents(PangoLayoutLine *
layout_line,
                                                PangoRectangle * ink_rect,
                                                PangoRectangle *
                                                logical_rect);
extern GType pango_layout_line_get_type(void);
extern void pango_layout_line_get_x_ranges(PangoLayoutLine * line,
                                                int start_index, int end_index,
                                                int **ranges, int *n_ranges);
extern void pango_layout_line_index_to_x(PangoLayoutLine * line,
                                                int index_, int trailing,
                                                int *x_pos);
extern PangoLayoutLine *pango_layout_line_ref(PangoLayoutLine *
line);
extern void pango_layout_line_unref(PangoLayoutLine * line);
extern gboolean pango_layout_line_x_to_index(PangoLayoutLine *
line,
                                                int x_pos, int *index_,
                                                int *trailing);
extern void pango_layout_move_cursor_visually(PangoLayout * layout,
                                                gboolean strong,
                                                int old_index,
                                                int old_trailing,
                                                int direction,
                                                int *new_index,
                                                int *new_trailing);
extern PangoLayout *pango_layout_new(PangoContext * context);

```



```

extern void pango_layout_set_alignment(PangoLayout * layout,
                                       PangoAlignment alignment);
extern void pango_layout_set_attributes(PangoLayout * layout,
                                       PangoAttrList * attrs);
extern void pango_layout_set_auto_dir(PangoLayout * layout,
                                       gboolean auto_dir);
extern void pango_layout_set_ellipsize(PangoLayout * layout,
                                       PangoEllipsizeMode ellipsize);
extern void pango_layout_set_font_description(PangoLayout * layout,
                                             const PangoFontDescription
*
                                             desc);
extern void pango_layout_set_height(PangoLayout * layout, int
height);
extern void pango_layout_set_indent(PangoLayout * layout, int
indent);
extern void pango_layout_set_justify(PangoLayout * layout,
                                     gboolean justify);
extern void pango_layout_set_markup(PangoLayout * layout,
                                    const char *markup, int length);
extern void pango_layout_set_markup_with_accel(PangoLayout *
layout,
                                             const char *markup,
                                             int length,
                                             gunichar accel_marker,
                                             gunichar * accel_char);
extern void pango_layout_set_single_paragraph_mode(PangoLayout *
layout,
                                                  gboolean setting);
extern void pango_layout_set_spacing(PangoLayout * layout, int
spacing);
extern void pango_layout_set_tabs(PangoLayout * layout,
                                  PangoTabArray * tabs);
extern void pango_layout_set_text(PangoLayout * layout, const char
*text,
                                  int length);
extern void pango_layout_set_width(PangoLayout * layout, int width);
extern void pango_layout_set_wrap(PangoLayout * layout,
                                  PangoWrapMode wrap);
extern gboolean pango_layout_xy_to_index(PangoLayout * layout, int
x,
                                       int y, int *index_,
                                       gint * trailing);
extern void pango_matrix_concat(PangoMatrix * matrix,
                               const PangoMatrix * new_matrix);
extern PangoMatrix *pango_matrix_copy(const PangoMatrix * matrix);
extern void pango_matrix_free(PangoMatrix * matrix);
extern double pango_matrix_get_font_scale_factor(const PangoMatrix
*
                                       matrix);
extern GType pango_matrix_get_type(void);
extern void pango_matrix_rotate(PangoMatrix * matrix, double
degrees);
extern void pango_matrix_scale(PangoMatrix * matrix, double scale_x,
                              double scale_y);
extern void pango_matrix_transform_distance(const PangoMatrix *
matrix,
                                       double *dx, double *dy);
extern void pango_matrix_transform_pixel_rectangle(const
PangoMatrix *
                                       matrix,
                                       PangoRectangle * rect);
extern void pango_matrix_transform_point(const PangoMatrix * matrix,
                                         double *x, double *y);
extern void pango_matrix_transform_rectangle(const PangoMatrix *
matrix,

```

```

        PangoRectangle * rect);
extern void pango_matrix_translate(PangoMatrix * matrix, double tx,
        double ty);
extern gboolean pango_parse_markup(const char *markup_text, int
length,
        gunichar accel_marker,
        PangoAttrList * attr_list, char
**text,
        gunichar * accel_char, GError *
*error);
extern GType pango_render_part_get_type(void);
extern void pango_renderer_activate(PangoRenderer * renderer);
extern void pango_renderer_deactivate(PangoRenderer * renderer);
extern void pango_renderer_draw_error_underline(PangoRenderer *
renderer,
        int x, int y, int width,
        int height);
extern void pango_renderer_draw_glyph(PangoRenderer * renderer,
        PangoFont * font, PangoGlyph glyph,
        double x, double y);
extern void pango_renderer_draw_glyph_item(PangoRenderer *
renderer,
        const char *text,
        PangoGlyphItem * glyph_item,
        int x, int y);
extern void pango_renderer_draw_glyphs(PangoRenderer * renderer,
        PangoFont * font,
        PangoGlyphString * glyphs, int x,
        int y);
extern void pango_renderer_draw_layout(PangoRenderer * renderer,
        PangoLayout * layout, int x, int
y);
extern void pango_renderer_draw_layout_line(PangoRenderer *
renderer,
        PangoLayoutLine * line, int x,
        int y);
extern void pango_renderer_draw_rectangle(PangoRenderer * renderer,
        PangoRenderPart part, int x,
        int y, int width, int height);
extern void pango_renderer_draw_trapezoid(PangoRenderer * renderer,
        PangoRenderPart part, double
y1_,
        double x11, double x21,
        double y2, double x12,
        double x22);
extern PangoColor *pango_renderer_get_color(PangoRenderer *
renderer,
        PangoRenderPart part);
extern PangoLayout *pango_renderer_get_layout(PangoRenderer *
renderer);
extern PangoLayoutLine
*pango_renderer_get_layout_line(PangoRenderer *
renderer);
extern const PangoMatrix *pango_renderer_get_matrix(PangoRenderer
*
        renderer);
extern GType pango_renderer_get_type(void);
extern void pango_renderer_part_changed(PangoRenderer * renderer,
        PangoRenderPart part);
extern void pango_renderer_set_color(PangoRenderer * renderer,
        PangoRenderPart part,
        const PangoColor * color);
extern void pango_renderer_set_matrix(PangoRenderer * renderer,
        const PangoMatrix * matrix);
extern GList *pango_reorder_items(GList * logical_items);
extern PangoScript pango_script_for_unichar(gunichar ch);

```

```

extern PangoLanguage *pango_script_get_sample_language(PangoScript
script);
extern GType pango_script_get_type(void);
extern void pango_script_iter_free(PangoScriptIter * iter);
extern void pango_script_iter_get_range(PangoScriptIter * iter,
const char **start,
const char **end,
PangoScript * script);
extern PangoScriptIter *pango_script_iter_new(const char *text,
int length);
extern gboolean pango_script_iter_next(PangoScriptIter * iter);
extern void pango_shape(const gchar * text, gint length,
const PangoAnalysis * analysis,
PangoGlyphString * glyphs);
extern GType pango_stretch_get_type(void);
extern GType pango_style_get_type(void);
extern GType pango_tab_align_get_type(void);
extern PangoTabArray *pango_tab_array_copy(PangoTabArray * src);
extern void pango_tab_array_free(PangoTabArray * tab_array);
extern
pango_tab_array_get_positions_in_pixels(PangoTabArray *
tab_array);
extern gint pango_tab_array_get_size(PangoTabArray * tab_array);
extern void pango_tab_array_get_tab(PangoTabArray * tab_array,
gint tab_index,
PangoTabAlign * alignment,
gint * location);
extern void pango_tab_array_get_tabs(PangoTabArray * tab_array,
PangoTabAlign * *alignments,
gint * *locations);
extern GType pango_tab_array_get_type(void);
extern PangoTabArray *pango_tab_array_new(gint initial_size,
gboolean positions_in_pixels);
extern PangoTabArray *pango_tab_array_new_with_positions(gint size,
gboolean
positions_in_pixels,
PangoTabAlign
first_alignment,
gint
first_position,
...);
extern void pango_tab_array_resize(PangoTabArray * tab_array,
gint new_size);
extern void pango_tab_array_set_tab(PangoTabArray * tab_array,
gint tab_index,
PangoTabAlign alignment,
gint location);
extern GType pango_underline_get_type(void);
extern PangoDirection pango_unichar_direction(gunichar ch);
extern int pango_units_from_double(double d);
extern double pango_units_to_double(int i);
extern GType pango_variant_get_type(void);
extern GType pango_weight_get_type(void);
extern GType pango_wrap_mode_get_type(void);

```

17.19 Interfaces for libpangocairo-1.0

Table 17-106 defines the library name and shared object name for the libpangocairo-1.0 library

Table 17-106 libpangocairo-1.0 Definition

Library:	libpangocairo-1.0

SONAME:	libpangocairo-1.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gobject 2.32] Gobject 2.32 Reference Manual

[Pango 1.30] Pango 1.30.1 Reference Manual

17.19.1 Pango Cairo Rendering

17.19.1.1 Interfaces for Pango Cairo Rendering

An LSB conforming implementation shall provide the generic functions for Pango Cairo Rendering specified in Table 17-107, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-107 libpangocairo-1.0 - Pango Cairo Rendering Function Interfaces

pango_cairo_context_get_font_options [Pango 1.30]	pango_cairo_context_get_resolution [Pango 1.30]
pango_cairo_context_get_shape_renderer [Pango 1.30]	pango_cairo_context_set_font_options [Pango 1.30]
pango_cairo_context_set_resolution [Pango 1.30]	pango_cairo_context_set_shape_renderer [Pango 1.30]
pango_cairo_create_context [Pango 1.30]	pango_cairo_create_layout [Pango 1.30]
pango_cairo_error_underline_path [Pango 1.30]	pango_cairo_font_get_scaled_font [Pango 1.30]
pango_cairo_font_get_type [Gobject 2.32]	pango_cairo_font_map_create_context [Pango 1.30]
pango_cairo_font_map_get_default [Pango 1.30]	pango_cairo_font_map_get_font_type [Pango 1.30]
pango_cairo_font_map_get_resolution [Pango 1.30]	pango_cairo_font_map_get_type [Gobject 2.32]
pango_cairo_font_map_new [Pango 1.30]	pango_cairo_font_map_new_for_font_type [Pango 1.30]
pango_cairo_font_map_set_default [Pango 1.30]	pango_cairo_font_map_set_resolution [Pango 1.30]
pango_cairo_glyph_string_path [Pango 1.30]	pango_cairo_layout_line_path [Pango 1.30]
pango_cairo_layout_path [Pango 1.30]	pango_cairo_show_error_underline [Pango 1.30]
pango_cairo_show_glyph_item [Pango 1.30]	pango_cairo_show_glyph_string [Pango 1.30]
pango_cairo_show_layout [Pango 1.30]	pango_cairo_show_layout_line [Pango 1.30]

pango_cairo_update_context [Pango 1.30]	pango_cairo_update_layout [Pango 1.30]

17.20 Data Definitions for libpangocairo-1.0

This section defines global identifiers and their values that are associated with interfaces contained in libpangocairo-1.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.20.1 pango-1.0/pango/pangocairo.h

```
#define PANGO_CAIRO_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_CAST \
    PANGO_TYPE_CAIRO_FONT_MAP, \
    PangoCairoFontMap)
#define PANGO_IS_CAIRO_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE \
    PANGO_TYPE_CAIRO_FONT_MAP)
#define PANGO_CAIRO_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST \
    ((object), PANGO_TYPE_CAIRO_FONT, PangoCairoFont))
#define PANGO_IS_CAIRO_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_CAIRO_FONT))
#define PANGO_TYPE_CAIRO_FONT (pango_cairo_font_get_type ())
#define PANGO_TYPE_CAIRO_FONT_MAP \
    (pango_cairo_font_map_get_type ())

typedef struct _PangoCairoFont PangoCairoFont;
typedef struct _PangoCairoFontMap PangoCairoFontMap;
typedef void (*PangoCairoShapeRendererFunc) (cairo_t * cr,
    PangoAttrShape * attr,
    gboolean do_path,
    gpointer data);

extern const cairo_font_options_t
    *pango_cairo_context_get_font_options(PangoContext * context);
extern double pango_cairo_context_get_resolution(PangoContext *
    context);
extern PangoCairoShapeRendererFunc
    pango_cairo_context_get_shape_renderer(PangoContext * context,
    gpointer * data);
extern void pango_cairo_context_set_font_options(PangoContext *
    context,
    const
    cairo_font_options_t
    * options);
extern void pango_cairo_context_set_resolution(PangoContext *
    context,
    double dpi);
```

```

extern void pango_cairo_context_set_shape_renderer(PangoContext *
context,

PangoCairoShapeRendererFunc
func, gpointer data,
GDestroyNotify dnotify);
extern PangoContext *pango_cairo_create_context(cairo_t * cr);
extern PangoLayout *pango_cairo_create_layout(cairo_t * cr);
extern void pango_cairo_error_underline_path(cairo_t * cr, double
x,
double y, double width,
double height);
extern
cairo_scaled_font_t
*pango_cairo_font_get_scaled_font(PangoCairoFont
* font);
extern GType pango_cairo_font_get_type(void);
extern
PangoContext
*pango_cairo_font_map_create_context(PangoCairoFontMap
* fontmap);
extern PangoFontMap *pango_cairo_font_map_get_default(void);
extern cairo_font_type_t
pango_cairo_font_map_get_font_type(PangoCairoFontMap * fontmap);
extern
double
pango_cairo_font_map_get_resolution(PangoCairoFontMap *
fontmap);
extern GType pango_cairo_font_map_get_type(void);
extern PangoFontMap *pango_cairo_font_map_new(void);
extern PangoFontMap
*pango_cairo_font_map_new_for_font_type(cairo_font_type_t
fonttype);
extern void pango_cairo_font_map_set_default(PangoCairoFontMap *
fontmap);
extern void pango_cairo_font_map_set_resolution(PangoCairoFontMap
*
fontmap, double dpi);
extern void pango_cairo_glyph_string_path(cairo_t * cr, PangoFont
* font,
PangoGlyphString * glyphs);
extern void pango_cairo_layout_line_path(cairo_t * cr,
PangoLayoutLine * line);
extern void pango_cairo_layout_path(cairo_t * cr, PangoLayout *
layout);
extern void pango_cairo_show_error_underline(cairo_t * cr, double
x,
double y, double width,
double height);
extern void pango_cairo_show_glyph_item(cairo_t * cr, const char
*text,
PangoGlyphItem * glyph_item);
extern void pango_cairo_show_glyph_string(cairo_t * cr, PangoFont
* font,
PangoGlyphString * glyphs);
extern void pango_cairo_show_layout(cairo_t * cr, PangoLayout *
layout);
extern void pango_cairo_show_layout_line(cairo_t * cr,
PangoLayoutLine * line);
extern void pango_cairo_update_context(cairo_t * cr,
PangoContext * context);
extern void pango_cairo_update_layout(cairo_t * cr, PangoLayout *
layout);

```

17.21 Interfaces for libpangoxft-1.0

Table 17-108 defines the library name and shared object name for the libpangoxft-1.0 library

Table 17-108 libpangoft-1.0 Definition

Library:	libpangoft-1.0
SONAME:	libpangoft-1.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gobject 2.32] Gobject 2.32 Reference Manual

[Pango 1.30] Pango 1.30.1 Reference Manual

17.21.1 Pango Xft Fonts and Rendering library

17.21.1.1 Interfaces for Pango Xft Fonts and Rendering library

An LSB conforming implementation shall provide the generic functions for Pango Xft Fonts and Rendering library specified in Table 17-109, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-109 libpangoft-1.0 - Pango Xft Fonts and Rendering library Function Interfaces

pango_xft_font_get_type [Gobject 2.32]	pango_xft_font_map_get_type [Gobject 2.32]
pango_xft_get_context [Pango 1.30]	pango_xft_get_font_map [Pango 1.30]
pango_xft_picture_render [Pango 1.30]	pango_xft_render [Pango 1.30]
pango_xft_render_layout [Pango 1.30]	pango_xft_render_layout_line [Pango 1.30]
pango_xft_render_transformed [Pango 1.30]	pango_xft_renderer_get_type [Gobject 2.32]
pango_xft_renderer_new [Pango 1.30]	pango_xft_renderer_set_default_color [Pango 1.30]
pango_xft_renderer_set_draw [Pango 1.30]	pango_xft_set_default_substitute [Pango 1.30]
pango_xft_shutdown_display [Pango 1.30]	pango_xft_substitute_changed [Pango 1.30]

17.22 Data Definitions for libpangoft-1.0

This section defines global identifiers and their values that are associated with interfaces contained in libpangoft-1.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and

application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.22.1 pango-1.0/pango/pangoxft.h

```
#define PANGO_XFT_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), PANGO_TYPE_XFT_RENDERER, \
    \
    PangoXftRendererClass))
#define PANGO_IS_XFT_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), PANGO_TYPE_XFT_RENDERER))
#define PANGO_XFT_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_XFT_FONT, \
    \
    PangoXftFont))
#define PANGO_XFT_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_XFT_FONT_MAP, \
    PangoXftFontMap))
#define PANGO_XFT_RENDERER(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_XFT_RENDERER, \
    PangoXftRenderer))
#define PANGO_XFT_IS_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_XFT_FONT))
#define PANGO_XFT_IS_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_XFT_FONT_MAP))
#define PANGO_IS_XFT_RENDERER(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_XFT_RENDERER))
#define PANGO_XFT_RENDERER_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), PANGO_TYPE_XFT_RENDERER, \
    \
    PangoXftRendererClass))
#define PANGO_TYPE_XFT_FONT (pango_xft_font_get_type ())
#define PANGO_TYPE_XFT_FONT_MAP (pango_xft_font_map_get_type ())
#define PANGO_TYPE_XFT_RENDERER (pango_xft_renderer_get_type ())
#define PANGO_RENDERER_TYPE_XFT "PangoRenderXft"

typedef struct _PangoXftRenderer {
    PangoRenderer parent_instance;
    Display *display;
    int screen;
    XftDraw *draw;
    PangoXftRendererPrivate *priv;
} PangoXftRenderer;
typedef void (*PangoXftSubstituteFunc) (FcPattern * pattern,
    gpointer data);
typedef struct _PangoXftFontMap PangoXftFontMap;
typedef struct _PangoXftRendererClass {
    PangoRendererClass parent_class;
    void (*composite_trapezoids) (PangoXftRenderer * xftrenderer,
        PangoRenderPart part,
        XTrapezoid * trapezoids,
        int n_trapezoids);
    void (*composite_glyphs) (PangoXftRenderer * xftrenderer,
        XftFont * xft_font, XftGlyphSpec * glyphs,
        int n_glyphs);
```



```

} PangoXftRendererClass;
typedef struct _PangoFcFontClass PangoFcFontClass;
typedef struct _PangoFcFont PangoFcFont;
typedef struct _PangoXftFont PangoXftFont;
typedef struct _PangoXftRendererPrivate PangoXftRendererPrivate;
extern GType pango_xft_font_get_type(void);
extern GType pango_xft_font_map_get_type(void);
extern PangoContext *pango_xft_get_context(Display * display, int
screen);
extern PangoFontMap *pango_xft_get_font_map(Display * display, int
screen);
extern void pango_xft_picture_render(Display * display,
Picture src_picture,
Picture dest_picture,
PangoFont * font,
PangoGlyphString * glyphs, gint x,
gint y);
extern void pango_xft_render(XftDraw * draw, XftColor * color,
PangoFont * font, PangoGlyphString *
glyphs,
gint x, gint y);
extern void pango_xft_render_layout(XftDraw * draw, XftColor *
color,
PangoLayout * layout, int x, int y);
extern void pango_xft_render_layout_line(XftDraw * draw, XftColor
* color,
PangoLayoutLine * line, int x,
int y);
extern void pango_xft_render_transformed(XftDraw * draw, XftColor
* color,
PangoMatrix * matrix,
PangoFont * font,
PangoGlyphString * glyphs, int x,
int y);
extern GType pango_xft_renderer_get_type(void);
extern PangoRenderer *pango_xft_renderer_new(Display * display,
int screen);
extern void pango_xft_renderer_set_default_color(PangoXftRenderer
*
xftrenderer,
PangoColor *
default_color);
extern void pango_xft_renderer_set_draw(PangoXftRenderer *
xftrenderer,
XftDraw * draw);
extern void pango_xft_set_default_substitute(Display * display, int
screen,
PangoXftSubstituteFunc func,
gpointer data,
GDestroyNotify notify);
extern void pango_xft_shutdown_display(Display * display, int
screen);
extern void pango_xft_substitute_changed(Display * display, int
screen);

```

17.23 Interfaces for libpangoft2-1.0

Table 17-110 defines the library name and shared object name for the libpangoft2-1.0 library

Table 17-110 libpangoft2-1.0 Definition

Library:	libpangoft2-1.0
SONAME:	libpangoft2-1.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gobject 2.32] Gobject 2.32 Reference Manual

[Pango 1.30] Pango 1.30.1 Reference Manual

17.23.1 Pango Freetype Fonts and Rendering library

17.23.1.1 Interfaces for Pango Freetype Fonts and Rendering library

An LSB conforming implementation shall provide the generic functions for Pango Freetype Fonts and Rendering library specified in Table 17-111, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-111 libpangoft2-1.0 - Pango Freetype Fonts and Rendering library Function Interfaces

pango_fc_decoder_get_charset [Pango 1.30]	pango_fc_decoder_get_glyph [Pango 1.30]
pango_fc_decoder_get_type [Gobject 2.32]	pango_fc_font_description_from_pattern [Pango 1.30]
pango_fc_font_get_type [Gobject 2.32]	pango_fc_font_lock_face [Pango 1.30]
pango_fc_font_map_add_decoder_find_func [Pango 1.30]	pango_fc_font_map_cache_clear [Pango 1.30]
pango_fc_font_map_find_decoder [Pango 1.30]	pango_fc_font_map_get_type [Gobject 2.32]
pango_fc_font_unlock_face [Pango 1.30]	pango_ft2_font_map_create_context [Pango 1.30]
pango_ft2_font_map_get_type [Gobject 2.32]	pango_ft2_font_map_new [Pango 1.30]
pango_ft2_font_map_set_default_substitute [Pango 1.30]	pango_ft2_font_map_set_resolution [Pango 1.30]
pango_ft2_font_map_substitute_changed [Pango 1.30]	pango_ft2_render [Pango 1.30]
pango_ft2_render_layout [Pango 1.30]	pango_ft2_render_layout_line [Pango 1.30]
pango_ft2_render_layout_line_subpixel [Pango 1.30]	pango_ft2_render_layout_subpixel [Pango 1.30]
pango_ft2_render_transformed [Pango 1.30]	

An LSB conforming implementation shall provide the generic deprecated functions for Pango Freetype Fonts and Rendering library specified in Table 17-112, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

**Table 17-112 libpangoft2-1.0 - Pango FreeType Fonts and Rendering library
Deprecated Function Interfaces**

pango_ft2_font_map_create_context [Pango 1.30]	

17.24 Data Definitions for libpangoft2-1.0

This section defines global identifiers and their values that are associated with interfaces contained in libpangoft2-1.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.24.1 pango-1.0/pango/pangofc-decoder.h

```
#define PANGO_FC_DECODER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), PANGO_TYPE_FC_DECODER, \
    PangoFcDecoderClass))
#define PANGO_IS_FC_DECODER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), PANGO_TYPE_FC_DECODER))
#define PANGO_FC_DECODER(object) \
    (G_TYPE_CHECK_INSTANCE_CAST \
    PANGO_TYPE_FC_DECODER, \
    PangoFcDecoder)
#define PANGO_IS_FC_DECODER(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE \
    PANGO_TYPE_FC_DECODER)
#define PANGO_FC_DECODER_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), PANGO_TYPE_FC_DECODER, \
    PangoFcDecoderClass))
#define PANGO_TYPE_FC_DECODER (pango_fc_decoder_get_type())

typedef struct _PangoFcDecoder {
    GObject parent_instance;
} PangoFcDecoder;
typedef struct _PangoFcDecoderClass {
    GObjectClass parent_class;
    FcCharSet *(*get_charset) (PangoFcDecoder * decoder,
    PangoFcFont * fcfont);
    PangoGlyph(*get_glyph) (PangoFcDecoder * decoder,
    PangoFcFont * fcfont, guint32 wc);
    void (*_pango_reserved1) (void);
    void (*_pango_reserved2) (void);
    void (*_pango_reserved3) (void);
    void (*_pango_reserved4) (void);
```

```

} PangoFcDecoderClass;
extern FcCharSet *pango_fc_decoder_get_charset(PangoFcDecoder *
decoder,
                                           PangoFcFont * fcfont);
extern PangoGlyph pango_fc_decoder_get_glyph(PangoFcDecoder *
decoder,
                                           PangoFcFont * fcfont,
                                           guint32 wc);
extern GType pango_fc_decoder_get_type(void);

```

17.24.2 pango-1.0/pango/pangofc-fontmap.h

```

#define PANGO_FC_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), \
    PANGO_TYPE_FC_FONT_MAP, \
    PangoFcFontMap))
#define PANGO_IS_FC_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), \
    PANGO_TYPE_FC_FONT_MAP))
#define PANGO_TYPE_FC_FONT_MAP (pango_fc_font_map_get_type ())
#define PANGO_FC_GRAVITY "pangogravity"
#define PANGO_FC_PRGNAME "pangoprgrname"
#define PANGO_FC_VERSION "pangoversion"

typedef struct _PangoFcFontMap PangoFcFontMap;
typedef PangoFcDecoder *(*PangoFcDecoderFindFunc) (FcPattern *
pattern,
                                                    gpointer user_data);
typedef struct _PangoFcFontMapClass PangoFcFontMapClass;
extern PangoFontDescription
    *pango_fc_font_description_from_pattern(FcPattern * pattern,
                                           gboolean include_size);
extern void pango_fc_font_map_add_decoder_find_func(PangoFcFontMap
*
                                                    fcfontmap,
                                                    PangoFcDecoderFindFunc
findfunc,
                                                    gpointer user_data,
                                                    GDestroyNotify
dnotify);
extern void pango_fc_font_map_cache_clear(PangoFcFontMap *
fcfontmap);
extern PangoFcDecoder
    *pango_fc_font_map_find_decoder(PangoFcFontMap *
fcfontmap,
FcPattern * pattern);
extern GType pango_fc_font_map_get_type(void);

```

17.24.3 pango-1.0/pango/pangoft2.h

```

#define PANGO_FT2_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), PANGO_TYPE_FT2_FONT, \
    PangoFcFont))
#define PANGO_FT2_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), \
    PANGO_TYPE_FT2_FONT_MAP, \
    PangoFT2FontMap))
#define PANGO_IS_FT2_FONT(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), PANGO_TYPE_FT2_FONT))
#define PANGO_FT2_IS_FONT_MAP(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), \
    PANGO_TYPE_FT2_FONT_MAP))

```

```

#define PANGO_TYPE_FC_FONT      (pango_fc_font_get_type ())
#define PANGO_TYPE_FT2_FONT_MAP (pango_ft2_font_map_get_type ())
#define PANGO_RENDER_TYPE_FT2   "PangoRenderFT2"

typedef struct _PangoFT2FontMap PangoFT2FontMap;
typedef void (*PangoFT2SubstituteFunc) (FcPattern * pattern,
                                         gpointer data);

extern GType pango_fc_font_get_type(void);
extern FT_Face pango_fc_font_lock_face(PangoFcFont * font);
extern void pango_fc_font_unlock_face(PangoFcFont * font);
extern PangoContext
*pango_ft2_font_map_create_context(PangoFT2FontMap *
                                   fontmap);

extern GType pango_ft2_font_map_get_type(void);
extern PangoFontMap *pango_ft2_font_map_new(void);
extern void
pango_ft2_font_map_set_default_substitute(PangoFT2FontMap *
                                           fontmap,
                                           PangoFT2SubstituteFunc
                                           func, gpointer data,
                                           GDestroyNotify
                                           notify);

extern void pango_ft2_font_map_set_resolution(PangoFT2FontMap *
                                              fontmap,
                                              double dpi_x, double dpi_y);
extern void pango_ft2_font_map_substitute_changed(PangoFT2FontMap *
                                                  *
                                                  fontmap);

extern void pango_ft2_render(FT_Bitmap * bitmap, PangoFont * font,
                             PangoGlyphString * glyphs, gint x, gint y);
extern void pango_ft2_render_layout(FT_Bitmap * bitmap,
                                     PangoLayout * layout, int x, int y);
extern void pango_ft2_render_layout_line(FT_Bitmap * bitmap,
                                          PangoLayoutLine * line, int x,
                                          int y);
extern void pango_ft2_render_layout_line_subpixel(FT_Bitmap *
                                                  bitmap,
                                                  PangoLayoutLine * line,
                                                  int x, int y);
extern void pango_ft2_render_layout_subpixel(FT_Bitmap * bitmap,
                                              PangoLayout * layout, int x,
                                              int y);
extern void pango_ft2_render_transformed(FT_Bitmap * bitmap,
                                          const PangoMatrix * matrix,
                                          PangoFont * font,
                                          PangoGlyphString * glyphs, int x,
                                          int y);

```

17.25 Interfaces for libgdk_pixbuf-2.0

Table 17-113 defines the library name and shared object name for the libgdk_pixbuf-2.0 library

Table 17-113 libgdk_pixbuf-2.0 Definition

Library:	libgdk_pixbuf-2.0
SONAME:	libgdk_pixbuf-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gdk-pixbuf 2.26] Gdk-pixbuf 2.26.0 Reference Manual

[Gobject 2.32] Gobject 2.32 Reference Manual

17.25.1 GDK pixbuf rendering on drawables library

17.25.1.1 Interfaces for GDK pixbuf rendering on drawables library

An LSB conforming implementation shall provide the generic functions for GDK pixbuf rendering on drawables library specified in Table 17-114, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-114 libgdk_pixbuf-2.0 - GDK pixbuf rendering on drawables library Function Interfaces

gdk_colorspace_get_type [Gobject 2.32]	gdk_interp_type_get_type [Gobject 2.32]
gdk_pixbuf_add_alpha [Gdk-pixbuf 2.26]	gdk_pixbuf_alpha_mode_get_type [Gobject 2.32]
gdk_pixbuf_animation_get_height [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_get_iter [Gdk-pixbuf 2.26]
gdk_pixbuf_animation_get_static_image [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_get_type [Gobject 2.32]
gdk_pixbuf_animation_get_width [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_is_static_image [Gdk-pixbuf 2.26]
gdk_pixbuf_animation_iter_advance [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_iter_get_delay_time [Gdk-pixbuf 2.26]
gdk_pixbuf_animation_iter_get_pixbuf [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_iter_get_type [Gobject 2.32]
gdk_pixbuf_animation_iter_on_currently_loading_frame [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_new_from_file [Gdk-pixbuf 2.26]
gdk_pixbuf_animation_ref [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_unref [Gdk-pixbuf 2.26]
gdk_pixbuf_apply_embedded_orientation [Gdk-pixbuf 2.26]	gdk_pixbuf_composite [Gdk-pixbuf 2.26]
gdk_pixbuf_composite_color [Gdk-pixbuf 2.26]	gdk_pixbuf_composite_color_simple [Gdk-pixbuf 2.26]
gdk_pixbuf_copy [Gdk-pixbuf 2.26]	gdk_pixbuf_copy_area [Gdk-pixbuf 2.26]
gdk_pixbuf_error_get_type [Gobject 2.32]	gdk_pixbuf_error_quark [Gdk-pixbuf 2.26]
gdk_pixbuf_fill [Gdk-pixbuf 2.26]	gdk_pixbuf_flip [Gdk-pixbuf 2.26]
gdk_pixbuf_format_copy [Gdk-pixbuf 2.26]	gdk_pixbuf_format_free [Gdk-pixbuf 2.26]
gdk_pixbuf_format_get_description [Gdk-pixbuf 2.26]	gdk_pixbuf_format_get_extensions [Gdk-pixbuf 2.26]
gdk_pixbuf_format_get_license [Gdk-pixbuf 2.26]	gdk_pixbuf_format_get_mime_types [Gdk-pixbuf 2.26]

gdk_pixbuf_format_get_name [Gdk-pixbuf 2.26]	gdk_pixbuf_format_get_type [Gobject 2.32]
gdk_pixbuf_format_is_disabled [Gdk-pixbuf 2.26]	gdk_pixbuf_format_is_scalable [Gdk-pixbuf 2.26]
gdk_pixbuf_format_is_writable [Gdk-pixbuf 2.26]	gdk_pixbuf_format_set_disabled [Gdk-pixbuf 2.26]
gdk_pixbuf_from_pixdata [Gdk-pixbuf 2.26]	gdk_pixbuf_get_bits_per_sample [Gdk-pixbuf 2.26]
gdk_pixbuf_get_byte_length [Gdk-pixbuf 2.26]	gdk_pixbuf_get_colorspace [Gdk-pixbuf 2.26]
gdk_pixbuf_get_file_info [Gdk-pixbuf 2.26]	gdk_pixbuf_get_formats [Gdk-pixbuf 2.26]
gdk_pixbuf_get_has_alpha [Gdk-pixbuf 2.26]	gdk_pixbuf_get_height [Gdk-pixbuf 2.26]
gdk_pixbuf_get_n_channels [Gdk-pixbuf 2.26]	gdk_pixbuf_get_option [Gdk-pixbuf 2.26]
gdk_pixbuf_get_pixels [Gdk-pixbuf 2.26]	gdk_pixbuf_get_pixels_with_length [Gdk-pixbuf 2.26]
gdk_pixbuf_get_rowstride [Gdk-pixbuf 2.26]	gdk_pixbuf_get_type [Gobject 2.32]
gdk_pixbuf_get_width [Gdk-pixbuf 2.26]	gdk_pixbuf_loader_close [Gdk-pixbuf 2.26]
gdk_pixbuf_loader_get_animation [Gdk-pixbuf 2.26]	gdk_pixbuf_loader_get_format [Gdk-pixbuf 2.26]
gdk_pixbuf_loader_get_pixbuf [Gdk-pixbuf 2.26]	gdk_pixbuf_loader_get_type [Gobject 2.32]
gdk_pixbuf_loader_new [Gdk-pixbuf 2.26]	gdk_pixbuf_loader_new_with_mime_type [Gdk-pixbuf 2.26]
gdk_pixbuf_loader_new_with_type [Gdk-pixbuf 2.26]	gdk_pixbuf_loader_set_size [Gdk-pixbuf 2.26]
gdk_pixbuf_loader_write [Gdk-pixbuf 2.26]	gdk_pixbuf_new [Gdk-pixbuf 2.26]
gdk_pixbuf_new_from_data [Gdk-pixbuf 2.26]	gdk_pixbuf_new_from_file [Gdk-pixbuf 2.26]
gdk_pixbuf_new_from_file_at_scale [Gdk-pixbuf 2.26]	gdk_pixbuf_new_from_file_at_size [Gdk-pixbuf 2.26]
gdk_pixbuf_new_from_inline [Gdk-pixbuf 2.26]	gdk_pixbuf_new_from_resource [Gdk-pixbuf 2.26]
gdk_pixbuf_new_from_resource_at_scale [Gdk-pixbuf 2.26]	gdk_pixbuf_new_from_stream [Gdk-pixbuf 2.26]
gdk_pixbuf_new_from_stream_async [Gdk-pixbuf 2.26]	gdk_pixbuf_new_from_stream_at_scale [Gdk-pixbuf 2.26]

gdk_pixbuf_new_from_stream_at_scale_async [Gdk-pixbuf 2.26]	gdk_pixbuf_new_from_stream_finish [Gdk-pixbuf 2.26]
gdk_pixbuf_new_from_xpm_data [Gdk-pixbuf 2.26]	gdk_pixbuf_new_subpixbuf [Gdk-pixbuf 2.26]
gdk_pixbuf_ref [Gdk-pixbuf 2.26]	gdk_pixbuf_rotate_simple [Gdk-pixbuf 2.26]
gdk_pixbuf_rotation_get_type [GObject 2.32]	gdk_pixbuf_saturate_and_pixelate [Gdk-pixbuf 2.26]
gdk_pixbuf_save [Gdk-pixbuf 2.26]	gdk_pixbuf_save_to_buffer [Gdk-pixbuf 2.26]
gdk_pixbuf_save_to_bufferv [Gdk-pixbuf 2.26]	gdk_pixbuf_save_to_callback [Gdk-pixbuf 2.26]
gdk_pixbuf_save_to_callbackv [Gdk-pixbuf 2.26]	gdk_pixbuf_save_to_stream [Gdk-pixbuf 2.26]
gdk_pixbuf_save_to_stream_async [Gdk-pixbuf 2.26]	gdk_pixbuf_save_to_stream_finish [Gdk-pixbuf 2.26]
gdk_pixbuf_savev [Gdk-pixbuf 2.26]	gdk_pixbuf_scale [Gdk-pixbuf 2.26]
gdk_pixbuf_scale_simple [Gdk-pixbuf 2.26]	gdk_pixbuf_simple_anim_add_frame [Gdk-pixbuf 2.26]
gdk_pixbuf_simple_anim_get_loop [Gdk-pixbuf 2.26]	gdk_pixbuf_simple_anim_get_type [GObject 2.32]
gdk_pixbuf_simple_anim_iter_get_type [GObject 2.32]	gdk_pixbuf_simple_anim_new [Gdk-pixbuf 2.26]
gdk_pixbuf_simple_anim_set_loop [Gdk-pixbuf 2.26]	gdk_pixbuf_unref [Gdk-pixbuf 2.26]
gdk_pixdata_deserialize [Gdk-pixbuf 2.26]	gdk_pixdata_from_pixbuf [Gdk-pixbuf 2.26]
gdk_pixdata_serialize [Gdk-pixbuf 2.26]	gdk_pixdata_to_csource [Gdk-pixbuf 2.26]

An LSB conforming implementation shall provide the generic deprecated functions for GDK pixbuf rendering on drawables library specified in Table 17-115, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

**Table 17-115 libgdk_pixbuf-2.0 - GDK pixbuf rendering on drawables library
Deprecated Function Interfaces**

gdk_pixbuf_animation_ref [Gdk-pixbuf 2.26]	gdk_pixbuf_animation_unref [Gdk-pixbuf 2.26]
gdk_pixbuf_ref [Gdk-pixbuf 2.26]	gdk_pixbuf_unref [Gdk-pixbuf 2.26]

An LSB conforming implementation shall provide the generic data interfaces for GDK pixbuf rendering on drawables library specified in Table 17-116, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-116 libgdk_pixbuf-2.0 - GDK pixbuf rendering on drawables library Data Interfaces

gdk_pixbuf_major_version [Gdk-pixbuf 2.26]	gdk_pixbuf_micro_version [Gdk-pixbuf 2.26]
gdk_pixbuf_minor_version [Gdk-pixbuf 2.26]	gdk_pixbuf_version [Gdk-pixbuf 2.26]

17.26 Data Definitions for libgdk_pixbuf-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libgdk_pixbuf-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.26.1 gdk-pixbuf-2.0/gdk-pixbuf/gdk-pixbuf.h

```
#define GDK_TYPE_PIXBUF_ANIMATION_ITER \
    (gdk_pixbuf_animation_iter_get_type ())
#define GDK_PIXBUF_LOADER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_PIXBUF_LOADER, \
    \
    GdkPixbufLoaderClass))
#define GDK_IS_PIXBUF_LOADER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_PIXBUF_LOADER))
#define GDK_PIXBUF_LOADER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GDK_TYPE_PIXBUF_LOADER, \
    \
    GdkPixbufLoader))
#define GDK_PIXBUF(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_PIXBUF, \
    GdkPixbuf))
#define GDK_PIXBUF_ANIMATION(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), \
    GDK_TYPE_PIXBUF_ANIMATION, \
    GdkPixbufAnimation))
#define GDK_PIXBUF_ANIMATION_ITER(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), \
    GDK_TYPE_PIXBUF_ANIMATION_ITER, GdkPixbufAnimationIter))
#define GDK_IS_PIXBUF_LOADER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GDK_TYPE_PIXBUF_LOADER))
```

```

#define GDK_IS_PIXBUF(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_PIXBUF))
#define GDK_IS_PIXBUF_ANIMATION(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), \
    GDK_TYPE_PIXBUF_ANIMATION))
#define GDK_IS_PIXBUF_ANIMATION_ITER(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), \
    GDK_TYPE_PIXBUF_ANIMATION_ITER))
#define GDK_PIXBUF_LOADER_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_PIXBUF_LOADER, \
    GdkPixbufLoaderClass))
#define GDK_PIXBUF_VERSION "2.26.0"
#define GDK_PIXBUF_MICRO (0)
#define GDK_PIXBUF_MAJOR (2)
#define GDK_PIXBUF_MINOR (26)
#define GDK_TYPE_COLORSPACE (gdk_colorspace_get_type())
#define GDK_TYPE_INTERP_TYPE (gdk_interp_type_get_type())
#define GDK_TYPE_PIXBUF_ALPHA_MODE (gdk_pixbuf_alpha_mode_get_type())
#define GDK_TYPE_PIXBUF_ANIMATION (gdk_pixbuf_animation_get_type())
#define GDK_TYPE_PIXBUF_ERROR (gdk_pixbuf_error_get_type())
#define GDK_TYPE_PIXBUF (gdk_pixbuf_get_type())
#define GDK_TYPE_PIXBUF_LOADER (gdk_pixbuf_loader_get_type())
#define GDK_TYPE_PIXBUF_ROTATION (gdk_pixbuf_rotation_get_type())
#define GDK_TYPE_PIXBUF_SIMPLE_ANIM (gdk_pixbuf_simple_anim_get_type())
#define GDK_PIXBUF_SIMPLE_ANIM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_PIXBUF_SIMPLE_ANIM, \
    GdkPixbufSimpleAnimClass))
#define GDK_IS_PIXBUF_SIMPLE_ANIM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_PIXBUF_SIMPLE_ANIM))
#define GDK_PIXBUF_SIMPLE_ANIM(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_PIXBUF_SIMPLE_ANIM, \
    GdkPixbufSimpleAnim))
#define GDK_IS_PIXBUF_SIMPLE_ANIM(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((object), \
    GDK_TYPE_PIXBUF_SIMPLE_ANIM))
#define GDK_PIXBUF_SIMPLE_ANIM_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_PIXBUF_SIMPLE_ANIM, \
    GdkPixbufSimpleAnimClass))
#define GDK_PIXBUF_VAR extern
#define GDK_PIXBUF_ERROR gdk_pixbuf_error_quark ()

typedef struct _GdkPixbuf GdkPixbuf;
typedef gboolean(*GdkPixbufSaveFunc) (const gchar *, gsize, GError
**);
typedef struct _GdkPixbufLoader {
    GObject parent_instance;
    gpointer priv;
} GdkPixbufLoader;
typedef struct _GdkPixbufFormat GdkPixbufFormat;
typedef enum {
    GDK_INTERP_NEAREST = 0,
    GDK_INTERP_TILES = 1,
    GDK_INTERP_BILINEAR = 2,
    GDK_INTERP_HYPER = 3
} GdkInterpType;
typedef enum {
    GDK_COLORSPACE_RGB = 0
} GdkColorspace;
typedef void (*GdkPixbufDestroyNotify) (guchar *, gpointer);

```

```

typedef struct _GdkPixbufAnimationIter GdkPixbufAnimationIter;
typedef struct _GdkPixbufAnimation GdkPixbufAnimation;
typedef enum {
    GDK_PIXBUF_ROTATE_NONE = 0,
    GDK_PIXBUF_ROTATE_COUNTERCLOCKWISE = 90,
    GDK_PIXBUF_ROTATE_UPSIDEDOWN = 180,
    GDK_PIXBUF_ROTATE_CLOCKWISE = 270
} GdkPixbufRotation;
typedef struct _GdkPixbufLoaderClass {
    GObjectClass parent_class;
    void (*size_prepared) (GdkPixbufLoader *, int, int);
    void (*area_prepared) (GdkPixbufLoader *);
    void (*area_updated) (GdkPixbufLoader *, int, int, int, int);
    void (*closed) (GdkPixbufLoader *);
} GdkPixbufLoaderClass;
typedef enum {
    GDK_PIXBUF_ERROR_CORRUPT_IMAGE,
    GDK_PIXBUF_ERROR_INSUFFICIENT_MEMORY,
    GDK_PIXBUF_ERROR_BAD_OPTION,
    GDK_PIXBUF_ERROR_UNKNOWN_TYPE,
    GDK_PIXBUF_ERROR_UNSUPPORTED_OPERATION,
    GDK_PIXBUF_ERROR_FAILED
} GdkPixbufError;
typedef enum {
    GDK_PIXBUF_ALPHA_BILEVEL,
    GDK_PIXBUF_ALPHA_FULL
} GdkPixbufAlphaMode;
typedef struct _GdkPixbufSimpleAnim GdkPixbufSimpleAnim;
typedef struct _GdkPixbufSimpleAnimClass GdkPixbufSimpleAnimClass;
extern GType gdk_colorspace_get_type(void);
extern GType gdk_interp_type_get_type(void);
extern GdkPixbuf *gdk_pixbuf_add_alpha(const GdkPixbuf * pixbuf,
    gboolean substitute_color, guchar
    r,
    guchar g, guchar b);
extern GType gdk_pixbuf_alpha_mode_get_type(void);
extern int gdk_pixbuf_animation_get_height(GdkPixbufAnimation *
    animation);
extern GdkPixbufAnimationIter
    *gdk_pixbuf_animation_get_iter(GdkPixbufAnimation * animation,
    const GTimeVal * start_time);
extern
    GdkPixbuf
    *gdk_pixbuf_animation_get_static_image(GdkPixbufAnimation
    * animation);
extern GType gdk_pixbuf_animation_get_type(void);
extern int gdk_pixbuf_animation_get_width(GdkPixbufAnimation *
    animation);
extern
    gboolean
    gdk_pixbuf_animation_is_static_image(GdkPixbufAnimation *
    animation);
extern
    gboolean
    gdk_pixbuf_animation_iter_advance(GdkPixbufAnimationIter *
    iter,
    const GTimeVal *
    current_time);
extern
    int
    gdk_pixbuf_animation_iter_get_delay_time(GdkPixbufAnimationIter
    * iter);
extern GdkPixbuf
    *gdk_pixbuf_animation_iter_get_pixbuf(GdkPixbufAnimationIter *
    iter);
extern GType gdk_pixbuf_animation_iter_get_type(void);
extern gboolean
    gdk_pixbuf_animation_iter_on_currently_loading_frame(GdkPixbufAni
    mationIter
    * iter);

```

```

extern                                     GdkPixbufAnimation
*gdk_pixbuf_animation_new_from_file(const char
                                     *filename,
                                     GError *
                                     *error);
extern                                     GdkPixbufAnimation
*gdk_pixbuf_animation_ref(GdkPixbufAnimation *
                                     animation);
extern void gdk_pixbuf_animation_unref(GdkPixbufAnimation *
animation);
extern GdkPixbuf *gdk_pixbuf_apply_embedded_orientation(GdkPixbuf
* src);
extern void gdk_pixbuf_composite(const GdkPixbuf * src, GdkPixbuf
* dest,
                                     int dest_x, int dest_y, int dest_width,
                                     int dest_height, double offset_x,
                                     double offset_y, double scale_x,
                                     double scale_y, GdkInterpType
interp_type,
                                     int overall_alpha);
extern void gdk_pixbuf_composite_color(const GdkPixbuf *src,
                                     GdkPixbuf * dest, int dest_x,
                                     int dest_y, int dest_width,
                                     int dest_height, double offset_x,
                                     double offset_y, double scale_x,
                                     double scale_y,
                                     GdkInterpType interp_type,
                                     int overall_alpha, int check_x,
                                     int check_y, int check_size,
                                     guint32 color1, guint32 color2);
extern      GdkPixbuf      *gdk_pixbuf_composite_color_simple(const
GdkPixbuf * src,
                                     int dest_width,
                                     int dest_height,
                                     GdkInterpType
interp_type,
                                     int overall_alpha,
                                     int check_size,
                                     guint32 color1,
                                     guint32 color2);
extern GdkPixbuf *gdk_pixbuf_copy(const GdkPixbuf * pixbuf);
extern void gdk_pixbuf_copy_area(const GdkPixbuf * src_pixbuf, int
src_x,
                                     int src_y, int width, int height,
                                     GdkPixbuf * dest_pixbuf, int dest_x,
                                     int dest_y);
extern GType gdk_pixbuf_error_get_type(void);
extern GQuark gdk_pixbuf_error_quark(void);
extern void gdk_pixbuf_fill(GdkPixbuf * pixbuf, guint32 pixel);
extern GdkPixbuf *gdk_pixbuf_flip(const GdkPixbuf * src,
gboolean horizontal);
extern      GdkPixbufFormat      *gdk_pixbuf_format_copy(const
GdkPixbufFormat *
                                     format);
extern void gdk_pixbuf_format_free(GdkPixbufFormat * format);
extern gchar *gdk_pixbuf_format_get_description(GdkPixbufFormat *
format);
extern gchar **gdk_pixbuf_format_get_extensions(GdkPixbufFormat *
format);
extern gchar *gdk_pixbuf_format_get_license(GdkPixbufFormat *
format);
extern gchar **gdk_pixbuf_format_get_mime_types(GdkPixbufFormat *
format);
extern gchar *gdk_pixbuf_format_get_name(GdkPixbufFormat * format);
extern GType gdk_pixbuf_format_get_type(void);

```

```

extern gboolean gdk_pixbuf_format_is_disabled(GdkPixbufFormat *
format);
extern gboolean gdk_pixbuf_format_is_scalable(GdkPixbufFormat *
format);
extern gboolean gdk_pixbuf_format_is_writable(GdkPixbufFormat *
format);
extern void gdk_pixbuf_format_set_disabled(GdkPixbufFormat *
format,
                                     gboolean disabled);
extern int gdk_pixbuf_get_bits_per_sample(const GdkPixbuf * pixbuf);
extern gsize gdk_pixbuf_get_byte_length(const GdkPixbuf * pixbuf);
extern GdkColorspace gdk_pixbuf_get_colorspace(const GdkPixbuf *
pixbuf);
extern GdkPixbufFormat *gdk_pixbuf_get_file_info(const gchar *
filename,
                                     gint * width,
                                     gint * height);
extern GSList *gdk_pixbuf_get_formats(void);
extern gboolean gdk_pixbuf_get_has_alpha(const GdkPixbuf * pixbuf);
extern int gdk_pixbuf_get_height(const GdkPixbuf * pixbuf);
extern int gdk_pixbuf_get_n_channels(const GdkPixbuf * pixbuf);
extern const gchar *gdk_pixbuf_get_option(GdkPixbuf * pixbuf,
                                     const gchar * key);
extern gchar *gdk_pixbuf_get_pixels(const GdkPixbuf * pixbuf);
extern gchar *gdk_pixbuf_get_pixels_with_length(const GdkPixbuf *
pixbuf,
                                     guint * length);
extern int gdk_pixbuf_get_rowstride(const GdkPixbuf * pixbuf);
extern GType gdk_pixbuf_get_type(void);
extern int gdk_pixbuf_get_width(const GdkPixbuf * pixbuf);
extern gboolean gdk_pixbuf_loader_close(GdkPixbufLoader * loader,
GError * *error);
extern GdkPixbufAnimation
*gdk_pixbuf_loader_get_animation(GdkPixbufLoader
* loader);
extern GdkPixbufFormat
*gdk_pixbuf_loader_get_format(GdkPixbufLoader *
loader);
extern GdkPixbuf *gdk_pixbuf_loader_get_pixbuf(GdkPixbufLoader *
loader);
extern GType gdk_pixbuf_loader_get_type(void);
extern GdkPixbufLoader *gdk_pixbuf_loader_new(void);
extern GdkPixbufLoader *gdk_pixbuf_loader_new_with_mime_type(const
char
                                     *mime_type,
                                     GError *
                                     *error);
extern GdkPixbufLoader *gdk_pixbuf_loader_new_with_type(const char
*image_type,
GError * *error);
extern void gdk_pixbuf_loader_set_size(GdkPixbufLoader * loader,
int width,
int height);
extern gboolean gdk_pixbuf_loader_write(GdkPixbufLoader * loader,
const gchar * buf, gsize count,
GError * *error);
extern const guint gdk_pixbuf_major_version;
extern const guint gdk_pixbuf_micro_version;
extern const guint gdk_pixbuf_minor_version;
extern GdkPixbuf *gdk_pixbuf_new(GdkColorspace colorspace,
gboolean has_alpha, int bits_per_sample,
int width, int height);
extern GdkPixbuf *gdk_pixbuf_new_from_data(const gchar * data,
GdkColorspace colorspace,
gboolean has_alpha,
int bits_per_sample, int width,

```

```

        int height, int rowstride,
        GdkPixbufDestroyNotify
        destroy_fn,
        gpointer destroy_fn_data);
extern GdkPixbuf *gdk_pixbuf_new_from_file(const char *filename,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_file_at_scale(const char
*filename,
        int width, int height,
        gboolean
        preserve_aspect_ratio,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_file_at_size(const char
*filename,
        int width, int height,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_inline(gint data_length,
        const guint8 * data,
        gboolean copy_pixels,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_resource(const char
*resource_path,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_resource_at_scale(const char
*resource_path,
        int width,
        int height,
        gboolean

preserve_aspect_ratio,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_stream(GInputStream * stream,
        Gancellable * cancellable,
        GError * *error);
extern void gdk_pixbuf_new_from_stream_async(GInputStream * stream,
        Gancellable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data);
extern GdkPixbuf *gdk_pixbuf_new_from_stream_at_scale(GInputStream
*
        stream, gint width,
        gint height,
        gboolean

preserve_aspect_ratio,
        Gancellable *
        cancellable,
        GError * *error);
extern void gdk_pixbuf_new_from_stream_at_scale_async(GInputStream
*
        stream, gint width,
        gint height,
        gboolean

preserve_aspect_ratio,
        Gancellable *
        cancellable,
        GAsyncReadyCallback
        callback,
        gpointer user_data);
extern GdkPixbuf *gdk_pixbuf_new_from_stream_finish(GAsyncResult *
        async_result,
        GError * *error);
extern GdkPixbuf *gdk_pixbuf_new_from_xpm_data(const char **data);
extern GdkPixbuf *gdk_pixbuf_new_subpixbuf(GdkPixbuf * src_pixbuf,
        int src_x, int src_y, int width,

```

```

        int height);
extern GdkPixbuf *gdk_pixbuf_ref(GdkPixbuf * pixbuf);
extern GdkPixbuf *gdk_pixbuf_rotate_simple(const GdkPixbuf * src,
        GdkPixbufRotation angle);
extern GType gdk_pixbuf_rotation_get_type(void);
extern void gdk_pixbuf_saturate_and_pixelate(const GdkPixbuf * src,
        GdkPixbuf * dest,
        gfloat saturation,
        gboolean pixelate);
extern gboolean gdk_pixbuf_save(GdkPixbuf * pixbuf, const char
*filename,
        const char *type, GError * *error, ...);
extern gboolean gdk_pixbuf_save_to_buffer(GdkPixbuf * pixbuf,
        gchar * *buffer,
        gsize * buffer_size,
        const char *type,
        GError * *error, ...);
extern gboolean gdk_pixbuf_save_to_bufferv(GdkPixbuf * pixbuf,
        gchar * *buffer,
        gsize * buffer_size,
        const char *type,
        char **option_keys,
        char **option_values,
        GError * *error);
extern gboolean gdk_pixbuf_save_to_callback(GdkPixbuf * pixbuf,
        GdkPixbufSaveFunc save_func,
        gpointer user_data,
        const char *type,
        GError * *error, ...);
extern gboolean gdk_pixbuf_save_to_callbackv(GdkPixbuf * pixbuf,
        GdkPixbufSaveFunc save_func,
        gpointer user_data,
        const char *type,
        char **option_keys,
        char **option_values,
        GError * *error);
extern gboolean gdk_pixbuf_save_to_stream(GdkPixbuf * pixbuf,
        GOutputStream * stream,
        const char *type,
        GCancelable * cancellable,
        GError * *error, ...);
extern void gdk_pixbuf_save_to_stream_async(GdkPixbuf * pixbuf,
        GOutputStream * stream,
        const char *type,
        GCancelable * cancellable,
        GAsyncReadyCallback callback,
        gpointer user_data, ...);
extern gboolean gdk_pixbuf_save_to_stream_finish(GAsyncResult *
        async_result,
        GError * *error);
extern gboolean gdk_pixbuf_savev(GdkPixbuf * pixbuf, const char
*filename,
        const char *type, char **option_keys,
        char **option_values, GError * *error);
extern void gdk_pixbuf_scale(const GdkPixbuf * src, GdkPixbuf *
dest,
        int dest_x, int dest_y, int dest_width,
        int dest_height, double offset_x,
        double offset_y, double scale_x,
        double scale_y, GdkInterpType interp_type);
extern GdkPixbuf *gdk_pixbuf_scale_simple(const GdkPixbuf * src,
        int dest_width, int dest_height,
        GdkInterpType interp_type);
extern void gdk_pixbuf_simple_anim_add_frame(GdkPixbufSimpleAnim *
        animation,
        GdkPixbuf * pixbuf);

```

```

extern
gdk_pixbuf_simple_anim_get_loop(GdkPixbufSimpleAnim *
                                animation);
extern GType gdk_pixbuf_simple_anim_get_type(void);
extern GType gdk_pixbuf_simple_anim_iter_get_type(void);
extern GdkPixbufSimpleAnim *gdk_pixbuf_simple_anim_new(gint width,
                                                         gint height,
                                                         gfloat rate);
extern void gdk_pixbuf_simple_anim_set_loop(GdkPixbufSimpleAnim *
                                             animation, gboolean loop);
extern void gdk_pixbuf_unref(GdkPixbuf * pixbuf);
extern const char *gdk_pixbuf_version;

```

17.26.2 gdk-pixbuf-2.0/gdk-pixbuf/gdk-pixdata.h

```

#define GDK_PIXBUF_MAGIC_NUMBER (0x47646b50)
#define GDK_PIXDATA_HEADER_LENGTH (4 + 4 + 4 + 4 + 4 + 4)

typedef struct _GdkPixdata {
    guint32 magic;
    guint32 length;
    guint32 pixdata_type;
    guint32 rowstride;
    guint32 width;
    guint32 height;
    guint8 *pixel_data;
} GdkPixdata;

typedef enum {
    GDK_PIXDATA_DUMP_PIXDATA_STREAM = 0,
    GDK_PIXDATA_DUMP_PIXDATA_STRUCT = 1,
    GDK_PIXDATA_DUMP_MACROS = 2,
    GDK_PIXDATA_DUMP_GTYPES = 0,
    GDK_PIXDATA_DUMP_CTYPES = 256,
    GDK_PIXDATA_DUMP_STATIC = 512,
    GDK_PIXDATA_DUMP_CONST = 1024,
    GDK_PIXDATA_DUMP_RLE_DECODER = 65536
} GdkPixdataDumpType;

typedef enum {
    GDK_PIXDATA_COLOR_TYPE_RGB = 0x01,
    GDK_PIXDATA_COLOR_TYPE_RGBA = 0x02,
    GDK_PIXDATA_COLOR_TYPE_MASK = 0xff,
    GDK_PIXDATA_SAMPLE_WIDTH_8 = 0x01 << 16,
    GDK_PIXDATA_SAMPLE_WIDTH_MASK = 0x0f << 16,
    GDK_PIXDATA_ENCODING_RAW = 0x01 << 24,
    GDK_PIXDATA_ENCODING_RLE = 0x02 << 24,
    GDK_PIXDATA_ENCODING_MASK = 0x0f << 24
} GdkPixdataType;

extern GdkPixbuf *gdk_pixbuf_from_pixdata(const GdkPixdata *
pixdata,
                                           gboolean copy_pixels,
                                           GError * *error);
extern gboolean gdk_pixdata_deserialize(GdkPixdata * pixdata,
                                         guint stream_length,
                                         const guint8 * stream,
                                         GError * *error);
extern gpointer gdk_pixdata_from_pixbuf(GdkPixdata * pixdata,
                                         const GdkPixbuf * pixbuf,
                                         gboolean use_rle);
extern guint8 *gdk_pixdata_serialize(const GdkPixdata * pixdata,
                                     guint * stream_length_p);
extern GString *gdk_pixdata_to_csource(GdkPixdata * pixdata,
                                       const gchar * name,
                                       GdkPixdataDumpType dump_type);

```


17.27 Interfaces for libgdk_pixbuf_xlib-2.0

Table 17-117 defines the library name and shared object name for the libgdk_pixbuf_xlib-2.0 library

Table 17-117 libgdk_pixbuf_xlib-2.0 Definition

Library:	libgdk_pixbuf_xlib-2.0
SONAME:	libgdk_pixbuf_xlib-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gdk-pixbuf 2.26] Gdk-pixbuf 2.26.0 Reference Manual

17.27.1 GDK pixbuf rendering to an X drawable library

17.27.1.1 Interfaces for GDK pixbuf rendering to an X drawable library

An LSB conforming implementation shall provide the generic functions for GDK pixbuf rendering to an X drawable library specified in Table 17-118, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-118 libgdk_pixbuf_xlib-2.0 - GDK pixbuf rendering to an X drawable library Function Interfaces

gdk_pixbuf_xlib_get_from_drawable [Gdk-pixbuf 2.26]	gdk_pixbuf_xlib_init [Gdk-pixbuf 2.26]
gdk_pixbuf_xlib_init_with_depth [Gdk-pixbuf 2.26]	gdk_pixbuf_xlib_render_pixmap_and_mask [Gdk-pixbuf 2.26]
gdk_pixbuf_xlib_render_threshold_alpha [Gdk-pixbuf 2.26]	gdk_pixbuf_xlib_render_to_drawable [Gdk-pixbuf 2.26]
gdk_pixbuf_xlib_render_to_drawable_alpha [Gdk-pixbuf 2.26]	xlib_draw_gray_image [Gdk-pixbuf 2.26]
xlib_draw_indexed_image [Gdk-pixbuf 2.26]	xlib_draw_rgb_32_image [Gdk-pixbuf 2.26]
xlib_draw_rgb_image [Gdk-pixbuf 2.26]	xlib_draw_rgb_image_dithalign [Gdk-pixbuf 2.26]
xlib_rgb_cmap_free [Gdk-pixbuf 2.26]	xlib_rgb_cmap_new [Gdk-pixbuf 2.26]
xlib_rgb_ditherable [Gdk-pixbuf 2.26]	xlib_rgb_gc_set_background [Gdk-pixbuf 2.26]
xlib_rgb_gc_set_foreground [Gdk-pixbuf 2.26]	xlib_rgb_get_cmap [Gdk-pixbuf 2.26]
xlib_rgb_get_depth [Gdk-pixbuf 2.26]	xlib_rgb_get_display [Gdk-pixbuf 2.26]
xlib_rgb_get_screen [Gdk-pixbuf 2.26]	xlib_rgb_get_visual [Gdk-pixbuf 2.26]

xlib_rgb_get_visual_info [Gdk-pixbuf 2.26]	xlib_rgb_init [Gdk-pixbuf 2.26]
xlib_rgb_init_with_depth [Gdk-pixbuf 2.26]	xlib_rgb_set_install [Gdk-pixbuf 2.26]
xlib_rgb_set_min_colors [Gdk-pixbuf 2.26]	xlib_rgb_set_verbose [Gdk-pixbuf 2.26]
xlib_rgb_xpixel_from_rgb [Gdk-pixbuf 2.26]	

17.28 Data Definitions for libgdk_pixbuf_xlib-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libgdk_pixbuf_xlib-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.28.1 gdk-pixbuf-2.0/gdk-pixbuf-xlib/gdk-pixbuf-xlib.h

```
typedef enum {
    XLIB_RGB_DITHER_NONE = 0,
    XLIB_RGB_DITHER_NORMAL = 1,
    XLIB_RGB_DITHER_MAX = 2
} XlibRgbDither;
typedef struct _XlibRgbCmap {
    unsigned int colors[256];
    unsigned char lut[256];
} XlibRgbCmap;
extern GdkPixbuf *gdk_pixbuf_xlib_get_from_drawable(GdkPixbuf *
dest,
                                                    Drawable src,
                                                    Colormap cmap,
                                                    Visual * visual,
                                                    int src_x, int src_y,
                                                    int dest_x, int dest_y,
                                                    int width, int height);
extern void gdk_pixbuf_xlib_init(Display * display, int screen_num);
extern void gdk_pixbuf_xlib_init_with_depth(Display * display,
                                                    int screen_num, int
prefDepth);
extern void gdk_pixbuf_xlib_render_pixmap_and_mask(GdkPixbuf *
pixbuf,
                                                    Pixmap * pixmap_return,
                                                    Pixmap * mask_return,
                                                    int alpha_threshold);
```

```

extern void gdk_pixbuf_xlib_render_threshold_alpha(GdkPixbuf *
pixbuf,
                                Pixmap bitmap,
                                int src_x, int src_y,
                                int dest_x, int dest_y,
                                int width, int height,
                                int alpha_threshold);
extern void gdk_pixbuf_xlib_render_to_drawable(GdkPixbuf * pixbuf,
                                Drawable drawable, GC gc,
                                int src_x, int src_y,
                                int dest_x, int dest_y,
                                int width, int height,
                                XlibRgbDither dither,
                                int x_dither, int y_dither);
extern void gdk_pixbuf_xlib_render_to_drawable_alpha(GdkPixbuf *
pixbuf,
                                Drawable drawable,
                                int src_x, int src_y,
                                int dest_x,
                                int dest_y, int width,
                                int height,
                                GdkPixbufAlphaMode
                                alpha_mode,
                                int alpha_threshold,
                                XlibRgbDither dither,
                                int x_dither,
                                int y_dither);
extern void xlib_draw_gray_image(Drawable drawable, GC gc, int x,
int y,
                                int width, int height, XlibRgbDither
dith,
                                unsigned char *buf, int rowstride);
extern void xlib_draw_indexed_image(Drawable drawable, GC gc, int
x, int y,
                                int width, int height,
                                XlibRgbDither dith, unsigned char
*buf,
                                int rowstride, XlibRgbCmap * cmap);
extern void xlib_draw_rgb_32_image(Drawable drawable, GC gc, int x,
int y,
                                int width, int height,
                                XlibRgbDither dith, unsigned char
*buf,
                                int rowstride);
extern void xlib_draw_rgb_image(Drawable drawable, GC gc, int x,
int y,
                                int width, int height, XlibRgbDither
dith,
                                unsigned char *rgb_buf, int rowstride);
extern void xlib_draw_rgb_image_dithalign(Drawable drawable, GC gc,
int x,
                                int y, int width, int height,
                                XlibRgbDither dith,
                                unsigned char *rgb_buf,
                                int rowstride, int xdith,
                                int ydith);
extern void xlib_rgb_cmap_free(XlibRgbCmap * cmap);
extern XlibRgbCmap *xlib_rgb_cmap_new(guint32 * colors, int
n_colors);
extern int xlib_rgb_ditherable(void);
extern void xlib_rgb_gc_set_background(GC gc, guint32 rgb);
extern void xlib_rgb_gc_set_foreground(GC gc, guint32 rgb);
extern Colormap xlib_rgb_get_cmap(void);
extern int xlib_rgb_get_depth(void);
extern Display *xlib_rgb_get_display(void);
extern Screen *xlib_rgb_get_screen(void);

```

```

extern Visual *xlib_rgb_get_visual(void);
extern XVisualInfo *xlib_rgb_get_visual_info(void);
extern void xlib_rgb_init(Display * display, Screen * screen);
extern void xlib_rgb_init_with_depth(Display * display, Screen *
                                     screen,
                                     int prefDepth);
extern void xlib_rgb_set_install(int install);
extern void xlib_rgb_set_min_colors(int min_colors);
extern void xlib_rgb_set_verbose(int verbose);
extern long unsigned int xlib_rgb_xpixel_from_rgb(guint32 rgb);

```

17.29 Interfaces for libgdk-x11-2.0

Table 17-119 defines the library name and shared object name for the libgdk-x11-2.0 library

Table 17-119 libgdk-x11-2.0 Definition

Library:	libgdk-x11-2.0
SONAME:	libgdk-x11-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gdk 2.10] Gdk 2.10.14 Reference Manual

[Gobject 2.32] Gobject 2.32 Reference Manual

17.29.1 GTK Drawing toolkit

17.29.1.1 Interfaces for GTK Drawing toolkit

An LSB conforming implementation shall provide the generic functions for GTK Drawing toolkit specified in Table 17-120, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-120 libgdk-x11-2.0 - GTK Drawing toolkit Function Interfaces

gdk_add_client_message_filter [Gdk 2.10]	gdk_atom_intern [Gdk 2.10]
gdk_atom_intern_static_string [Gdk 2.10]	gdk_atom_name [Gdk 2.10]
gdk_axis_use_get_type [Gobject 2.32]	gdk_beep [Gdk 2.10]
gdk_bitmap_create_from_data [Gdk 2.10]	gdk_byte_order_get_type [Gobject 2.32]
gdk_cairo_create [Gdk 2.10]	gdk_cairo_rectangle [Gdk 2.10]
gdk_cairo_region [Gdk 2.10]	gdk_cairo_set_source_color [Gdk 2.10]
gdk_cairo_set_source_pixbuf [Gdk 2.10]	gdk_cairo_set_source_pixmap [Gdk 2.10]
gdk_cap_style_get_type [Gobject 2.32]	gdk_color_copy [Gdk 2.10]
gdk_color_equal [Gdk 2.10]	gdk_color_free [Gdk 2.10]
gdk_color_get_type [Gobject 2.32]	gdk_color_hash [Gdk 2.10]

gdk_color_parse [Gdk 2.10]	gdk_colormap_alloc_color [Gdk 2.10]
gdk_colormap_alloc_colors [Gdk 2.10]	gdk_colormap_free_colors [Gdk 2.10]
gdk_colormap_get_screen [Gdk 2.10]	gdk_colormap_get_system [Gdk 2.10]
gdk_colormap_get_type [Gobject 2.32]	gdk_colormap_get_visual [Gdk 2.10]
gdk_colormap_new [Gdk 2.10]	gdk_colormap_query_color [Gdk 2.10]
gdk_crossing_mode_get_type [Gobject 2.32]	gdk_cursor_get_display [Gdk 2.10]
gdk_cursor_get_image [Gdk 2.10]	gdk_cursor_get_type [Gobject 2.32]
gdk_cursor_new [Gdk 2.10]	gdk_cursor_new_for_display [Gdk 2.10]
gdk_cursor_new_from_name [Gdk 2.10]	gdk_cursor_new_from_pixbuf [Gdk 2.10]
gdk_cursor_new_from_pixmap [Gdk 2.10]	gdk_cursor_ref [Gdk 2.10]
gdk_cursor_type_get_type [Gobject 2.32]	gdk_cursor_unref [Gdk 2.10]
gdk_device_free_history [Gdk 2.10]	gdk_device_get_axis [Gdk 2.10]
gdk_device_get_core_pointer [Gdk 2.10]	gdk_device_get_history [Gdk 2.10]
gdk_device_get_state [Gdk 2.10]	gdk_device_get_type [Gobject 2.32]
gdk_device_set_axis_use [Gdk 2.10]	gdk_device_set_key [Gdk 2.10]
gdk_device_set_mode [Gdk 2.10]	gdk_device_set_source [Gdk 2.10]
gdk_devices_list [Gdk 2.10]	gdk_display_add_client_message_filter [Gdk 2.10]
gdk_display_beep [Gdk 2.10]	gdk_display_close [Gdk 2.10]
gdk_display_flush [Gdk 2.10]	gdk_display_get_core_pointer [Gdk 2.10]
gdk_display_get_default [Gdk 2.10]	gdk_display_get_default_cursor_size [Gdk 2.10]
gdk_display_get_default_group [Gdk 2.10]	gdk_display_get_default_screen [Gdk 2.10]
gdk_display_get_event [Gdk 2.10]	gdk_display_get_maximal_cursor_size [Gdk 2.10]
gdk_display_get_n_screens [Gdk 2.10]	gdk_display_get_name [Gdk 2.10]
gdk_display_get_pointer [Gdk 2.10]	gdk_display_get_screen [Gdk 2.10]

gdk_display_get_type [Gobject 2.32]	gdk_display_get_window_at_pointer [Gdk 2.10]
gdk_display_keyboard_ungrab [Gdk 2.10]	gdk_display_list_devices [Gdk 2.10]
gdk_display_manager_get [Gdk 2.10]	gdk_display_manager_get_default_display [Gdk 2.10]
gdk_display_manager_get_type [Gobject 2.32]	gdk_display_manager_list_displays [Gdk 2.10]
gdk_display_manager_set_default_display [Gdk 2.10]	gdk_display_open [Gdk 2.10]
gdk_display_peek_event [Gdk 2.10]	gdk_display_pointer_is_grabbed [Gdk 2.10]
gdk_display_pointer_ungrab [Gdk 2.10]	gdk_display_put_event [Gdk 2.10]
gdk_display_request_selection_notification [Gdk 2.10]	gdk_display_set_double_click_distance [Gdk 2.10]
gdk_display_set_double_click_time [Gdk 2.10]	gdk_display_set_pointer_hooks [Gdk 2.10]
gdk_display_store_clipboard [Gdk 2.10]	gdk_display_supports_clipboard_persistence [Gdk 2.10]
gdk_display_supports_cursor_alpha [Gdk 2.10]	gdk_display_supports_cursor_color [Gdk 2.10]
gdk_display_supports_input_shapes [Gdk 2.10]	gdk_display_supports_selection_notification [Gdk 2.10]
gdk_display_supports_shapes [Gdk 2.10]	gdk_display_sync [Gdk 2.10]
gdk_display_warp_pointer [Gdk 2.10]	gdk_drag_abort [Gdk 2.10]
gdk_drag_action_get_type [Gobject 2.32]	gdk_drag_begin [Gdk 2.10]
gdk_drag_context_get_type [Gobject 2.32]	gdk_drag_context_new [Gdk 2.10]
gdk_drag_drop [Gdk 2.10]	gdk_drag_drop_succeeded [Gdk 2.10]
gdk_drag_find_window [Gdk 2.10]	gdk_drag_find_window_for_screen [Gdk 2.10]
gdk_drag_get_protocol [Gdk 2.10]	gdk_drag_get_protocol_for_display [Gdk 2.10]
gdk_drag_get_selection [Gdk 2.10]	gdk_drag_motion [Gdk 2.10]
gdk_drag_protocol_get_type [Gobject 2.32]	gdk_drag_status [Gdk 2.10]
gdk_draw_arc [Gdk 2.10]	gdk_draw_drawable [Gdk 2.10]

gdk_draw_glyphs [Gdk 2.10]	gdk_draw_glyphs_transformed [Gdk 2.10]
gdk_draw_gray_image [Gdk 2.10]	gdk_draw_image [Gdk 2.10]
gdk_draw_indexed_image [Gdk 2.10]	gdk_draw_layout [Gdk 2.10]
gdk_draw_layout_line [Gdk 2.10]	gdk_draw_layout_line_with_colors [Gdk 2.10]
gdk_draw_layout_with_colors [Gdk 2.10]	gdk_draw_line [Gdk 2.10]
gdk_draw_lines [Gdk 2.10]	gdk_draw_pixbuf [Gdk 2.10]
gdk_draw_point [Gdk 2.10]	gdk_draw_points [Gdk 2.10]
gdk_draw_polygon [Gdk 2.10]	gdk_draw_rectangle [Gdk 2.10]
gdk_draw_rgb_32_image [Gdk 2.10]	gdk_draw_rgb_32_image_dithalign [Gdk 2.10]
gdk_draw_rgb_image [Gdk 2.10]	gdk_draw_rgb_image_dithalign [Gdk 2.10]
gdk_draw_segments [Gdk 2.10]	gdk_draw_trapezoids [Gdk 2.10]
gdk_drawable_copy_to_image [Gdk 2.10]	gdk_drawable_get_clip_region [Gdk 2.10]
gdk_drawable_get_colormap [Gdk 2.10]	gdk_drawable_get_depth [Gdk 2.10]
gdk_drawable_get_display [Gdk 2.10]	gdk_drawable_get_image [Gdk 2.10]
gdk_drawable_get_screen [Gdk 2.10]	gdk_drawable_get_size [Gdk 2.10]
gdk_drawable_get_type [Gobject 2.32]	gdk_drawable_get_visible_region [Gdk 2.10]
gdk_drawable_get_visual [Gdk 2.10]	gdk_drawable_set_colormap [Gdk 2.10]
gdk_drop_finish [Gdk 2.10]	gdk_drop_reply [Gdk 2.10]
gdk_error_trap_pop [Gdk 2.10]	gdk_error_trap_push [Gdk 2.10]
gdk_event_copy [Gdk 2.10]	gdk_event_free [Gdk 2.10]
gdk_event_get [Gdk 2.10]	gdk_event_get_axis [Gdk 2.10]
gdk_event_get_coords [Gdk 2.10]	gdk_event_get_graphics_expose [Gdk 2.10]
gdk_event_get_root_coords [Gdk 2.10]	gdk_event_get_screen [Gdk 2.10]
gdk_event_get_state [Gdk 2.10]	gdk_event_get_time [Gdk 2.10]
gdk_event_get_type [Gobject 2.32]	gdk_event_handler_set [Gdk 2.10]

gdk_event_mask_get_type [Gobject 2.32]	gdk_event_new [Gdk 2.10]
gdk_event_peek [Gdk 2.10]	gdk_event_put [Gdk 2.10]
gdk_event_send_client_message [Gdk 2.10]	gdk_event_send_client_message_for_display [Gdk 2.10]
gdk_event_send_clientmessage_toall [Gdk 2.10]	gdk_event_set_screen [Gdk 2.10]
gdk_event_type_get_type [Gobject 2.32]	gdk_events_pending [Gdk 2.10]
gdk_extension_mode_get_type [Gobject 2.32]	gdk_fill_get_type [Gobject 2.32]
gdk_fill_rule_get_type [Gobject 2.32]	gdk_filter_return_get_type [Gobject 2.32]
gdk_flush [Gdk 2.10]	gdk_font_type_get_type [Gobject 2.32]
gdk_free_compound_text [Gdk 2.10]	gdk_free_text_list [Gdk 2.10]
gdk_function_get_type [Gobject 2.32]	gdk_gc_copy [Gdk 2.10]
gdk_gc_get_colormap [Gdk 2.10]	gdk_gc_get_screen [Gdk 2.10]
gdk_gc_get_type [Gobject 2.32]	gdk_gc_get_values [Gdk 2.10]
gdk_gc_new [Gdk 2.10]	gdk_gc_new_with_values [Gdk 2.10]
gdk_gc_offset [Gdk 2.10]	gdk_gc_set_background [Gdk 2.10]
gdk_gc_set_clip_mask [Gdk 2.10]	gdk_gc_set_clip_origin [Gdk 2.10]
gdk_gc_set_clip_rectangle [Gdk 2.10]	gdk_gc_set_clip_region [Gdk 2.10]
gdk_gc_set_colormap [Gdk 2.10]	gdk_gc_set_dashes [Gdk 2.10]
gdk_gc_set_exposures [Gdk 2.10]	gdk_gc_set_fill [Gdk 2.10]
gdk_gc_set_foreground [Gdk 2.10]	gdk_gc_set_function [Gdk 2.10]
gdk_gc_set_line_attributes [Gdk 2.10]	gdk_gc_set_rgb_bg_color [Gdk 2.10]
gdk_gc_set_rgb_fg_color [Gdk 2.10]	gdk_gc_set_stipple [Gdk 2.10]
gdk_gc_set_subwindow [Gdk 2.10]	gdk_gc_set_tile [Gdk 2.10]
gdk_gc_set_ts_origin [Gdk 2.10]	gdk_gc_set_values [Gdk 2.10]
gdk_gc_values_mask_get_type [Gobject 2.32]	gdk_get_default_root_window [Gdk 2.10]
gdk_get_display [Gdk 2.10]	gdk_get_display_arg_name [Gdk 2.10]
gdk_get_program_class [Gdk 2.10]	gdk_get_show_events [Gdk 2.10]
gdk_grab_status_get_type [Gobject 2.32]	gdk_gravity_get_type [Gobject 2.32]

gdk_image_get_colormap [Gdk 2.10]	gdk_image_get_pixel [Gdk 2.10]
gdk_image_get_type [Gobject 2.32]	gdk_image_new [Gdk 2.10]
gdk_image_put_pixel [Gdk 2.10]	gdk_image_set_colormap [Gdk 2.10]
gdk_image_type_get_type [Gobject 2.32]	gdk_init [Gdk 2.10]
gdk_init_check [Gdk 2.10]	gdk_input_condition_get_type [Gobject 2.32]
gdk_input_mode_get_type [Gobject 2.32]	gdk_input_set_extension_events [Gdk 2.10]
gdk_input_source_get_type [Gobject 2.32]	gdk_join_style_get_type [Gobject 2.32]
gdk_keyboard_grab [Gdk 2.10]	gdk_keyboard_ungrab [Gdk 2.10]
gdk_keymap_get_default [Gdk 2.10]	gdk_keymap_get_direction [Gdk 2.10]
gdk_keymap_get_entries_for_keycode [Gdk 2.10]	gdk_keymap_get_entries_for_keyval [Gdk 2.10]
gdk_keymap_get_for_display [Gdk 2.10]	gdk_keymap_get_type [Gobject 2.32]
gdk_keymap_lookup_key [Gdk 2.10]	gdk_keymap_translate_keyboard_state [Gdk 2.10]
gdk_keyval_convert_case [Gdk 2.10]	gdk_keyval_from_name [Gdk 2.10]
gdk_keyval_is_lower [Gdk 2.10]	gdk_keyval_is_upper [Gdk 2.10]
gdk_keyval_name [Gdk 2.10]	gdk_keyval_to_lower [Gdk 2.10]
gdk_keyval_to_unicode [Gdk 2.10]	gdk_keyval_to_upper [Gdk 2.10]
gdk_line_style_get_type [Gobject 2.32]	gdk_list_visuals [Gdk 2.10]
gdk_modifier_type_get_type [Gobject 2.32]	gdk_net_wm_supports [Gdk 2.10]
gdk_notify_startup_complete [Gdk 2.10]	gdk_notify_type_get_type [Gobject 2.32]
gdk_overlap_type_get_type [Gobject 2.32]	gdk_owner_change_get_type [Gobject 2.32]
gdk_pango_attr_embossed_new [Gdk 2.10]	gdk_pango_attr_stipple_new [Gdk 2.10]
gdk_pango_context_get [Gdk 2.10]	gdk_pango_context_get_for_screen [Gdk 2.10]
gdk_pango_layout_get_clip_region [Gdk 2.10]	gdk_pango_layout_line_get_clip_region [Gdk 2.10]
gdk_pango_renderer_get_default [Gdk 2.10]	gdk_pango_renderer_get_type [Gobject 2.32]

gdk_pango_renderer_new [Gdk 2.10]	gdk_pango_renderer_set_drawable [Gdk 2.10]
gdk_pango_renderer_set_gc [Gdk 2.10]	gdk_pango_renderer_set_override_color [Gdk 2.10]
gdk_pango_renderer_set_stipple [Gdk 2.10]	gdk_parse_args [Gdk 2.10]
gdk_pixbuf_get_from_drawable [Gdk 2.10]	gdk_pixbuf_get_from_image [Gdk 2.10]
gdk_pixbuf_render_pixmap_and_mask [Gdk 2.10]	gdk_pixbuf_render_pixmap_and_mask_for_colormap [Gdk 2.10]
gdk_pixbuf_render_threshold_alpha [Gdk 2.10]	gdk_pixmap_colormap_create_from_xpm [Gdk 2.10]
gdk_pixmap_colormap_create_from_xpm_d [Gdk 2.10]	gdk_pixmap_create_from_data [Gdk 2.10]
gdk_pixmap_create_from_xpm [Gdk 2.10]	gdk_pixmap_create_from_xpm_d [Gdk 2.10]
gdk_pixmap_foreign_new [Gdk 2.10]	gdk_pixmap_foreign_new_for_display [Gdk 2.10]
gdk_pixmap_foreign_new_for_screen [Gdk 2.10]	gdk_pixmap_get_type [Gobject 2.32]
gdk_pixmap_lookup [Gdk 2.10]	gdk_pixmap_lookup_for_display [Gdk 2.10]
gdk_pixmap_new [Gdk 2.10]	gdk_pointer_grab [Gdk 2.10]
gdk_pointer_is_grabbed [Gdk 2.10]	gdk_pointer_ungrab [Gdk 2.10]
gdk_prop_mode_get_type [Gobject 2.32]	gdk_property_change [Gdk 2.10]
gdk_property_delete [Gdk 2.10]	gdk_property_get [Gdk 2.10]
gdk_property_state_get_type [Gobject 2.32]	gdk_query_depths [Gdk 2.10]
gdk_query_visual_types [Gdk 2.10]	gdk_rectangle_get_type [Gobject 2.32]
gdk_rectangle_intersect [Gdk 2.10]	gdk_rectangle_union [Gdk 2.10]
gdk_region_copy [Gdk 2.10]	gdk_region_destroy [Gdk 2.10]
gdk_region_empty [Gdk 2.10]	gdk_region_equal [Gdk 2.10]
gdk_region_get_clipbox [Gdk 2.10]	gdk_region_get_rectangles [Gdk 2.10]
gdk_region_intersect [Gdk 2.10]	gdk_region_new [Gdk 2.10]
gdk_region_offset [Gdk 2.10]	gdk_region_point_in [Gdk 2.10]
gdk_region_polygon [Gdk 2.10]	gdk_region_rect_in [Gdk 2.10]
gdk_region_rectangle [Gdk 2.10]	gdk_region_shrink [Gdk 2.10]

gdk_region_spans_intersect_foreach [Gdk 2.10]	gdk_region_subtract [Gdk 2.10]
gdk_region_union [Gdk 2.10]	gdk_region_union_with_rect [Gdk 2.10]
gdk_region_xor [Gdk 2.10]	gdk_rgb_cmap_free [Gdk 2.10]
gdk_rgb_cmap_new [Gdk 2.10]	gdk_rgb_colormap_ditherable [Gdk 2.10]
gdk_rgb_dither_get_type [Gobject 2.32]	gdk_rgb_ditherable [Gdk 2.10]
gdk_rgb_find_color [Gdk 2.10]	gdk_rgb_get_colormap [Gdk 2.10]
gdk_rgb_get_visual [Gdk 2.10]	gdk_rgb_set_install [Gdk 2.10]
gdk_rgb_set_min_colors [Gdk 2.10]	gdk_rgb_set_verbose [Gdk 2.10]
gdk_screen_broadcast_client_message [Gdk 2.10]	gdk_screen_get_active_window [Gdk 2.10]
gdk_screen_get_default [Gdk 2.10]	gdk_screen_get_default_colormap [Gdk 2.10]
gdk_screen_get_display [Gdk 2.10]	gdk_screen_get_font_options [Gdk 2.10]
gdk_screen_get_height [Gdk 2.10]	gdk_screen_get_height_mm [Gdk 2.10]
gdk_screen_get_monitor_at_point [Gdk 2.10]	gdk_screen_get_monitor_at_window [Gdk 2.10]
gdk_screen_get_monitor_geometry [Gdk 2.10]	gdk_screen_get_n_monitors [Gdk 2.10]
gdk_screen_get_number [Gdk 2.10]	gdk_screen_get_resolution [Gdk 2.10]
gdk_screen_get_rgb_colormap [Gdk 2.10]	gdk_screen_get_rgb_visual [Gdk 2.10]
gdk_screen_get_rgba_colormap [Gdk 2.10]	gdk_screen_get_rgba_visual [Gdk 2.10]
gdk_screen_get_root_window [Gdk 2.10]	gdk_screen_get_setting [Gdk 2.10]
gdk_screen_get_system_colormap [Gdk 2.10]	gdk_screen_get_system_visual [Gdk 2.10]
gdk_screen_get_toplevel_windows [Gdk 2.10]	gdk_screen_get_type [Gobject 2.32]
gdk_screen_get_width [Gdk 2.10]	gdk_screen_get_width_mm [Gdk 2.10]
gdk_screen_get_window_stack [Gdk 2.10]	gdk_screen_height [Gdk 2.10]

gdk_screen_height_mm [Gdk 2.10]	gdk_screen_is_composited [Gdk 2.10]
gdk_screen_list_visuals [Gdk 2.10]	gdk_screen_make_display_name [Gdk 2.10]
gdk_screen_set_default_colormap [Gdk 2.10]	gdk_screen_set_font_options [Gdk 2.10]
gdk_screen_set_resolution [Gdk 2.10]	gdk_screen_width [Gdk 2.10]
gdk_screen_width_mm [Gdk 2.10]	gdk_scroll_direction_get_type [Gobject 2.32]
gdk_selection_convert [Gdk 2.10]	gdk_selection_owner_get [Gdk 2.10]
gdk_selection_owner_get_for_display [Gdk 2.10]	gdk_selection_owner_set [Gdk 2.10]
gdk_selection_owner_set_for_display [Gdk 2.10]	gdk_selection_property_get [Gdk 2.10]
gdk_selection_send_notify [Gdk 2.10]	gdk_selection_send_notify_for_display [Gdk 2.10]
gdk_set_double_click_time [Gdk 2.10]	gdk_set_locale [Gdk 2.10]
gdk_set_pointer_hooks [Gdk 2.10]	gdk_set_program_class [Gdk 2.10]
gdk_set_show_events [Gdk 2.10]	gdk_set_sm_client_id [Gdk 2.10]
gdk_setting_action_get_type [Gobject 2.32]	gdk_setting_get [Gdk 2.10]
gdk_spawn_command_line_on_screen [Gdk 2.10]	gdk_spawn_on_screen [Gdk 2.10]
gdk_spawn_on_screen_with_pipes [Gdk 2.10]	gdk_status_get_type [Gobject 2.32]
gdk_string_to_compound_text [Gdk 2.10]	gdk_string_to_compound_text_for_display [Gdk 2.10]
gdk_subwindow_mode_get_type [Gobject 2.32]	gdk_synthesize_window_state [Gdk 2.10]
gdk_text_property_to_text_list [Gdk 2.10]	gdk_text_property_to_text_list_for_display [Gdk 2.10]
gdk_text_property_to_utf8_list [Gdk 2.10]	gdk_text_property_to_utf8_list_for_display [Gdk 2.10]
gdk_threads_enter [Gdk 2.10]	gdk_threads_init [Gdk 2.10]
gdk_threads_leave [Gdk 2.10]	gdk_threads_set_lock_functions [Gdk 2.10]
gdk_unicode_to_keyval [Gdk 2.10]	gdk_utf8_to_compound_text [Gdk 2.10]
gdk_utf8_to_compound_text_for_display [Gdk 2.10]	gdk_utf8_to_string_target [Gdk 2.10]

gdk_visibility_state_get_type [Gobject 2.32]	gdk_visual_get_best [Gdk 2.10]
gdk_visual_get_best_depth [Gdk 2.10]	gdk_visual_get_best_type [Gdk 2.10]
gdk_visual_get_best_with_both [Gdk 2.10]	gdk_visual_get_best_with_depth [Gdk 2.10]
gdk_visual_get_best_with_type [Gdk 2.10]	gdk_visual_get_screen [Gdk 2.10]
gdk_visual_get_system [Gdk 2.10]	gdk_visual_get_type [Gobject 2.32]
gdk_visual_type_get_type [Gobject 2.32]	gdk_window_add_filter [Gdk 2.10]
gdk_window_at_pointer [Gdk 2.10]	gdk_window_attributes_type_get_type [Gobject 2.32]
gdk_window_begin_move_drag [Gdk 2.10]	gdk_window_begin_paint_rect [Gdk 2.10]
gdk_window_begin_paint_region [Gdk 2.10]	gdk_window_begin_resize_drag [Gdk 2.10]
gdk_window_class_get_type [Gobject 2.32]	gdk_window_clear [Gdk 2.10]
gdk_window_clear_area [Gdk 2.10]	gdk_window_clear_area_e [Gdk 2.10]
gdk_window_configure_finished [Gdk 2.10]	gdk_window_constrain_size [Gdk 2.10]
gdk_window_deiconify [Gdk 2.10]	gdk_window_destroy [Gdk 2.10]
gdk_window_destroy_notify [Gdk 2.10]	gdk_window_edge_get_type [Gobject 2.32]
gdk_window_enable_synchronized_configure [Gdk 2.10]	gdk_window_end_paint [Gdk 2.10]
gdk_window_focus [Gdk 2.10]	gdk_window_foreign_new [Gdk 2.10]
gdk_window_foreign_new_for_display [Gdk 2.10]	gdk_window_freeze_updates [Gdk 2.10]
gdk_window_fullscreen [Gdk 2.10]	gdk_window_get_children [Gdk 2.10]
gdk_window_get_decorations [Gdk 2.10]	gdk_window_get_events [Gdk 2.10]
gdk_window_get_frame_extents [Gdk 2.10]	gdk_window_get_geometry [Gdk 2.10]
gdk_window_get_group [Gdk 2.10]	gdk_window_get_internal_paint_info [Gdk 2.10]
gdk_window_get_origin [Gdk 2.10]	gdk_window_get_parent [Gdk 2.10]

gdk_window_get_pointer [Gdk 2.10]	gdk_window_get_position [Gdk 2.10]
gdk_window_get_root_origin [Gdk 2.10]	gdk_window_get_state [Gdk 2.10]
gdk_window_get_toplevel [Gdk 2.10]	gdk_window_get_toplevels [Gdk 2.10]
gdk_window_get_type_hint [Gdk 2.10]	gdk_window_get_update_area [Gdk 2.10]
gdk_window_get_user_data [Gdk 2.10]	gdk_window_get_window_type [Gdk 2.10]
gdk_window_hide [Gdk 2.10]	gdk_window_hints_get_type [Gobject 2.32]
gdk_window_iconify [Gdk 2.10]	gdk_window_input_shape_combine_mask [Gdk 2.10]
gdk_window_input_shape_combine_region [Gdk 2.10]	gdk_window_invalidate_maybe_recurse [Gdk 2.10]
gdk_window_invalidate_rect [Gdk 2.10]	gdk_window_invalidate_region [Gdk 2.10]
gdk_window_is_viewable [Gdk 2.10]	gdk_window_is_visible [Gdk 2.10]
gdk_window_lookup [Gdk 2.10]	gdk_window_lookup_for_display [Gdk 2.10]
gdk_window_lower [Gdk 2.10]	gdk_window_maximize [Gdk 2.10]
gdk_window_merge_child_input_shapes [Gdk 2.10]	gdk_window_merge_child_shapes [Gdk 2.10]
gdk_window_move [Gdk 2.10]	gdk_window_move_region [Gdk 2.10]
gdk_window_move_resize [Gdk 2.10]	gdk_window_new [Gdk 2.10]
gdk_window_object_get_type [Gobject 2.32]	gdk_window_peek_children [Gdk 2.10]
gdk_window_process_all_updates [Gdk 2.10]	gdk_window_process_updates [Gdk 2.10]
gdk_window_raise [Gdk 2.10]	gdk_window_register_dnd [Gdk 2.10]
gdk_window_remove_filter [Gdk 2.10]	gdk_window_reparent [Gdk 2.10]
gdk_window_resize [Gdk 2.10]	gdk_window_scroll [Gdk 2.10]
gdk_window_set_accept_focus [Gdk 2.10]	gdk_window_set_back_pixmap [Gdk 2.10]
gdk_window_set_background [Gdk 2.10]	gdk_window_set_child_input_shapes [Gdk 2.10]

gdk_window_set_child_shapes [Gdk 2.10]	gdk_window_set_cursor [Gdk 2.10]
gdk_window_set_debug_updates [Gdk 2.10]	gdk_window_set_decorations [Gdk 2.10]
gdk_window_set_events [Gdk 2.10]	gdk_window_set_focus_on_map [Gdk 2.10]
gdk_window_set_functions [Gdk 2.10]	gdk_window_set_geometry_hints [Gdk 2.10]
gdk_window_set_group [Gdk 2.10]	gdk_window_set_icon [Gdk 2.10]
gdk_window_set_icon_list [Gdk 2.10]	gdk_window_set_icon_name [Gdk 2.10]
gdk_window_set_keep_above [Gdk 2.10]	gdk_window_set_keep_below [Gdk 2.10]
gdk_window_set_modal_hint [Gdk 2.10]	gdk_window_set_override_redirect [Gdk 2.10]
gdk_window_set_role [Gdk 2.10]	gdk_window_set_skip_pager_hint [Gdk 2.10]
gdk_window_set_skip_taskbar_hint [Gdk 2.10]	gdk_window_set_static_gravities [Gdk 2.10]
gdk_window_set_title [Gdk 2.10]	gdk_window_set_transient_for [Gdk 2.10]
gdk_window_set_type_hint [Gdk 2.10]	gdk_window_set_urgency_hint [Gdk 2.10]
gdk_window_set_user_data [Gdk 2.10]	gdk_window_shape_combine_mask [Gdk 2.10]
gdk_window_shape_combine_region [Gdk 2.10]	gdk_window_show [Gdk 2.10]
gdk_window_show_unraised [Gdk 2.10]	gdk_window_state_get_type [Gobject 2.32]
gdk_window_stick [Gdk 2.10]	gdk_window_thaw_updates [Gdk 2.10]
gdk_window_type_get_type [Gobject 2.32]	gdk_window_type_hint_get_type [Gobject 2.32]
gdk_window_unfullscreen [Gdk 2.10]	gdk_window_unmaximize [Gdk 2.10]
gdk_window_unstick [Gdk 2.10]	gdk_window_withdraw [Gdk 2.10]
gdk_wm_decoration_get_type [Gobject 2.32]	gdk_wm_function_get_type [Gobject 2.32]
gdk_x11_atom_to_xatom [Gdk 2.10]	gdk_x11_atom_to_xatom_for_display [Gdk 2.10]
gdk_x11_colormap_foreign_new [Gdk 2.10]	gdk_x11_colormap_get_xcolormap [Gdk 2.10]

gdk_x11_colormap_get_xdisplay [Gdk 2.10]	gdk_x11_cursor_get_xcursor [Gdk 2.10]
gdk_x11_cursor_get_xdisplay [Gdk 2.10]	gdk_x11_display_get_user_time [Gdk 2.10]
gdk_x11_display_get_xdisplay [Gdk 2.10]	gdk_x11_display_grab [Gdk 2.10]
gdk_x11_display_set_cursor_theme [Gdk 2.10]	gdk_x11_display_ungrab [Gdk 2.10]
gdk_x11_drawable_get_xdisplay [Gdk 2.10]	gdk_x11_drawable_get_xid [Gdk 2.10]
gdk_x11_gc_get_xdisplay [Gdk 2.10]	gdk_x11_gc_get_xgc [Gdk 2.10]
gdk_x11_get_default_root_xwindow [Gdk 2.10]	gdk_x11_get_default_screen [Gdk 2.10]
gdk_x11_get_default_xdisplay [Gdk 2.10]	gdk_x11_get_server_time [Gdk 2.10]
gdk_x11_get_xatom_by_name [Gdk 2.10]	gdk_x11_get_xatom_by_name_for_display [Gdk 2.10]
gdk_x11_get_xatom_name [Gdk 2.10]	gdk_x11_get_xatom_name_for_display [Gdk 2.10]
gdk_x11_grab_server [Gdk 2.10]	gdk_x11_image_get_xdisplay [Gdk 2.10]
gdk_x11_image_get_ximage [Gdk 2.10]	gdk_x11_lookup_xdisplay [Gdk 2.10]
gdk_x11_register_standard_event_type [Gdk 2.10]	gdk_x11_screen_get_screen_number [Gdk 2.10]
gdk_x11_screen_get_window_manager_name [Gdk 2.10]	gdk_x11_screen_get_xscreen [Gdk 2.10]
gdk_x11_screen_lookup_visual [Gdk 2.10]	gdk_x11_screen_supports_net_wm_hint [Gdk 2.10]
gdk_x11_ungrab_server [Gdk 2.10]	gdk_x11_visual_get_xvisual [Gdk 2.10]
gdk_x11_window_move_to_current_desktop [Gdk 2.10]	gdk_x11_window_set_user_time [Gdk 2.10]
gdk_x11_xatom_to_atom [Gdk 2.10]	gdk_x11_xatom_to_atom_for_display [Gdk 2.10]
gdk_xid_table_lookup [Gdk 2.10]	gdk_xid_table_lookup_for_display [Gdk 2.10]
gdkx_visual_get [Gdk 2.10]	

An LSB conforming implementation shall provide the generic data interfaces for GTK Drawing toolkit specified in Table 17-121, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-121 libgdk-x11-2.0 - GTK Drawing toolkit Data Interfaces

gdk_display [Gdk 2.10]	gdk_threads_lock [Gdk 2.10]
gdk_threads_unlock [Gdk 2.10]	

17.30 Data Definitions for libgdk-x11-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libgdk-x11-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.30.1 gtk-2.0/gdk/gdk.h

```
#define GDK_WINDOWING_X11
#define GDK_PIXMAP_OBJECT(object) \
    ((GdkPixmapObject *) GDK_PIXMAP (object))
#define GDK_WINDOW_OBJECT(object) \
    ((GdkWindowObject *) GDK_WINDOW (object))
#define GDK_TYPE_WINDOW_ATTRIBUTES_TYPE \
    (gdk_window_attributes_type_get_type())
#define GDK_COLORMAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_COLORMAP, \
    GdkColormapClass))
#define GDK_DEVICE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_DEVICE, \
    GdkDeviceClass))
#define GDK_DISPLAY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_DISPLAY, \
    GdkDisplayClass))
#define GDK_DISPLAY_MANAGER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_DISPLAY_MANAGER, \
    GdkDisplayManagerClass))
#define GDK_DRAG_CONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_DRAG_CONTEXT, \
    GdkDragContextClass))
#define GDK_DRAWABLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_DRAWABLE, \
    GdkDrawableClass))
#define GDK_GC_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_GC, GdkGCClass))
#define GDK_IMAGE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_IMAGE, \
    GdkImageClass))
#define GDK_KEYMAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_KEYMAP, \
    GdkKeymapClass))
```

```

#define GDK_PANGO_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_PANGO_RENDERER, \
    \
    GdkPangoRendererClass))
#define GDK_PIXMAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_PIXMAP, \
    GdkPixmapObjectClass))
#define GDK_SCREEN_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_SCREEN, \
    GdkScreenClass))
#define GDK_VISUAL_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_VISUAL, \
    GdkVisualClass))
#define GDK_WINDOW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GDK_TYPE_WINDOW, \
    GdkWindowObjectClass))
#define GDK_IS_COLORMAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_COLORMAP))
#define GDK_IS_DEVICE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_DEVICE))
#define GDK_IS_DISPLAY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_DISPLAY))
#define GDK_IS_DISPLAY_MANAGER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_DISPLAY_MANAGER))
#define GDK_IS_DRAG_CONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_DRAG_CONTEXT))
#define GDK_IS_DRAWABLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_DRAWABLE))
#define GDK_IS_GC_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_GC))
#define GDK_IS_IMAGE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_IMAGE))
#define GDK_IS_KEYMAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_KEYMAP))
#define GDK_IS_PANGO_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_PANGO_RENDERER))
#define GDK_IS_PIXMAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_PIXMAP))
#define GDK_IS_SCREEN_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_SCREEN))
#define GDK_IS_VISUAL_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_VISUAL))
#define GDK_IS_WINDOW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GDK_TYPE_WINDOW))
#define GDK_COLORMAP(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_COLORMAP, \
    GdkColormap))
#define GDK_DEVICE(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_DEVICE, \
    GdkDevice))
#define GDK_DISPLAY_OBJECT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_DISPLAY, \
    GdkDisplay))
#define GDK_DISPLAY_MANAGER(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_DISPLAY_MANAGER, \
    GdkDisplayManager))
#define GDK_DRAG_CONTEXT(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_DRAG_CONTEXT, \
    GdkDragContext))
#define GDK_DRAWABLE(object) \
    (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_DRAWABLE, \
    GdkDrawable))
#define GDK_GC(object) \

```

```

        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_GC, GdkGC))
#define GDK_IMAGE(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_IMAGE,
GdkImage))
#define GDK_KEYMAP(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_KEYMAP,
GdkKeymap))
#define GDK_PANGO_RENDERER(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object),
GDK_TYPE_PANGO_RENDERER, \
        GdkPangoRenderer))
#define GDK_PIXMAP(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_PIXMAP,
GdkPixmap))
#define GDK_SCREEN(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_SCREEN,
GdkScreen))
#define GDK_VISUAL(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_VISUAL,
GdkVisual))
#define GDK_WINDOW(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GDK_TYPE_WINDOW,
GdkWindow))
#define GDK_IS_COLORMAP(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_COLORMAP))
#define GDK_IS_DEVICE(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_DEVICE))
#define GDK_IS_DISPLAY(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_DISPLAY))
#define GDK_IS_DISPLAY_MANAGER(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object),
GDK_TYPE_DISPLAY_MANAGER))
#define GDK_IS_DRAG_CONTEXT(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object),
GDK_TYPE_DRAG_CONTEXT))
#define GDK_IS_DRAWABLE(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_DRAWABLE))
#define GDK_IS_GC(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_GC))
#define GDK_IS_IMAGE(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_IMAGE))
#define GDK_IS_KEYMAP(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_KEYMAP))
#define GDK_IS_PANGO_RENDERER(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object),
GDK_TYPE_PANGO_RENDERER))
#define GDK_IS_PIXMAP(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_PIXMAP))
#define GDK_IS_SCREEN(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_SCREEN))
#define GDK_IS_VISUAL(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_VISUAL))
#define GDK_IS_WINDOW(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object), GDK_TYPE_WINDOW))
#define GDK_COLORMAP_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_COLORMAP, \
        GdkColormapClass))
#define GDK_DEVICE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_DEVICE,
GdkDeviceClass))
#define GDK_DISPLAY_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_DISPLAY, \
        GdkDisplayClass))
#define GDK_DISPLAY_MANAGER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_DISPLAY_MANAGER,
\

```

```

        GdkDisplayManagerClass))
#define GDK_DRAG_CONTEXT_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_DRAG_CONTEXT, \
        GdkDragContextClass))
#define GDK_DRAWABLE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_DRAWABLE, \
        GdkDrawableClass))
#define GDK_GC_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_GC, GdkGCClass))
#define GDK_IMAGE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_IMAGE, \
        GdkImageClass))
#define GDK_KEYMAP_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_KEYMAP, \
        GdkKeymapClass))
#define GDK_PANGO_RENDERER_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_PANGO_RENDERER, \
        \
        GdkPangoRendererClass))
#define GDK_PIXMAP_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_PIXMAP, \
        GdkPixmapObjectClass))
#define GDK_SCREEN_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_SCREEN, \
        GdkScreenClass))
#define GDK_VISUAL_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_VISUAL, \
        GdkVisualClass))
#define GDK_WINDOW_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GDK_TYPE_WINDOW, \
        GdkWindowObjectClass))
#define GDK_THREADS_ENTER() \
    G_STMT_START { if (gdk_threads_lock) (*gdk_threads_lock) \
    (); } \
    G_STMT_END
#define GDK_THREADS_LEAVE() \
    G_STMT_START { if (gdk_threads_unlock) (*gdk_threads_unlock) \
    (); } \
    G_STMT_END
#define GDK_POINTER_TO_ATOM(ptr) ((GdkAtom) (ptr))
#define _GDK_MAKE_ATOM(val) ((GdkAtom) GUINT_TO_POINTER(val))
#define GDK_ATOM_TO_POINTER(atom) (atom)
#define GDK_TYPE_AXIS_USE (gdk_axis_use_get_type())
#define GDK_TYPE_BYTE_ORDER (gdk_byte_order_get_type())
#define GDK_TYPE_CAP_STYLE (gdk_cap_style_get_type())
#define GDK_TYPE_COLORMAP (gdk_colormap_get_type())
#define GDK_TYPE_COLOR (gdk_color_get_type())
#define GDK_TYPE_CROSSING_MODE (gdk_crossing_mode_get_type())
#define GDK_TYPE_CURSOR (gdk_cursor_get_type())
#define GDK_TYPE_CURSOR_TYPE (gdk_cursor_type_get_type())
#define GDK_TYPE_DEVICE (gdk_device_get_type())
#define GDK_TYPE_DISPLAY (gdk_display_get_type())
#define GDK_TYPE_DISPLAY_MANAGER \
    (gdk_display_manager_get_type())
#define GDK_TYPE_DRAG_ACTION (gdk_drag_action_get_type())
#define GDK_TYPE_DRAG_CONTEXT (gdk_drag_context_get_type())
#define GDK_TYPE_DRAG_PROTOCOL (gdk_drag_protocol_get_type())
#define GDK_TYPE_DRAWABLE (gdk_drawable_get_type())
#define GDK_TYPE_EVENT (gdk_event_get_type())
#define GDK_TYPE_EVENT_MASK (gdk_event_mask_get_type())
#define GDK_TYPE_EVENT_TYPE (gdk_event_type_get_type())
#define GDK_TYPE_EXTENSION_MODE (gdk_extension_mode_get_type())
#define GDK_TYPE_FILL (gdk_fill_get_type())
#define GDK_TYPE_FILL_RULE (gdk_fill_rule_get_type())
#define GDK_TYPE_FILTER_RETURN (gdk_filter_return_get_type())
#define GDK_TYPE_FONT_TYPE (gdk_font_type_get_type())

```

```

#define GDK_TYPE_FUNCTION      (gdk_function_get_type())
#define GDK_TYPE_GC            (gdk_gc_get_type ())
#define GDK_TYPE_GC_VALUES_MASK (gdk_gc_values_mask_get_type())
#define GDK_TYPE_GRAB_STATUS   (gdk_grab_status_get_type())
#define GDK_TYPE_GRAVITY       (gdk_gravity_get_type())
#define GDK_TYPE_IMAGE         (gdk_image_get_type ())
#define GDK_TYPE_IMAGE_TYPE     (gdk_image_type_get_type())
#define GDK_TYPE_INPUT_CONDITION
(gdk_input_condition_get_type())
#define GDK_TYPE_INPUT_MODE     (gdk_input_mode_get_type())
#define GDK_TYPE_INPUT_SOURCE   (gdk_input_source_get_type())
#define GDK_TYPE_JOIN_STYLE     (gdk_join_style_get_type())
#define GDK_TYPE_KEYMAP         (gdk_keymap_get_type ())
#define GDK_TYPE_LINE_STYLE     (gdk_line_style_get_type())
#define GDK_TYPE_MODIFIER_TYPE  (gdk_modifier_type_get_type())
#define GDK_TYPE_NOTIFY_TYPE    (gdk_notify_type_get_type())
#define GDK_TYPE_OVERLAP_TYPE   (gdk_overlap_type_get_type())
#define GDK_TYPE_OWNER_CHANGE   (gdk_owner_change_get_type())
#define GDK_TYPE_PANGO_RENDERER (gdk_pango_renderer_get_type())
#define GDK_TYPE_PIXMAP         (gdk_pixmap_get_type ())
#define GDK_TYPE_PROPERTY_STATE (gdk_property_state_get_type())
#define GDK_TYPE_PROP_MODE      (gdk_prop_mode_get_type())
#define GDK_TYPE_RECTANGLE      (gdk_rectangle_get_type ())
#define GDK_TYPE_RGB_DITHER     (gdk_rgb_dither_get_type())
#define GDK_TYPE_SCREEN         (gdk_screen_get_type ())
#define GDK_TYPE_SCROLL_DIRECTION
(gdk_scroll_direction_get_type())
#define GDK_TYPE_SETTING_ACTION (gdk_setting_action_get_type())
#define GDK_TYPE_STATUS         (gdk_status_get_type())
#define GDK_TYPE_SUBWINDOW_MODE (gdk_subwindow_mode_get_type())
#define GDK_TYPE_VISIBILITY_STATE
(gdk_visibility_state_get_type())
#define GDK_TYPE_VISUAL         (gdk_visual_get_type ())
#define GDK_TYPE_VISUAL_TYPE    (gdk_visual_type_get_type())
#define GDK_TYPE_WINDOW_CLASS   (gdk_window_class_get_type())
#define GDK_TYPE_WINDOW_EDGE    (gdk_window_edge_get_type())
#define GDK_TYPE_WINDOW_HINTS   (gdk_window_hints_get_type())
#define GDK_TYPE_WINDOW         (gdk_window_object_get_type ())
#define GDK_TYPE_WINDOW_STATE   (gdk_window_state_get_type())
#define GDK_TYPE_WINDOW_TYPE    (gdk_window_type_get_type())
#define GDK_TYPE_WINDOW_TYPE_HINT
(gdk_window_type_hint_get_type())
#define GDK_TYPE_WM_DECORATION  (gdk_wm_decoration_get_type())
#define GDK_TYPE_WM_FUNCTION    (gdk_wm_function_get_type())
#define GDK_PRIORITY_EVENTS      (G_PRIORITY_DEFAULT)
#define GDK_PRIORITY_REDRAW      (G_PRIORITY_HIGH_IDLE + 20)
#define GDK_CURRENT_TIME         0L
#define GDK_HAVE_WCHAR_H         1
#define GDK_HAVE_WCTYPE_H        1
#define GDK_MAX_TIMECOORD_AXES   128
#define GDK_PARENT_RELATIVE      1L
#define GDKVAR extern
#define gdk_draw_bitmap gdk_draw_drawable
#define GDK_NONE _GDK_MAKE_ATOM (0)
#define GDK_SELECTION_PRIMARY _GDK_MAKE_ATOM (1)
#define GDK_SELECTION_TYPE_DRAWABLE _GDK_MAKE_ATOM (17)
#define GDK_TARGET_DRAWABLE _GDK_MAKE_ATOM (17)
#define GDK_SELECTION_TYPE_INTEGER _GDK_MAKE_ATOM (19)
#define GDK_SELECTION_SECONDARY _GDK_MAKE_ATOM (2)
#define GDK_SELECTION_TYPE_PIXMAP _GDK_MAKE_ATOM (20)
#define GDK_TARGET_PIXMAP _GDK_MAKE_ATOM (20)
#define GDK_SELECTION_TYPE_STRING _GDK_MAKE_ATOM (31)
#define GDK_TARGET_STRING _GDK_MAKE_ATOM (31)
#define GDK_SELECTION_TYPE_WINDOW _GDK_MAKE_ATOM (33)
#define GDK_SELECTION_TYPE_ATOM _GDK_MAKE_ATOM (4)
#define GDK_SELECTION_TYPE_BITMAP _GDK_MAKE_ATOM (5)

```

```

#define GDK_TARGET_BITMAP      _GDK_MAKE_ATOM (5)
#define GDK_SELECTION_CLIPBOARD _GDK_MAKE_ATOM (69)
#define GDK_SELECTION_TYPE_COLORMAP _GDK_MAKE_ATOM (7)
#define GDK_TARGET_COLORMAP    _GDK_MAKE_ATOM (7)

typedef struct _GdkDrawable {
    GObject parent_instance;
} GdkWindow;
typedef struct _GdkColor {
    guint32 pixel;
    guint16 red;
    guint16 green;
    guint16 blue;
} GdkColor;
typedef enum {
    GDK_VISUAL_STATIC_GRAY = 0,
    GDK_VISUAL_GRAYSCALE = 1,
    GDK_VISUAL_STATIC_COLOR = 2,
    GDK_VISUAL_PSEUDO_COLOR = 3,
    GDK_VISUAL_TRUE_COLOR = 4,
    GDK_VISUAL_DIRECT_COLOR = 5
} GdkVisualType;
typedef enum {
    GDK_LSB_FIRST = 0,
    GDK_MSB_FIRST = 1
} GdkByteOrder;
typedef struct _GdkVisual {
    GObject parent_instance;
    GdkVisualType type;
    gint depth;
    GdkByteOrder byte_order;
    gint colormap_size;
    gint bits_per_rgb;
    guint32 red_mask;
    gint red_shift;
    gint red_prec;
    guint32 green_mask;
    gint green_shift;
    gint green_prec;
    guint32 blue_mask;
    gint blue_shift;
    gint blue_prec;
} GdkVisual;
typedef struct _GdkColormap {
    GObject parent_instance;
    gint size;
    GdkColor *colors;
    GdkVisual *visual;
    gpointer windowing_data;
} GdkColormap;
typedef struct _GdkGC {
    GObject parent_instance;
    gint clip_x_origin;
    gint clip_y_origin;
    gint ts_x_origin;
    gint ts_y_origin;
    GdkColormap *colormap;
} GdkGC;
typedef struct _GdkDrawable {
    GObject parent_instance;
} GdkDrawable;
typedef enum {
    GDK_SOURCE_MOUSE = 0,
    GDK_SOURCE_PEN = 1,
    GDK_SOURCE_ERASER = 2,
    GDK_SOURCE_CURSOR = 3

```

```

} GdkInputSource;
typedef enum {
    GDK_MODE_DISABLED = 0,
    GDK_MODE_SCREEN = 1,
    GDK_MODE_WINDOW = 2
} GdkInputMode;
typedef enum {
    GDK_AXIS_IGNORE = 0,
    GDK_AXIS_X = 1,
    GDK_AXIS_Y = 2,
    GDK_AXIS_PRESSURE = 3,
    GDK_AXIS_XTILT = 4,
    GDK_AXIS_YTILT = 5,
    GDK_AXIS_WHEEL = 6,
    GDK_AXIS_LAST = 7
} GdkAxisUse;
typedef struct _GdkDeviceAxis {
    GdkAxisUse use;
    gdouble min;
    gdouble max;
} GdkDeviceAxis;
typedef enum {
    GDK_SHIFT_MASK = 1,
    GDK_LOCK_MASK = 2,
    GDK_CONTROL_MASK = 4,
    GDK_MOD1_MASK = 8,
    GDK_MOD2_MASK = 16,
    GDK_MOD3_MASK = 32,
    GDK_MOD4_MASK = 64,
    GDK_MOD5_MASK = 128,
    GDK_BUTTON1_MASK = 256,
    GDK_BUTTON2_MASK = 512,
    GDK_BUTTON3_MASK = 1024,
    GDK_BUTTON4_MASK = 2048,
    GDK_BUTTON5_MASK = 4096,
    GDK_RELEASE_MASK = 1073741824,
    GDK_MODIFIER_MASK = 1073750015
} GdkModifierType;
typedef struct _GdkDeviceKey {
    guint keyval;
    GdkModifierType modifiers;
} GdkDeviceKey;
typedef struct _GdkDevice {
    GObject parent_instance;
    gchar *name;
    GdkInputSource source;
    GdkInputMode mode;
    gboolean has_cursor;
    gint num_axes;
    GdkDeviceAxis *axes;
    gint num_keys;
    GdkDeviceKey *keys;
} GdkDevice;
typedef struct _GdkDisplay {
    GObject parent_instance;
    GList *queued_events;
    GList *queued_tail;
    guint32 button_click_time[2];
    GdkWindow *button_window[2];
    gint button_number[2];
    guint double_click_time;
    GdkDevice *core_pointer;
    const GdkDisplayPointerHooks *pointer_hooks;
    guint closed:1;
    guint double_click_distance;
    gint button_x[2];

```



```

    gint button_y[2];
} GdkDisplay;
typedef struct _GdkScreen GdkScreen;
typedef struct _GdkDisplayPointerHooks {
    void (*get_pointer) (GdkDisplay *, GdkScreen * *, gint *, gint
*,
                        GdkModifierType *);
    GdkWindow *(*window_get_pointer) (GdkDisplay *, GdkWindow *,
gint *,
                                gint *, GdkModifierType *);
    GdkWindow *(*window_at_pointer) (GdkDisplay *, gint *, gint *);
} GdkDisplayPointerHooks;
typedef enum {
    GDK_EXPOSURE_MASK = 2,
    GDK_POINTER_MOTION_MASK = 4,
    GDK_POINTER_MOTION_HINT_MASK = 8,
    GDK_BUTTON_MOTION_MASK = 16,
    GDK_BUTTON1_MOTION_MASK = 32,
    GDK_BUTTON2_MOTION_MASK = 64,
    GDK_BUTTON3_MOTION_MASK = 128,
    GDK_BUTTON_PRESS_MASK = 256,
    GDK_BUTTON_RELEASE_MASK = 512,
    GDK_KEY_PRESS_MASK = 1024,
    GDK_KEY_RELEASE_MASK = 2048,
    GDK_ENTER_NOTIFY_MASK = 4096,
    GDK_LEAVE_NOTIFY_MASK = 8192,
    GDK_FOCUS_CHANGE_MASK = 16384,
    GDK_STRUCTURE_MASK = 32768,
    GDK_PROPERTY_CHANGE_MASK = 65536,
    GDK_VISIBILITY_NOTIFY_MASK = 131072,
    GDK_PROXIMITY_IN_MASK = 262144,
    GDK_PROXIMITY_OUT_MASK = 524288,
    GDK_SUBSTRUCTURE_MASK = 1048576,
    GDK_SCROLL_MASK = 2097152,
    GDK_ALL_EVENTS_MASK = 4194302
} GdkEventMask;
typedef enum {
    GDK_X_CURSOR = 0,
    GDK_ARROW = 2,
    GDK_BASED_ARROW_DOWN = 4,
    GDK_BASED_ARROW_UP = 6,
    GDK_BOAT = 8,
    GDK_BOGOSITY = 10,
    GDK_BOTTOM_LEFT_CORNER = 12,
    GDK_BOTTOM_RIGHT_CORNER = 14,
    GDK_BOTTOM_SIDE = 16,
    GDK_BOTTOM_TEE = 18,
    GDK_BOX_SPIRAL = 20,
    GDK_CENTER_PTR = 22,
    GDK_CIRCLE = 24,
    GDK_CLOCK = 26,
    GDK_COFFEE_MUG = 28,
    GDK_CROSS = 30,
    GDK_CROSS_REVERSE = 32,
    GDK_CROSSHAIR = 34,
    GDK_DIAMOND_CROSS = 36,
    GDK_DOT = 38,
    GDK_DOTBOX = 40,
    GDK_DOUBLE_ARROW = 42,
    GDK_DRAFT_LARGE = 44,
    GDK_DRAFT_SMALL = 46,
    GDK_DRAPED_BOX = 48,
    GDK_EXCHANGE = 50,
    GDK_FLEUR = 52,
    GDK_GOBBLER = 54,
    GDK_GUMBY = 56,

```



```

GDK_HAND1 = 58,
GDK_HAND2 = 60,
GDK_HEART = 62,
GDK_ICON = 64,
GDK_IRON_CROSS = 66,
GDK_LEFT_PTR = 68,
GDK_LEFT_SIDE = 70,
GDK_LEFT_TEE = 72,
GDK_LEFTBUTTON = 74,
GDK_LL_ANGLE = 76,
GDK_LR_ANGLE = 78,
GDK_MAN = 80,
GDK_MIDDLEBUTTON = 82,
GDK_MOUSE = 84,
GDK_PENCIL = 86,
GDK_PIRATE = 88,
GDK_PLUS = 90,
GDK_QUESTION_ARROW = 92,
GDK_RIGHT_PTR = 94,
GDK_RIGHT_SIDE = 96,
GDK_RIGHT_TEE = 98,
GDK_RIGHTBUTTON = 100,
GDK_RTL_LOGO = 102,
GDK_SAILBOAT = 104,
GDK_SB_DOWN_ARROW = 106,
GDK_SB_H_DOUBLE_ARROW = 108,
GDK_SB_LEFT_ARROW = 110,
GDK_SB_RIGHT_ARROW = 112,
GDK_SB_UP_ARROW = 114,
GDK_SB_V_DOUBLE_ARROW = 116,
GDK_SHUTTLE = 118,
GDK_SIZING = 120,
GDK_SPIDER = 122,
GDK_SPRAYCAN = 124,
GDK_STAR = 126,
GDK_TARGET = 128,
GDK_TCROSS = 130,
GDK_TOP_LEFT_ARROW = 132,
GDK_TOP_LEFT_CORNER = 134,
GDK_TOP_RIGHT_CORNER = 136,
GDK_TOP_SIDE = 138,
GDK_TOP_TEE = 140,
GDK_TREK = 142,
GDK_UL_ANGLE = 144,
GDK_UMBRELLA = 146,
GDK_UR_ANGLE = 148,
GDK_WATCH = 150,
GDK_XTERM = 152,
GDK_LAST_CURSOR = 153,
GDK_CURSOR_IS_PIXMAP = -1
} GdkCursorType;
typedef struct _GdkCursor {
    GdkCursorType type;
    guint ref_count;
} GdkCursor;
typedef struct _GdkKeymap {
    GObject parent_instance;
    GdkDisplay *display;
} GdkKeymap;
typedef struct _GdkAtom *GdkAtom;
typedef enum {
    GDK_FILTER_CONTINUE = 0,
    GDK_FILTER_TRANSLATE = 1,
    GDK_FILTER_REMOVE = 2
} GdkFilterReturn;
typedef void GdkXEvent;

```

```

typedef enum {
    GDK_NOTHING = -1,
    GDK_DELETE = 0,
    GDK_DESTROY = 1,
    GDK_EXPOSE = 2,
    GDK_MOTION_NOTIFY = 3,
    GDK_BUTTON_PRESS = 4,
    GDK_2BUTTON_PRESS = 5,
    GDK_3BUTTON_PRESS = 6,
    GDK_BUTTON_RELEASE = 7,
    GDK_KEY_PRESS = 8,
    GDK_KEY_RELEASE = 9,
    GDK_ENTER_NOTIFY = 10,
    GDK_LEAVE_NOTIFY = 11,
    GDK_FOCUS_CHANGE = 12,
    GDK_CONFIGURE = 13,
    GDK_MAP = 14,
    GDK_UNMAP = 15,
    GDK_PROPERTY_NOTIFY = 16,
    GDK_SELECTION_CLEAR = 17,
    GDK_SELECTION_REQUEST = 18,
    GDK_SELECTION_NOTIFY = 19,
    GDK_PROXIMITY_IN = 20,
    GDK_PROXIMITY_OUT = 21,
    GDK_DRAG_ENTER = 22,
    GDK_DRAG_LEAVE = 23,
    GDK_DRAG_MOTION = 24,
    GDK_DRAG_STATUS = 25,
    GDK_DROP_START = 26,
    GDK_DROP_FINISHED = 27,
    GDK_CLIENT_EVENT = 28,
    GDK_VISIBILITY_NOTIFY = 29,
    GDK_NO_EXPOSE = 30,
    GDK_SCROLL = 31,
    GDK_WINDOW_STATE = 32,
    GDK_SETTING = 33,
    GDK_OWNER_CHANGE = 34
} GdkEventType;

typedef struct _GdkEventAny {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
} GdkEventAny;

typedef struct _GdkRectangle {
    gint x;
    gint y;
    gint width;
    gint height;
} GdkRectangle;

typedef struct _GdkRegion GdkRegion;
typedef struct _GdkEventExpose {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkRectangle area;
    GdkRegion *region;
    gint count;
} GdkEventExpose;

typedef struct _GdkEventNoExpose {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
} GdkEventNoExpose;

typedef enum {
    GDK_VISIBILITY_UNOBSCURED = 0,
    GDK_VISIBILITY_PARTIAL = 1,

```

```

        GDK_VISIBILITY_FULLY_OBSCURED = 2
    } GdkVisibilityState;
typedef struct _GdkEventVisibility {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkVisibilityState state;
} GdkEventVisibility;
typedef struct _GdkEventMotion {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    guint32 time;
    gdouble x;
    gdouble y;
    gdouble *axes;
    guint state;
    gint16 is_hint;
    GdkDevice *device;
    gdouble x_root;
    gdouble y_root;
} GdkEventMotion;
typedef struct _GdkEventButton {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    guint32 time;
    gdouble x;
    gdouble y;
    gdouble *axes;
    guint state;
    guint button;
    GdkDevice *device;
    gdouble x_root;
    gdouble y_root;
} GdkEventButton;
typedef enum {
    GDK_SCROLL_UP = 0,
    GDK_SCROLL_DOWN = 1,
    GDK_SCROLL_LEFT = 2,
    GDK_SCROLL_RIGHT = 3
} GdkScrollDirection;
typedef struct _GdkEventScroll {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    guint32 time;
    gdouble x;
    gdouble y;
    guint state;
    GdkScrollDirection direction;
    GdkDevice *device;
    gdouble x_root;
    gdouble y_root;
} GdkEventScroll;
typedef struct _GdkEventKey {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    guint32 time;
    guint state;
    guint keyval;
    gint length;
    gchar *string;
    guint16 hardware_keycode;
    guint8 group;

```

```

    } GdkEventKey;
typedef enum {
    GDK_CROSSING_NORMAL = 0,
    GDK_CROSSING_GRAB = 1,
    GDK_CROSSING_UNGRAB = 2
} GdkCrossingMode;
typedef enum {
    GDK_NOTIFY_ANCESTOR = 0,
    GDK_NOTIFY_VIRTUAL = 1,
    GDK_NOTIFY_INFERIOR = 2,
    GDK_NOTIFY_NONLINEAR = 3,
    GDK_NOTIFY_NONLINEAR_VIRTUAL = 4,
    GDK_NOTIFY_UNKNOWN = 5
} GdkNotifyType;
typedef struct _GdkEventCrossing {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkWindow *subwindow;
    guint32 time;
    gdouble x;
    gdouble y;
    gdouble x_root;
    gdouble y_root;
    GdkCrossingMode mode;
    GdkNotifyType detail;
    gboolean focus;
    guint state;
} GdkEventCrossing;
typedef struct _GdkEventFocus {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    gint16 in;
} GdkEventFocus;
typedef struct _GdkEventConfigure {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    gint x;
    gint y;
    gint width;
    gint height;
} GdkEventConfigure;
typedef struct _GdkEventProperty {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkAtom atom;
    guint32 time;
    guint state;
} GdkEventProperty;
typedef guint32 GdkNativeWindow;
typedef struct _GdkEventSelection {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkAtom selection;
    GdkAtom target;
    GdkAtom property;
    guint32 time;
    GdkNativeWindow requestor;
} GdkEventSelection;
typedef enum {
    GDK_OWNER_CHANGE_NEW_OWNER = 0,
    GDK_OWNER_CHANGE_DESTROY = 1,

```

```

        GDK_OWNER_CHANGE_CLOSE = 2
    } GdkOwnerChange;
typedef struct _GdkEventOwnerChange {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkNativeWindow owner;
    GdkOwnerChange reason;
    GdkAtom selection;
    guint32 time;
    guint32 selection_time;
} GdkEventOwnerChange;
typedef struct _GdkEventProximity {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    guint32 time;
    GdkDevice *device;
} GdkEventProximity;
typedef struct _GdkEventClient {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkAtom message_type;
    gushort data_format;
    union {
        char b[20];
        short int s[10];
        long int l[5];
    } data;
} GdkEventClient;
typedef enum {
    GDK_DRAG_PROTO_MOTIF = 0,
    GDK_DRAG_PROTO_XDND = 1,
    GDK_DRAG_PROTO_ROOTWIN = 2,
    GDK_DRAG_PROTO_NONE = 3,
    GDK_DRAG_PROTO_WIN32_DROPFILES = 4,
    GDK_DRAG_PROTO_OLE2 = 5,
    GDK_DRAG_PROTO_LOCAL = 6
} GdkDragProtocol;
typedef enum {
    GDK_ACTION_DEFAULT = 1,
    GDK_ACTION_COPY = 2,
    GDK_ACTION_MOVE = 4,
    GDK_ACTION_LINK = 8,
    GDK_ACTION_PRIVATE = 16,
    GDK_ACTION_ASK = 32
} GdkDragAction;
typedef struct _GdkDragContext {
    GObject parent_instance;
    GdkDragProtocol protocol;
    gboolean is_source;
    GdkWindow *source_window;
    GdkWindow *dest_window;
    GList *targets;
    GdkDragAction actions;
    GdkDragAction suggested_action;
    GdkDragAction action;
    guint32 start_time;
    gpointer windowing_data;
} GdkDragContext;
typedef struct _GdkEventDND {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkDragContext *context;

```

```

    guint32 time;
    gshort x_root;
    gshort y_root;
} GdkEventDND;
typedef enum {
    GDK_WINDOW_STATE_WITHDRAWN = 1,
    GDK_WINDOW_STATE_ICONIFIED = 2,
    GDK_WINDOW_STATE_MAXIMIZED = 4,
    GDK_WINDOW_STATE_STICKY = 8,
    GDK_WINDOW_STATE_FULLSCREEN = 16,
    GDK_WINDOW_STATE_ABOVE = 32,
    GDK_WINDOW_STATE_BELOW = 64
} GdkWindowState;
typedef struct _GdkEventWindowState {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkWindowState changed_mask;
    GdkWindowState new_window_state;
} GdkEventWindowState;
typedef enum {
    GDK_SETTING_ACTION_NEW = 0,
    GDK_SETTING_ACTION_CHANGED = 1,
    GDK_SETTING_ACTION_DELETED = 2
} GdkSettingAction;
typedef struct _GdkEventSetting {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    GdkSettingAction action;
    char *name;
} GdkEventSetting;
typedef union _GdkEvent {
    GdkEventType type;
    GdkEventAny any;
    GdkEventExpose expose;
    GdkEventNoExpose no_expose;
    GdkEventVisibility visibility;
    GdkEventMotion motion;
    GdkEventButton button;
    GdkEventScroll scroll;
    GdkEventKey key;
    GdkEventCrossing crossing;
    GdkEventFocus focus_change;
    GdkEventConfigure configure;
    GdkEventProperty property;
    GdkEventSelection selection;
    GdkEventOwnerChange owner_change;
    GdkEventProximity proximity;
    GdkEventClient client;
    GdkEventDND dnd;
    GdkEventWindowState window_state;
    GdkEventSetting setting;
} GdkEvent;
typedef GdkFilterReturn(*GdkFilterFunc) (GdkXEvent * xevent,
                                         GdkEvent * event, gpointer data);
typedef struct _GdkDrawable {
    GObject parent_instance;
} GdkPixmap;
typedef struct _GdkDrawable {
    GObject parent_instance;
} GdkBitmap;
typedef struct _GdkDisplayManager GdkDisplayManager;
typedef enum {
    GDK_CLIP_BY_CHILDREN = 0,
    GDK_INCLUDE_INFERIORS = 1

```

```

} GdkSubwindowMode;
typedef enum {
    GDK_WINDOW_EDGE_NORTH_WEST = 0,
    GDK_WINDOW_EDGE_NORTH = 1,
    GDK_WINDOW_EDGE_NORTH_EAST = 2,
    GDK_WINDOW_EDGE_WEST = 3,
    GDK_WINDOW_EDGE_EAST = 4,
    GDK_WINDOW_EDGE_SOUTH_WEST = 5,
    GDK_WINDOW_EDGE_SOUTH = 6,
    GDK_WINDOW_EDGE_SOUTH_EAST = 7
} GdkWindowEdge;
typedef enum {
    GDK_IMAGE_NORMAL = 0,
    GDK_IMAGE_SHARED = 1,
    GDK_IMAGE_FASTEST = 2
} GdkImageType;
typedef struct _GdkImage {
    GObject parent_instance;
    GdkImageType type;
    GdkVisual *visual;
    GdkByteOrder byte_order;
    gint width;
    gint height;
    guint16 depth;
    guint16 bpp;
    guint16 bpl;
    guint16 bits_per_pixel;
    gpointer mem;
    GdkColormap *colormap;
    gpointer windowing_data;
} GdkImage;
typedef struct _GdkPangoRendererPrivate GdkPangoRendererPrivate;
typedef struct _GdkPangoRenderer {
    PangoRenderer parent_instance;
    GdkPangoRendererPrivate *priv;
} GdkPangoRenderer;
typedef enum {
    GDK_GRAB_SUCCESS = 0,
    GDK_GRAB_ALREADY_GRABBED = 1,
    GDK_GRAB_INVALID_TIME = 2,
    GDK_GRAB_NOT_VIEWABLE = 3,
    GDK_GRAB_FROZEN = 4
} GdkGrabStatus;
typedef enum {
    GDK_SOLID = 0,
    GDK_TILED = 1,
    GDK_STIPPLED = 2,
    GDK_OPAQUE_STIPPLED = 3
} GdkFill;
typedef enum {
    GDK_WINDOW_TYPE_HINT_NORMAL = 0,
    GDK_WINDOW_TYPE_HINT_DIALOG = 1,
    GDK_WINDOW_TYPE_HINT_MENU = 2,
    GDK_WINDOW_TYPE_HINT_TOOLBAR = 3,
    GDK_WINDOW_TYPE_HINT_SPLASHSCREEN = 4,
    GDK_WINDOW_TYPE_HINT_UTILITY = 5,
    GDK_WINDOW_TYPE_HINT_DOCK = 6,
    GDK_WINDOW_TYPE_HINT_DESKTOP = 7,
    GDK_WINDOW_TYPE_HINT_DROPDOWN_MENU,
    GDK_WINDOW_TYPE_HINT_POPUP_MENU,
    GDK_WINDOW_TYPE_HINT_TOOLTIP,
    GDK_WINDOW_TYPE_HINT_NOTIFICATION,
    GDK_WINDOW_TYPE_HINT_COMBO,
    GDK_WINDOW_TYPE_HINT_DND
} GdkWindowTypeHint;
typedef struct _GdkTimeCoord {

```

```

    guint32 time;
    gdouble axes[128];
} GdkTimeCoord;
typedef struct _GdkPoint {
    gint x;
    gint y;
} GdkPoint;
typedef struct _GdkFont GdkFont;
typedef enum {
    GDK_DECOR_ALL = 1,
    GDK_DECOR_BORDER = 2,
    GDK_DECOR_RESIZEH = 4,
    GDK_DECOR_TITLE = 8,
    GDK_DECOR_MENU = 16,
    GDK_DECOR_MINIMIZE = 32,
    GDK_DECOR_MAXIMIZE = 64
} GdkWMDecoration;
typedef enum {
    GDK_FUNC_ALL = 1,
    GDK_FUNC_RESIZE = 2,
    GDK_FUNC_MOVE = 4,
    GDK_FUNC_MINIMIZE = 8,
    GDK_FUNC_MAXIMIZE = 16,
    GDK_FUNC_CLOSE = 32
} GdkWMFunction;
typedef struct _GdkKeymapKey {
    guint keycode;
    gint group;
    gint level;
} GdkKeymapKey;
typedef enum {
    GDK_RGB_DITHER_NONE = 0,
    GDK_RGB_DITHER_NORMAL = 1,
    GDK_RGB_DITHER_MAX = 2
} GdkRgbDither;
typedef struct _GdkRgbCmap {
    guint32 colors[256];
    gint n_colors;
    GSList *info_list;
} GdkRgbCmap;
typedef enum {
    GDK_COPY = 0,
    GDK_INVERT = 1,
    GDK_XOR = 2,
    GDK_CLEAR = 3,
    GDK_AND = 4,
    GDK_AND_REVERSE = 5,
    GDK_AND_INVERT = 6,
    GDK_NOOP = 7,
    GDK_OR = 8,
    GDK_EQUIV = 9,
    GDK_OR_REVERSE = 10,
    GDK_COPY_INVERT = 11,
    GDK_OR_INVERT = 12,
    GDK_NAND = 13,
    GDK_NOR = 14,
    GDK_SET = 15
} GdkFunction;
typedef enum {
    GDK_LINE_SOLID = 0,
    GDK_LINE_ON_OFF_DASH = 1,
    GDK_LINE_DOUBLE_DASH = 2
} GdkLineStyle;
typedef enum {
    GDK_CAP_NOT_LAST = 0,
    GDK_CAP_BUTT = 1,

```



```

        GDK_CAP_ROUND = 2,
        GDK_CAP_PROJECTING = 3
    } GdkCapStyle;
typedef enum {
    GDK_JOIN_MITER = 0,
    GDK_JOIN_ROUND = 1,
    GDK_JOIN_BEVEL = 2
} GdkJoinStyle;
typedef struct _GdkGCValues {
    GdkColor foreground;
    GdkColor background;
    GdkFont *font;
    GdkFunction function;
    GdkFill fill;
    GdkPixmap *tile;
    GdkPixmap *stipple;
    GdkPixmap *clip_mask;
    GdkSubwindowMode subwindow_mode;
    gint ts_x_origin;
    gint ts_y_origin;
    gint clip_x_origin;
    gint clip_y_origin;
    gint graphics_exposures;
    gint line_width;
    GdkLineStyle line_style;
    GdkCapStyle cap_style;
    GdkJoinStyle join_style;
} GdkGCValues;
typedef enum {
    GDK_GC_FOREGROUND = 1,
    GDK_GC_BACKGROUND = 2,
    GDK_GC_FONT = 4,
    GDK_GC_FUNCTION = 8,
    GDK_GC_FILL = 16,
    GDK_GC_TILE = 32,
    GDK_GC_STIPPLE = 64,
    GDK_GC_CLIP_MASK = 128,
    GDK_GC_SUBWINDOW = 256,
    GDK_GC_TS_X_ORIGIN = 512,
    GDK_GC_TS_Y_ORIGIN = 1024,
    GDK_GC_CLIP_X_ORIGIN = 2048,
    GDK_GC_CLIP_Y_ORIGIN = 4096,
    GDK_GC_EXPOSURES = 8192,
    GDK_GC_LINE_WIDTH = 16384,
    GDK_GC_LINE_STYLE = 32768,
    GDK_GC_CAP_STYLE = 65536,
    GDK_GC_JOIN_STYLE = 131072
} GdkGCValuesMask;
typedef enum {
    GDK_WINDOW_ROOT = 0,
    GDK_WINDOW_TOPLEVEL = 1,
    GDK_WINDOW_CHILD = 2,
    GDK_WINDOW_DIALOG = 3,
    GDK_WINDOW_TEMP = 4,
    GDK_WINDOW_FOREIGN = 5
} GdkWindowType;
typedef struct _GdkSpan {
    gint x;
    gint y;
    gint width;
} GdkSpan;
typedef void (*GdkSpanFunc) (GdkSpan *, gpointer);
typedef enum {
    GDK_GRAVITY_NORTH_WEST = 1,
    GDK_GRAVITY_NORTH = 2,
    GDK_GRAVITY_NORTH_EAST = 3,

```

```

        GDK_GRAVITY_WEST = 4,
        GDK_GRAVITY_CENTER = 5,
        GDK_GRAVITY_EAST = 6,
        GDK_GRAVITY_SOUTH_WEST = 7,
        GDK_GRAVITY_SOUTH = 8,
        GDK_GRAVITY_SOUTH_EAST = 9,
        GDK_GRAVITY_STATIC = 10
    } GdkGravity;
typedef struct _GdkGeometry {
    gint min_width;
    gint min_height;
    gint max_width;
    gint max_height;
    gint base_width;
    gint base_height;
    gint width_inc;
    gint height_inc;
    gdouble min_aspect;
    gdouble max_aspect;
    GdkGravity win_gravity;
} GdkGeometry;
typedef enum {
    GDK_HINT_POS = 1,
    GDK_HINT_MIN_SIZE = 2,
    GDK_HINT_MAX_SIZE = 4,
    GDK_HINT_BASE_SIZE = 8,
    GDK_HINT_ASPECT = 16,
    GDK_HINT_RESIZE_INC = 32,
    GDK_HINT_WIN_GRAVITY = 64,
    GDK_HINT_USER_POS = 128,
    GDK_HINT_USER_SIZE = 256
} GdkWindowHints;
typedef void (*GdkEventFunc) (GdkEvent *, gpointer);
typedef enum {
    GDK_OVERLAP_RECTANGLE_IN = 0,
    GDK_OVERLAP_RECTANGLE_OUT = 1,
    GDK_OVERLAP_RECTANGLE_PART = 2
} GdkOverlapType;
typedef struct _GdkSegment {
    gint x1;
    gint y1;
    gint x2;
    gint y2;
} GdkSegment;
typedef enum {
    GDK_PROP_MODE_REPLACE = 0,
    GDK_PROP_MODE_PREPEND = 1,
    GDK_PROP_MODE_APPEND = 2
} GdkPropMode;
typedef enum {
    GDK_INPUT_OUTPUT = 0,
    GDK_INPUT_ONLY = 1
} GdkWindowClass;
typedef struct _GdkWindowAttr {
    gchar *title;
    gint event_mask;
    gint x;
    gint y;
    gint width;
    gint height;
    GdkWindowClass wclass;
    GdkVisual *visual;
    GdkColormap *colormap;
    GdkWindowType window_type;
    GdkCursor *cursor;
    gchar *wmclass_name;

```

```

    gchar *wmclass_class;
    gboolean override_redirect;
} GdkWindowAttr;
typedef struct _GdkTrapezoid {
    double y1;
    double x11;
    double x21;
    double y2;
    double x12;
    double x22;
} GdkTrapezoid;
typedef enum {
    GDK_EVEN_ODD_RULE = 0,
    GDK_WINDING_RULE = 1
} GdkFillRule;
typedef struct _GdkPointerHooks {
    GdkWindow *(*get_pointer) (GdkWindow *, gint *, gint *,
                               GdkModifierType *);
    GdkWindow *(*window_at_pointer) (GdkScreen *, gint *, gint *);
} GdkPointerHooks;
typedef enum {
    GDK_EXTENSION_EVENTS_NONE = 0,
    GDK_EXTENSION_EVENTS_ALL = 1,
    GDK_EXTENSION_EVENTS_CURSOR = 2
} GdkExtensionMode;
typedef struct _GdkWindowObject {
    GdkDrawable parent_instance;
    GdkDrawable *impl;
    GdkWindowObject *parent;
    gpointer user_data;
    gint x;
    gint y;
    gint extension_events;
    GList *filters;
    GList *children;
    GdkColor bg_color;
    GdkPixmap *bg_pixmap;
    GSList *paint_stack;
    GdkRegion *update_area;
    guint update_freeze_count;
    guint8 window_type;
    guint8 depth;
    guint8 resize_count;
    GdkWindowState state;
    guint guffaw_gravity:1;
    guint input_only:1;
    guint modal_hint:1;
    guint destroyed:2;
    guint accept_focus:1;
    guint focus_on_map:1;
    GdkEventMask event_mask;
} GdkWindowObject;
typedef struct _GdkScreenClass {
    GObjectClass parent_class;
    void (*size_changed) (GdkScreen *);
} GdkScreenClass;
typedef guint32 GdkWChar;
typedef struct _GdkPixmapObject {
    GdkDrawable parent_instance;
    GdkDrawable *impl;
    gint depth;
} GdkPixmapObject;
typedef enum {
    GDK_INPUT_READ = 1,
    GDK_INPUT_WRITE = 2,
    GDK_INPUT_EXCEPTION = 4

```

```

} GdkInputCondition;
typedef void (*GdkInputFunction) (gpointer, gint,
GdkInputCondition);
typedef struct _GdkImageClass {
    GObjectClass parent_class;
} GdkImageClass;
typedef void (*GdkDestroyNotify) (gpointer);
typedef struct _GdkKeymapClass {
    GObjectClass parent_class;
    void (*direction_changed) (GdkKeymap *);
    void (*keys_changed) (GdkKeymap *);
} GdkKeymapClass;
typedef struct _GdkDrawableClass {
    GObjectClass parent_class;
    GdkGC *(*create_gc) (GdkDrawable *, GdkGCValues *,
GdkGCValuesMask);
    void (*draw_rectangle) (GdkDrawable *, GdkGC *, gboolean, gint,
gint,
gint, gint);
    void (*draw_arc) (GdkDrawable *, GdkGC *, gboolean, gint, gint,
gint,
gint, gint, gint);
    void (*draw_polygon) (GdkDrawable *, GdkGC *, gboolean, GdkPoint
*,
gint);
    void (*draw_text) (GdkDrawable *, GdkFont *, GdkGC *, gint, gint,
const gchar *, gint);
    void (*draw_text_wc) (GdkDrawable *, GdkFont *, GdkGC *, gint,
gint,
const GdkWChar *, gint);
    void (*draw_drawable) (GdkDrawable *, GdkGC *, GdkDrawable *,
gint,
gint, gint, gint, gint);
    void (*draw_points) (GdkDrawable *, GdkGC *, GdkPoint *, gint);
    void (*draw_segments) (GdkDrawable *, GdkGC *, GdkSegment *,
gint);
    void (*draw_lines) (GdkDrawable *, GdkGC *, GdkPoint *, gint);
    void (*draw_glyphs) (GdkDrawable *, GdkGC *, PangoFont *, gint,
gint,
PangoGlyphString *);
    void (*draw_image) (GdkDrawable *, GdkGC *, GdkImage *, gint,
gint,
gint, gint, gint, gint);
    gint(*get_depth) (GdkDrawable *);
    void (*get_size) (GdkDrawable *, gint *, gint *);
    void (*set_colormap) (GdkDrawable *, GdkColormap *);
    GdkColormap *(*get_colormap) (GdkDrawable *);
    GdkVisual *(*get_visual) (GdkDrawable *);
    GdkScreen *(*get_screen) (GdkDrawable *);
    GdkImage *(*get_image) (GdkDrawable *, gint, gint, gint, gint);
    GdkRegion *(*get_clip_region) (GdkDrawable *);
    GdkRegion *(*get_visible_region) (GdkDrawable *);
    GdkDrawable *(*get_composite_drawable) (GdkDrawable *, gint,
gint,
gint, gint, gint *, gint *);
    void (*draw_pixbuf) (GdkDrawable *, GdkGC *, GdkPixbuf *, gint,
gint,
gint, gint, gint, gint, GdkRgbDither, gint,
gint);
    GdkImage *(*_copy_to_image) (GdkDrawable *, GdkImage *, gint,
gint,
gint, gint, gint, gint);
    void (*draw_glyphs_transformed) (GdkDrawable *, GdkGC *,
PangoMatrix *,
PangoFont *, gint, gint,
PangoGlyphString *);

```

```

    void (*draw_trapezoids) (GdkDrawable *, GdkGC *, GdkTrapezoid
*, gint);
    cairo_surface_t *(*ref_cairo_surface) (void);
    GdkDrawable *(*get_source_drawable) (void);
    void (*set_cairo_clip) (void);
    cairo_surface_t *(*create_cairo_surface) (void);
    void (*draw_drawable_with_src) (void);
    void (*_gdk_reserved9) (void);
    void (*_gdk_reserved10) (void);
    void (*_gdk_reserved11) (void);
    void (*_gdk_reserved12) (void);
    void (*_gdk_reserved13) (void);
    void (*_gdk_reserved14) (void);
    void (*_gdk_reserved15) (void);
    void (*_gdk_reserved16) (void);
} GdkDrawableClass;
typedef struct _GdkPangoAttrEmbossed {
    PangoAttribute attr;
    gboolean embossed;
} GdkPangoAttrEmbossed;
typedef struct _GdkDisplayManagerClass {
    GObjectClass parent_class;
    void (*display_opened) (GdkDisplayManager *, GdkDisplay *);
} GdkDisplayManagerClass;
typedef struct _GdkPixmapObjectClass {
    GdkDrawableClass parent_class;
} GdkPixmapObjectClass;
typedef struct _GdkPangoRendererClass {
    PangoRendererClass parent_class;
} GdkPangoRendererClass;
typedef struct _GdkDisplayClass {
    GObjectClass parent_class;
    const gchar *(*get_display_name) (GdkDisplay *);
    gint(*get_n_screens) (GdkDisplay *);
    GdkScreen *(*get_screen) (GdkDisplay *, gint);
    GdkScreen *(*get_default_screen) (GdkDisplay *);
    void (*closed) (GdkDisplay *, gboolean);
} GdkDisplayClass;
typedef struct _GdkPangoAttrStipple {
    PangoAttribute attr;
    GdkBitmap *stipple;
} GdkPangoAttrStipple;
typedef struct _GdkColormapClass {
    GObjectClass parent_class;
} GdkColormapClass;
typedef struct _GdkDragContextClass {
    GObjectClass parent_class;
} GdkDragContextClass;
typedef struct _GdkWindowObjectClass {
    GdkDrawableClass parent_class;
} GdkWindowObjectClass;
typedef struct _GdkGCClass {
    GObjectClass parent_class;
    void (*get_values) (GdkGC *, GdkGCValues *);
    void (*set_values) (GdkGC *, GdkGCValues *, GdkGCValuesMask);
    void (*set_dashes) (GdkGC *, gint, gint8 *, gint);
    void (*_gdk_reserved1) (void);
    void (*_gdk_reserved2) (void);
    void (*_gdk_reserved3) (void);
    void (*_gdk_reserved4) (void);
} GdkGCClass;
typedef struct _GdkDeviceClass GdkDeviceClass;
typedef struct _GdkVisualClass GdkVisualClass;
typedef enum {
    GDK_OK = 0,
    GDK_ERROR = -1,

```

```

        GDK_ERROR_PARAM = -2,
        GDK_ERROR_FILE = -3,
        GDK_ERROR_MEM = -4
    } GdkStatus;
typedef enum {
    GDK_PROPERTY_NEW_VALUE,
    GDK_PROPERTY_DELETE
} GdkPropertyState;
typedef enum {
    GDK_WA_TITLE = 1 << 1,
    GDK_WA_X = 1 << 2,
    GDK_WA_Y = 1 << 3,
    GDK_WA_CURSOR = 1 << 4,
    GDK_WA_COLORMAP = 1 << 5,
    GDK_WA_VISUAL = 1 << 6,
    GDK_WA_WMCLASS = 1 << 7,
    GDK_WA_NOREDIR = 1 << 8
} GdkWindowAttributesType;
typedef struct _GdkEventGrabBroken {
    GdkEventType type;
    GdkWindow *window;
    gint8 send_event;
    gboolean keyboard;
    gboolean implicit;
    GdkWindow *grab_window;
} GdkEventGrabBroken;
extern void gdk_add_client_message_filter(GdkAtom message_type,
                                          GdkFilterFunc func,
                                          gpointer data);
extern GdkAtom gdk_atom_intern(const gchar * atom_name,
                               gboolean only_if_exists);
extern GdkAtom gdk_atom_intern_static_string(const gchar *
atom_name);
extern gchar *gdk_atom_name(GdkAtom atom);
extern GType gdk_axis_use_get_type(void);
extern void gdk_beep(void);
extern struct _GdkDrawable
*gdk_bitmap_create_from_data(GdkDrawable *
                               drawable,
                               const gchar * data,
                               gint width,
                               gint height);

extern GType gdk_byte_order_get_type(void);
extern GType gdk_cap_style_get_type(void);
extern GdkColor *gdk_color_copy(const GdkColor * color);
extern gboolean gdk_color_equal(const GdkColor * colora,
                                const GdkColor * colorb);
extern void gdk_color_free(GdkColor * color);
extern GType gdk_color_get_type(void);
extern guint gdk_color_hash(const GdkColor * colora);
extern gint gdk_color_parse(const gchar * spec, GdkColor * color);
extern gboolean gdk_colormap_alloc_color(GdkColormap * colormap,
                                          GdkColor * color,
                                          gboolean writeable,
                                          gboolean best_match);
extern gint gdk_colormap_alloc_colors(GdkColormap * colormap,
                                      GdkColor * colors, gint ncolors,
                                      gboolean writeable,
                                      gboolean best_match,
                                      gboolean * success);
extern void gdk_colormap_free_colors(GdkColormap * colormap,
                                      const GdkColor * colors,
                                      gint ncolors);
extern GdkScreen *gdk_colormap_get_screen(GdkColormap * cmap);
extern GdkColormap *gdk_colormap_get_system(void);
extern GType gdk_colormap_get_type(void);

```

```

extern GdkVisual *gdk_colormap_get_visual(GdkColormap * colormap);
extern GdkColormap *gdk_colormap_new(GdkVisual * visual,
                                     gboolean allocate);
extern void gdk_colormap_query_color(GdkColormap * colormap, gulong
pixel,
                                   GdkColor * result);
extern GType gdk_crossing_mode_get_type(void);
extern GdkDisplay *gdk_cursor_get_display(GdkCursor * cursor);
extern GdkPixbuf *gdk_cursor_get_image(GdkCursor * cursor);
extern GType gdk_cursor_get_type(void);
extern GdkCursor *gdk_cursor_new(GdkCursorType cursor_type);
extern GdkCursor *gdk_cursor_new_for_display(GdkDisplay * display,
                                             GdkCursorType cursor_type);
extern GdkCursor *gdk_cursor_new_from_name(GdkDisplay * display,
                                           const gchar * name);
extern GdkCursor *gdk_cursor_new_from_pixbuf(GdkDisplay * display,
                                             GdkPixbuf * pixbuf, gint x,
                                             gint y);
extern GdkCursor *gdk_cursor_new_from_pixmap(GdkPixmap * source,
                                             GdkPixmap * mask,
                                             const GdkColor * fg,
                                             const GdkColor * bg, gint x,
                                             gint y);
extern GdkCursor *gdk_cursor_ref(GdkCursor * cursor);
extern GType gdk_cursor_type_get_type(void);
extern void gdk_cursor_unref(GdkCursor * cursor);
extern void gdk_device_free_history(GdkTimeCoord * *events, gint
n_events);
extern gboolean gdk_device_get_axis(GdkDevice * device, gdouble *
axes,
                                   GdkAxisUse use, gdouble * value);
extern GdkDevice *gdk_device_get_core_pointer(void);
extern gboolean gdk_device_get_history(GdkDevice * device,
                                       GdkWindow * window, guint32 start,
                                       guint32 stop,
                                       GdkTimeCoord * **events,
                                       gint * n_events);
extern void gdk_device_get_state(GdkDevice * device, GdkWindow *
window,
                                gdouble * axes, GdkModifierType * mask);
extern GType gdk_device_get_type(void);
extern void gdk_device_set_axis_use(GdkDevice * device, guint
index_,
                                   GdkAxisUse use);
extern void gdk_device_set_key(GdkDevice * device, guint index_,
                              guint keyval, GdkModifierType modifiers);
extern gboolean gdk_device_set_mode(GdkDevice * device,
GdkInputMode mode);
extern void gdk_device_set_source(GdkDevice * device,
GdkInputSource source);
extern GList *gdk_devices_list(void);
extern void gdk_display_add_client_message_filter(GdkDisplay *
display,
                                                  GdkAtom message_type,
                                                  GdkFilterFunc func,
                                                  gpointer data);
extern void gdk_display_beep(GdkDisplay * display);
extern void gdk_display_close(GdkDisplay * display);
extern void gdk_display_flush(GdkDisplay * display);
extern GdkDevice *gdk_display_get_core_pointer(GdkDisplay *
display);
extern GdkDisplay *gdk_display_get_default(void);
extern guint gdk_display_get_default_cursor_size(GdkDisplay *
display);
extern GdkWindow *gdk_display_get_default_group(GdkDisplay *
display);

```

```

extern GdkScreen *gdk_display_get_default_screen(GdkDisplay *
display);
extern GdkEvent *gdk_display_get_event(GdkDisplay * display);
extern void gdk_display_get_maximal_cursor_size(GdkDisplay *
display,
                                guint * width,
                                guint * height);
extern gint gdk_display_get_n_screens(GdkDisplay * display);
extern const gchar *gdk_display_get_name(GdkDisplay * display);
extern void gdk_display_get_pointer(GdkDisplay * display,
                                GdkScreen * *screen, gint * x,
                                gint * y, GdkModifierType * mask);
extern GdkScreen *gdk_display_get_screen(GdkDisplay * display,
                                gint screen_num);
extern GType gdk_display_get_type(void);
extern GdkWindow *gdk_display_get_window_at_pointer(GdkDisplay *
display,
                                gint * win_x,
                                gint * win_y);
extern void gdk_display_keyboard_ungrab(GdkDisplay * display,
                                guint32 time_);
extern GList *gdk_display_list_devices(GdkDisplay * display);
extern GdkDisplayManager *gdk_display_manager_get(void);
extern GdkDisplay
    *gdk_display_manager_get_default_display(GdkDisplayManager *
display_manager);
extern GType gdk_display_manager_get_type(void);
extern GSList *gdk_display_manager_list_displays(GdkDisplayManager
*
display_manager);
extern
void
gdk_display_manager_set_default_display(GdkDisplayManager *
display_manager,
GdkDisplay * display);
extern GdkDisplay *gdk_display_open(const gchar * display_name);
extern GdkEvent *gdk_display_peek_event(GdkDisplay * display);
extern gboolean gdk_display_pointer_is_grabbed(GdkDisplay *
display);
extern void gdk_display_pointer_ungrab(GdkDisplay * display,
                                guint32 time_);
extern void gdk_display_put_event(GdkDisplay * display,
                                const GdkEvent * event);
extern
gboolean
gdk_display_request_selection_notification(GdkDisplay *
display,
GdkAtom
selection);
extern void gdk_display_set_double_click_distance(GdkDisplay *
display,
                                guint distance);
extern void gdk_display_set_double_click_time(GdkDisplay * display,
                                guint msec);
extern
GdkDisplayPointerHooks
*gdk_display_set_pointer_hooks(GdkDisplay *
display,
const
GdkDisplayPointerHooks
* new_hooks);
extern void gdk_display_store_clipboard(GdkDisplay * display,
                                GdkWindow * clipboard_window,
                                guint32 time_,
                                const GdkAtom * targets,
                                gint n_targets);
extern
gboolean
gdk_display_supports_clipboard_persistence(GdkDisplay *

```



```

display);
extern gboolean gdk_display_supports_cursor_alpha(GdkDisplay *
display);
extern gboolean gdk_display_supports_cursor_color(GdkDisplay *
display);
extern gboolean gdk_display_supports_input_shapes(GdkDisplay *
display);
extern
gboolean
gdk_display_supports_selection_notification(GdkDisplay *
display);
extern gboolean gdk_display_supports_shapes(GdkDisplay * display);
extern void gdk_display_sync(GdkDisplay * display);
extern void gdk_display_warp_pointer(GdkDisplay * display,
GdkScreen * screen, gint x, gint y);
extern void gdk_drag_abort(GdkDragContext * context, guint32 time_);
extern GType gdk_drag_action_get_type(void);
extern GdkDragContext *gdk_drag_begin(GdkWindow * window, GList *
targets);
extern GType gdk_drag_context_get_type(void);
extern GdkDragContext *gdk_drag_context_new(void);
extern void gdk_drag_drop(GdkDragContext * context, guint32 time_);
extern gboolean gdk_drag_drop_succeeded(GdkDragContext * context);
extern void gdk_drag_find_window(GdkDragContext * context,
GdkWindow * drag_window, gint x_root,
gint y_root, GdkWindow * *dest_window,
GdkDragProtocol * protocol);
extern void gdk_drag_find_window_for_screen(GdkDragContext *
context,
GdkWindow * drag_window,
GdkScreen * screen,
gint x_root, gint y_root,
GdkWindow * *dest_window,
GdkDragProtocol * protocol);
extern guint32 gdk_drag_get_protocol(guint32 xid,
GdkDragProtocol * protocol);
extern guint32 gdk_drag_get_protocol_for_display(GdkDisplay *
display,
guint32 xid,
GdkDragProtocol *
protocol);
extern GdkAtom gdk_drag_get_selection(GdkDragContext * context);
extern gboolean gdk_drag_motion(GdkDragContext * context,
GdkWindow * dest_window,
GdkDragProtocol protocol, gint x_root,
gint y_root,
GdkDragAction suggested_action,
GdkDragAction possible_actions,
guint32 time_);
extern GType gdk_drag_protocol_get_type(void);
extern void gdk_drag_status(GdkDragContext * context, GdkDragAction
action,
guint32 time_);
extern void gdk_draw_arc(GdkDrawable * drawable, GdkGC * gc,
gboolean filled, gint x, gint y, gint width,
gint height, gint angle1, gint angle2);
extern void gdk_draw_drawable(GdkDrawable * drawable, GdkGC * gc,
GdkDrawable * src, gint xsrc, gint ysrc,
gint xdest, gint ydest, gint width,
gint height);
extern void gdk_draw_glyphs(GdkDrawable * drawable, GdkGC * gc,
PangoFont * font, gint x, gint y,
PangoGlyphString * glyphs);
extern void gdk_draw_glyphs_transformed(GdkDrawable * drawable,
GdkGC * gc,
const PangoMatrix * matrix,
PangoFont * font, gint x, gint y,

```

```

        PangoGlyphString * glyphs);
extern void gdk_draw_gray_image(GdkDrawable * drawable, GdkGC * gc,
    gint x,
        gint y, gint width, gint height,
        GdkRgbDither dith, const guchar * buf,
        gint rowstride);
extern void gdk_draw_image(GdkDrawable * drawable, GdkGC * gc,
    GdkImage * image, gint xsrc, gint ysrc,
    gint xdest, gint ydest, gint width,
    gint height);
extern void gdk_draw_indexed_image(GdkDrawable * drawable, GdkGC *
    gc,
        gint x, gint y, gint width, gint
    height,
        GdkRgbDither dith, const guchar * buf,
        gint rowstride, GdkRgbCmap * cmap);
extern void gdk_draw_layout(GdkDrawable * drawable, GdkGC * gc, gint
    x,
        int y, PangoLayout * layout);
extern void gdk_draw_layout_line(GdkDrawable * drawable, GdkGC *
    gc,
        gint x, gint y, PangoLayoutLine * line);
extern void gdk_draw_layout_line_with_colors(GdkDrawable *
    drawable,
        GdkGC * gc, gint x, gint y,
        PangoLayoutLine * line,
        const GdkColor * foreground,
        const GdkColor * background);
extern void gdk_draw_layout_with_colors(GdkDrawable * drawable,
    GdkGC * gc,
        int x, int y, PangoLayout *
    layout,
        const GdkColor * foreground,
        const GdkColor * background);
extern void gdk_draw_line(GdkDrawable * drawable, GdkGC * gc, gint
    x1_,
        gint y1_, gint x2_, gint y2_);
extern void gdk_draw_lines(GdkDrawable * drawable, GdkGC * gc,
    const GdkPoint * points, gint npoints);
extern void gdk_draw_pixbuf(GdkDrawable * drawable, GdkGC * gc,
    const GdkPixbuf * pixbuf, gint src_x,
    gint src_y, gint dest_x, gint dest_y,
    gint width, gint height, GdkRgbDither
    dither,
        gint x_dither, gint y_dither);
extern void gdk_draw_point(GdkDrawable * drawable, GdkGC * gc, gint
    x,
        gint y);
extern void gdk_draw_points(GdkDrawable * drawable, GdkGC * gc,
    const GdkPoint * points, gint npoints);
extern void gdk_draw_polygon(GdkDrawable * drawable, GdkGC * gc,
    gboolean filled, const GdkPoint * points,
    gint npoints);
extern void gdk_draw_rectangle(GdkDrawable * drawable, GdkGC * gc,
    gboolean filled, gint x, gint y, gint
    width,
        gint height);
extern void gdk_draw_rgb_32_image(GdkDrawable * drawable, GdkGC *
    gc,
        gint x, gint y, gint width, gint height,
        GdkRgbDither dith, const guchar * buf,
        gint rowstride);
extern void gdk_draw_rgb_32_image_dithalign(GdkDrawable * drawable,
    GdkGC * gc, gint x, gint y,
    gint width, gint height,
    GdkRgbDither dith,

```

```

const guchar * buf,
gint rowstride, gint xdith,
gint ydith);
extern void gdk_draw_rgb_image(GdkDrawable * drawable, GdkGC * gc,
gint x,
gint y, gint width, gint height,
GdkRgbDither dith, const guchar * rgb_buf,
gint rowstride);
extern void gdk_draw_rgb_image_dithalign(GdkDrawable * drawable,
GdkGC * gc, gint x, gint y,
gint width, gint height,
GdkRgbDither dith,
const guchar * rgb_buf,
gint rowstride, gint xdith,
gint ydith);
extern void gdk_draw_segments(GdkDrawable * drawable, GdkGC * gc,
const GdkSegment * segs, gint nsegs);
extern void gdk_draw_trapezoids(GdkDrawable * drawable, GdkGC * gc,
const GdkTrapezoid * trapezoids,
gint n_trapezoids);
extern GdkImage *gdk_drawable_copy_to_image(GdkDrawable * drawable,
GdkImage * image, gint src_x,
gint src_y, gint dest_x,
gint dest_y, gint width,
gint height);
extern GdkRegion *gdk_drawable_get_clip_region(GdkDrawable *
drawable);
extern GdkColormap *gdk_drawable_get_colormap(GdkDrawable *
drawable);
extern gint gdk_drawable_get_depth(GdkDrawable * drawable);
extern GdkDisplay *gdk_drawable_get_display(GdkDrawable *
drawable);
extern GdkImage *gdk_drawable_get_image(GdkDrawable * drawable,
gint x,
gint y, gint width, gint height);
extern GdkScreen *gdk_drawable_get_screen(GdkDrawable * drawable);
extern void gdk_drawable_get_size(GdkDrawable * drawable, gint *
width,
gint * height);
extern GType gdk_drawable_get_type(void);
extern GdkRegion *gdk_drawable_get_visible_region(GdkDrawable *
drawable);
extern GdkVisual *gdk_drawable_get_visual(GdkDrawable * drawable);
extern void gdk_drawable_set_colormap(GdkDrawable * drawable,
GdkColormap * colormap);
extern void gdk_drop_finish(GdkDragContext * context, gboolean
success,
guint32 time_);
extern void gdk_drop_reply(GdkDragContext * context, gboolean ok,
guint32 time_);
extern gint gdk_error_trap_pop(void);
extern void gdk_error_trap_push(void);
extern GdkEvent *gdk_event_copy(const GdkEvent * event);
extern void gdk_event_free(GdkEvent * event);
extern GdkEvent *gdk_event_get(void);
extern gboolean gdk_event_get_axis(const GdkEvent * event,
GdkAxisUse axis_use, gdouble * value);
extern gboolean gdk_event_get_coords(const GdkEvent * event,
gdouble * x_win, gdouble * y_win);
extern GdkEvent *gdk_event_get_graphics_expose(GdkWindow * window);
extern gboolean gdk_event_get_root_coords(const GdkEvent * event,
gdouble * x_root,
gdouble * y_root);
extern GdkScreen *gdk_event_get_screen(const GdkEvent * event);
extern gboolean gdk_event_get_state(const GdkEvent * event,
GdkModifierType * state);

```

```

extern guint32 gdk_event_get_time(const GdkEvent * event);
extern GType gdk_event_get_type(void);
extern void gdk_event_handler_set(GdkEventFunc func, gpointer data,
                                  GDestroyNotify notify);
extern GType gdk_event_mask_get_type(void);
extern GdkEvent *gdk_event_new(GdkEventType type);
extern GdkEvent *gdk_event_peek(void);
extern void gdk_event_put(const GdkEvent * event);
extern gboolean gdk_event_send_client_message(GdkEvent * event,
                                              GdkNativeWindow winid);
extern
gdk_event_send_client_message_for_display(GdkDisplay *
                                          display,
                                          GdkEvent * event,
                                          GdkNativeWindow
                                          winid);

extern void gdk_event_send_clientmessage_toall(GdkEvent * event);
extern void gdk_event_set_screen(GdkEvent * event, GdkScreen *
screen);
extern GType gdk_event_type_get_type(void);
extern gboolean gdk_events_pending(void);
extern GType gdk_extension_mode_get_type(void);
extern GType gdk_fill_get_type(void);
extern GType gdk_fill_rule_get_type(void);
extern GType gdk_filter_return_get_type(void);
extern void gdk_flush(void);
extern GType gdk_font_type_get_type(void);
extern void gdk_free_compound_text(guchar * ctext);
extern void gdk_free_text_list(gchar * *list);
extern GType gdk_function_get_type(void);
extern void gdk_gc_copy(GdkGC * dst_gc, GdkGC * src_gc);
extern GdkColormap *gdk_gc_get_colormap(GdkGC * gc);
extern GdkScreen *gdk_gc_get_screen(GdkGC * gc);
extern GType gdk_gc_get_type(void);
extern void gdk_gc_get_values(GdkGC * gc, GdkGCValues * values);
extern GdkGC *gdk_gc_new(GdkDrawable * drawable);
extern GdkGC *gdk_gc_new_with_values(GdkDrawable * drawable,
                                     GdkGCValues * values,
                                     GdkGCValuesMask values_mask);
extern void gdk_gc_offset(GdkGC * gc, gint x_offset, gint y_offset);
extern void gdk_gc_set_background(GdkGC * gc, const GdkColor *
color);
extern void gdk_gc_set_clip_mask(GdkGC * gc, GdkBitmap * mask);
extern void gdk_gc_set_clip_origin(GdkGC * gc, gint x, gint y);
extern void gdk_gc_set_clip_rectangle(GdkGC * gc,
                                       const GdkRectangle * rectangle);
extern void gdk_gc_set_clip_region(GdkGC * gc, const GdkRegion *
region);
extern void gdk_gc_set_colormap(GdkGC * gc, GdkColormap * colormap);
extern void gdk_gc_set_dashes(GdkGC * gc, gint dash_offset,
                              gint8 * dash_list, gint n);
extern void gdk_gc_set_exposures(GdkGC * gc, gboolean exposures);
extern void gdk_gc_set_fill(GdkGC * gc, GdkFill fill);
extern void gdk_gc_set_foreground(GdkGC * gc, const GdkColor *
color);
extern void gdk_gc_set_function(GdkGC * gc, GdkFunction function);
extern void gdk_gc_set_line_attributes(GdkGC * gc, gint line_width,
                                       GdkLineStyle line_style,
                                       GdkCapStyle cap_style,
                                       GdkJoinStyle join_style);
extern void gdk_gc_set_rgb_bg_color(GdkGC * gc, const GdkColor *
color);
extern void gdk_gc_set_rgb_fg_color(GdkGC * gc, const GdkColor *
color);
extern void gdk_gc_set_stipple(GdkGC * gc, GdkPixmap * stipple);
extern void gdk_gc_set_subwindow(GdkGC * gc, GdkSubwindowMode mode);

```

```

extern void gdk_gc_set_tile(GdkGC * gc, GdkPixmap * tile);
extern void gdk_gc_set_ts_origin(GdkGC * gc, gint x, gint y);
extern void gdk_gc_set_values(GdkGC * gc, GdkGCValues * values,
                              GdkGCValuesMask values_mask);
extern GType gdk_gc_values_mask_get_type(void);
extern GdkWindow *gdk_get_default_root_window(void);
extern gchar *gdk_get_display(void);
extern const gchar *gdk_get_display_arg_name(void);
extern const char *gdk_get_program_class(void);
extern gboolean gdk_get_show_events(void);
extern GType gdk_grab_status_get_type(void);
extern GType gdk_gravity_get_type(void);
extern GdkColormap *gdk_image_get_colormap(GdkImage * image);
extern guint32 gdk_image_get_pixel(GdkImage * image, gint x, gint
y);
extern GType gdk_image_get_type(void);
extern GdkImage *gdk_image_new(GdkImageType type, GdkVisual *
visual,
                              gint width, gint height);
extern void gdk_image_put_pixel(GdkImage * image, gint x, gint y,
guint32 pixel);
extern void gdk_image_set_colormap(GdkImage * image,
GdkColormap * colormap);
extern GType gdk_image_type_get_type(void);
extern void gdk_init(int *argc, char ***argv);
extern gboolean gdk_init_check(int *argc, char ***argv);
extern GType gdk_input_condition_get_type(void);
extern GType gdk_input_mode_get_type(void);
extern void gdk_input_set_extension_events(GdkWindow * window, gint
mask,
GdkExtensionMode mode);
extern GType gdk_input_source_get_type(void);
extern GType gdk_join_style_get_type(void);
extern GdkGrabStatus gdk_keyboard_grab(GdkWindow * window,
gboolean owner_events,
guint32 time_);
extern void gdk_keyboard_ungrab(guint32 time_);
extern GdkKeymap *gdk_keymap_get_default(void);
extern PangoDirection gdk_keymap_get_direction(GdkKeymap * keymap);
extern gboolean gdk_keymap_get_entries_for_keycode(GdkKeymap *
keymap,
guint hardware_keycode,
GdkKeymapKey * *keys,
guint * *keyvals,
gint * n_entries);
extern gboolean gdk_keymap_get_entries_for_keyval(GdkKeymap *
keymap,
guint keyval,
GdkKeymapKey * *keys,
gint * n_keys);
extern GdkKeymap *gdk_keymap_get_for_display(GdkDisplay * display);
extern GType gdk_keymap_get_type(void);
extern guint gdk_keymap_lookup_key(GdkKeymap * keymap,
const GdkKeymapKey * key);
extern gboolean gdk_keymap_translate_keyboard_state(GdkKeymap *
keymap,
guint hardware_keycode,
GdkModifierType state,
gint group,
guint * keyval,
gint * effective_group,
gint * level,
GdkModifierType *
consumed_modifiers);
extern void gdk_keyval_convert_case(guint symbol, guint * lower,
guint * upper);

```

```

extern guint gdk_keyval_from_name(const gchar * keyval_name);
extern gboolean gdk_keyval_is_lower(guint keyval);
extern gboolean gdk_keyval_is_upper(guint keyval);
extern gchar *gdk_keyval_name(guint keyval);
extern guint gdk_keyval_to_lower(guint keyval);
extern guint32 gdk_keyval_to_unicode(guint keyval);
extern guint gdk_keyval_to_upper(guint keyval);
extern GType gdk_line_style_get_type(void);
extern GList *gdk_list_visuals(void);
extern GType gdk_modifier_type_get_type(void);
extern void gdk_notify_startup_complete(void);
extern GType gdk_notify_type_get_type(void);
extern GType gdk_overlap_type_get_type(void);
extern GType gdk_owner_change_get_type(void);
extern PangoAttribute *gdk_pango_attr_embossed_new(gboolean
embossed);
extern PangoAttribute *gdk_pango_attr_stipple_new(GdkBitmap *
stipple);
extern PangoContext *gdk_pango_context_get(void);
extern PangoContext *gdk_pango_context_get_for_screen(GdkScreen *
screen);
extern GdkRegion *gdk_pango_layout_get_clip_region(PangoLayout *
layout,
gint x_origin,
gint y_origin,
const gint *
index_ranges,
gint n_ranges);
extern GdkRegion
*gdk_pango_layout_line_get_clip_region(PangoLayoutLine *
line,
gint x_origin,
gint y_origin,
const gint *
index_ranges,
gint n_ranges);
extern PangoRenderer *gdk_pango_renderer_get_default(GdkScreen *
screen);
extern GType gdk_pango_renderer_get_type(void);
extern PangoRenderer *gdk_pango_renderer_new(GdkScreen * screen);
extern void gdk_pango_renderer_set_drawable(GdkPangoRenderer *
gdk_renderer,
GdkDrawable * drawable);
extern void gdk_pango_renderer_set_gc(GdkPangoRenderer *
gdk_renderer,
GdkGC * gc);
extern void gdk_pango_renderer_set_override_color(GdkPangoRenderer
*
gdk_renderer,
PangoRenderPart part,
const GdkColor * color);
extern void gdk_pango_renderer_set_stipple(GdkPangoRenderer *
gdk_renderer,
PangoRenderPart part,
GdkBitmap * stipple);
extern void gdk_parse_args(int *argc, char ***argv);
extern GdkPixbuf *gdk_pixbuf_get_from_drawable(GdkPixbuf * dest,
GdkDrawable * src,
GdkColormap * cmap,
int src_x, int src_y,
int dest_x, int dest_y,
int width, int height);
extern GdkPixbuf *gdk_pixbuf_get_from_image(GdkPixbuf * dest,
GdkImage * src,
GdkColormap * cmap, int src_x,
int src_y, int dest_x,

```

```

        int dest_y, int width,
        int height);
extern void gdk_pixbuf_render_pixmap_and_mask(GdkPixbuf * pixbuf,
        GdkPixmap * *pixmap_return,
        GdkBitmap * *mask_return,
        int alpha_threshold);
extern
gdk_pixbuf_render_pixmap_and_mask_for_colormap(GdkPixbuf *
        pixbuf,
        GdkColormap *
        colormap,
        GdkPixmap *
        *pixmap_return,
        GdkBitmap *
        *mask_return,
        int
        alpha_threshold);
extern void gdk_pixbuf_render_threshold_alpha(GdkPixbuf * pixbuf,
        GdkBitmap * bitmap,
        int src_x, int src_y,
        int dest_x, int dest_y,
        int width, int height,
        int alpha_threshold);
extern GdkPixmap *gdk_pixmap_colormap_create_from_xpm(GdkDrawable
*
        drawable,
        GdkColormap *
        colormap,
        GdkBitmap * *mask,
        const GdkColor *
        transparent_color,
        const gchar *
        filename);
extern
GdkPixmap
*gdk_pixmap_colormap_create_from_xpm_d(GdkDrawable *
        drawable,
        GdkColormap *
        colormap,
        GdkBitmap * *mask,
        const GdkColor *
        transparent_color,
        gchar * *data);
extern GdkPixmap *gdk_pixmap_create_from_data(GdkDrawable *
drawable,
        const gchar * data,
        gint width, gint height,
        gint depth,
        const GdkColor * fg,
        const GdkColor * bg);
extern GdkPixmap *gdk_pixmap_create_from_xpm(GdkDrawable *
drawable,
        GdkBitmap * *mask,
        const GdkColor *
        transparent_color,
        const gchar * filename);
extern GdkPixmap *gdk_pixmap_create_from_xpm_d(GdkDrawable *
drawable,
        GdkBitmap * *mask,
        const GdkColor *
        transparent_color,
        gchar * *data);
extern GdkPixmap *gdk_pixmap_foreign_new(GdkNativeWindow anid);
extern GdkPixmap *gdk_pixmap_foreign_new_for_display(GdkDisplay *
display,
        GdkNativeWindow anid);

```

```

extern GdkPixmap *gdk_pixmap_foreign_new_for_screen(GdkScreen *
screen,
                                                    GdkNativeWindow anid,
                                                    gint width,
                                                    gint height,
                                                    gint depth);

extern GType gdk_pixmap_get_type(void);
extern GdkPixmap *gdk_pixmap_lookup(GdkNativeWindow anid);
extern GdkPixmap *gdk_pixmap_lookup_for_display(GdkDisplay *
display,
                                                    GdkNativeWindow anid);
extern GdkPixmap *gdk_pixmap_new(GdkDrawable * drawable, gint width,
gint height, gint depth);
extern GdkGrabStatus gdk_pointer_grab(GdkWindow * window,
gboolean owner_events,
GdkEventMask event_mask,
GdkWindow * confine_to,
GdkCursor * cursor, guint32 time_);
extern gboolean gdk_pointer_is_grabbed(void);
extern void gdk_pointer_ungrab(guint32 time_);
extern GType gdk_prop_mode_get_type(void);
extern void gdk_property_change(GdkWindow * window, GdkAtom
property,
GdkAtom type, gint format,
GdkPropMode mode, const guchar * data,
gint nelements);
extern void gdk_property_delete(GdkWindow * window, GdkAtom
property);
extern gboolean gdk_property_get(GdkWindow * window, GdkAtom
property,
GdkAtom type, gulong offset,
gulong length, gint pdelete,
GdkAtom * actual_property_type,
gint * actual_format,
gint * actual_length, guchar * *data);
extern GType gdk_property_state_get_type(void);
extern void gdk_query_depths(gint * *depths, gint * count);
extern void gdk_query_visual_types(GdkVisualType * *visual_types,
gint * count);
extern GType gdk_rectangle_get_type(void);
extern gboolean gdk_rectangle_intersect(const GdkRectangle * src1,
const GdkRectangle * src2,
GdkRectangle * dest);
extern void gdk_rectangle_union(const GdkRectangle * src1,
const GdkRectangle * src2,
GdkRectangle * dest);
extern GdkRegion *gdk_region_copy(const GdkRegion * region);
extern void gdk_region_destroy(GdkRegion * region);
extern gboolean gdk_region_empty(const GdkRegion * region);
extern gboolean gdk_region_equal(const GdkRegion * region1,
const GdkRegion * region2);
extern void gdk_region_get_clipbox(const GdkRegion * region,
GdkRectangle * rectangle);
extern void gdk_region_get_rectangles(const GdkRegion * region,
GdkRectangle * *rectangles,
gint * n_rectangles);
extern void gdk_region_intersect(GdkRegion * source1,
const GdkRegion * source2);
extern GdkRegion *gdk_region_new(void);
extern void gdk_region_offset(GdkRegion * region, gint dx, gint dy);
extern gboolean gdk_region_point_in(const GdkRegion * region, int
x,
int y);
extern GdkRegion *gdk_region_polygon(const GdkPoint * points, gint
npoints,
GdkFillRule fill_rule);

```



```

extern GdkOverlapType gdk_region_rect_in(const GdkRegion * region,
                                         const GdkRectangle * rectangle);
extern GdkRegion *gdk_region_rectangle(const GdkRectangle *
rectangle);
extern void gdk_region_shrink(GdkRegion * region, int dx, int dy);
extern void gdk_region_spans_intersect_foreach(GdkRegion * region,
                                                const GdkSpan * spans,
                                                int n_spans,
                                                gboolean sorted,
                                                GdkSpanFunc function,
                                                gpointer data);
extern void gdk_region_subtract(GdkRegion * source1,
                                const GdkRegion * source2);
extern void gdk_region_union(GdkRegion * source1,
                             const GdkRegion * source2);
extern void gdk_region_union_with_rect(GdkRegion * region,
                                       const GdkRectangle * rect);
extern void gdk_region_xor(GdkRegion * source1, const GdkRegion *
source2);
extern void gdk_rgb_cmap_free(GdkRgbCmap * cmap);
extern GdkRgbCmap *gdk_rgb_cmap_new(guint32 * colors, gint
n_colors);
extern gboolean gdk_rgb_colormap_ditherable(GdkColormap * cmap);
extern GType gdk_rgb_dither_get_type(void);
extern gboolean gdk_rgb_ditherable(void);
extern void gdk_rgb_find_color(GdkColormap * colormap, GdkColor *
color);
extern GdkColormap *gdk_rgb_get_colormap(void);
extern GdkVisual *gdk_rgb_get_visual(void);
extern void gdk_rgb_set_install(gboolean install);
extern void gdk_rgb_set_min_colors(gint min_colors);
extern void gdk_rgb_set_verbose(gboolean verbose);
extern void gdk_screen_broadcast_client_message(GdkScreen * screen,
                                                GdkEvent * event);
extern GdkWindow *gdk_screen_get_active_window(GdkScreen * screen);
extern GdkScreen *gdk_screen_get_default(void);
extern GdkColormap *gdk_screen_get_default_colormap(GdkScreen *
screen);
extern GdkDisplay *gdk_screen_get_display(GdkScreen * screen);
extern const cairo_font_options_t
*gdk_screen_get_font_options(GdkScreen *
                                screen);
extern gint gdk_screen_get_height(GdkScreen * screen);
extern gint gdk_screen_get_height_mm(GdkScreen * screen);
extern gint gdk_screen_get_monitor_at_point(GdkScreen * screen,
gint x,
                                gint y);
extern gint gdk_screen_get_monitor_at_window(GdkScreen * screen,
GdkWindow * window);
extern void gdk_screen_get_monitor_geometry(GdkScreen * screen,
gint monitor_num,
GdkRectangle * dest);
extern gint gdk_screen_get_n_monitors(GdkScreen * screen);
extern gint gdk_screen_get_number(GdkScreen * screen);
extern gdouble gdk_screen_get_resolution(GdkScreen * screen);
extern GdkColormap *gdk_screen_get_rgb_colormap(GdkScreen *
screen);
extern GdkVisual *gdk_screen_get_rgb_visual(GdkScreen * screen);
extern GdkColormap *gdk_screen_get_rgba_colormap(GdkScreen *
screen);
extern GdkVisual *gdk_screen_get_rgba_visual(GdkScreen * screen);
extern GdkWindow *gdk_screen_get_root_window(GdkScreen * screen);
extern gboolean gdk_screen_get_setting(GdkScreen * screen,
const gchar * name, GValue *
value);

```

```

extern GdkColormap *gdk_screen_get_system_colormap(GdkScreen *
screen);
extern GdkVisual *gdk_screen_get_system_visual(GdkScreen * screen);
extern GList *gdk_screen_get_toplevel_windows(GdkScreen * screen);
extern GType gdk_screen_get_type(void);
extern gint gdk_screen_get_width(GdkScreen * screen);
extern gint gdk_screen_get_width_mm(GdkScreen * screen);
extern GList *gdk_screen_get_window_stack(GdkScreen * screen);
extern gint gdk_screen_height(void);
extern gint gdk_screen_height_mm(void);
extern gboolean gdk_screen_is_composited(GdkScreen * screen);
extern GList *gdk_screen_list_visuals(GdkScreen * screen);
extern gchar *gdk_screen_make_display_name(GdkScreen * screen);
extern void gdk_screen_set_default_colormap(GdkScreen * screen,
GdkColormap * colormap);
extern void gdk_screen_set_font_options(GdkScreen * screen,
const cairo_font_options_t *
options);
extern void gdk_screen_set_resolution(GdkScreen * screen, gdouble
dpi);
extern gint gdk_screen_width(void);
extern gint gdk_screen_width_mm(void);
extern GType gdk_scroll_direction_get_type(void);
extern void gdk_selection_convert(GdkWindow * requestor, GdkAtom
selection,
GdkAtom target, guint32 time_);
extern GdkWindow *gdk_selection_owner_get(GdkAtom selection);
extern GdkWindow *gdk_selection_owner_get_for_display(GdkDisplay *
display,
GdkAtom selection);
extern gboolean gdk_selection_owner_set(GdkWindow * owner,
GdkAtom selection, guint32 time_,
gboolean send_event);
extern gboolean gdk_selection_owner_set_for_display(GdkDisplay *
display,
GdkWindow * owner,
GdkAtom selection,
guint32 time_,
gboolean send_event);
extern gboolean gdk_selection_property_get(GdkWindow * requestor,
guchar * *data,
GdkAtom * prop_type,
gint * prop_format);
extern void gdk_selection_send_notify(guint32 requestor, GdkAtom
selection,
GdkAtom target, GdkAtom property,
guint32 time_);
extern void gdk_selection_send_notify_for_display(GdkDisplay *
display,
guint32 requestor,
GdkAtom selection,
GdkAtom target,
GdkAtom property,
guint32 time_);
extern void gdk_set_double_click_time(guint msec);
extern gchar *gdk_set_locale(void);
extern GdkPointerHooks *gdk_set_pointer_hooks(const
GdkPointerHooks *
new_hooks);
extern void gdk_set_program_class(const char *program_class);
extern void gdk_set_show_events(gboolean show_events);
extern void gdk_set_sm_client_id(const gchar * sm_client_id);
extern GType gdk_setting_action_get_type(void);
extern gboolean gdk_setting_get(const gchar * name, GValue * value);
extern gboolean gdk_spawn_command_line_on_screen(GdkScreen *
screen,

```

```

        const gchar *
        command_line,
        GError * *error);
extern gboolean gdk_spawn_on_screen(GdkScreen * screen,
        const gchar * working_directory,
        gchar * *argv, gchar * *envp,
        GSpawnFlags flags,
        GSpawnChildSetupFunc child_setup,
        gpointer user_data, gint * child_pid,
        GError * *error);
extern gboolean gdk_spawn_on_screen_with_pipes(GdkScreen * screen,
        const gchar *
        working_directory,
        gchar * *argv,
        gchar * *envp,
        GSpawnFlags flags,
        GSpawnChildSetupFunc
        child_setup,
        gpointer user_data,
        gint * child_pid,
        gint * standard_input,
        gint * standard_output,
        gint * standard_error,
        GError * *error);

extern GType gdk_status_get_type(void);
extern gint gdk_string_to_compound_text(const gchar * str,
        GdkAtom * encoding, gint * format,
        gchar * *ctext, gint * length);
extern gint gdk_string_to_compound_text_for_display(GdkDisplay *
display,
        const gchar * str,
        GdkAtom * encoding,
        gint * format,
        gchar * *ctext,
        gint * length);

extern GType gdk_subwindow_mode_get_type(void);
extern gint gdk_text_property_to_text_list(GdkAtom encoding, gint
format,
        const gchar * text,
        gint length, gchar * **list);
extern gint gdk_text_property_to_text_list_for_display(GdkDisplay
*
        display,
        GdkAtom encoding,
        gint format,
        const gchar * text,
        gint length,
        gchar * **list);
extern gint gdk_text_property_to_utf8_list(GdkAtom encoding, gint
format,
        const gchar * text,
        gint length, gchar * **list);
extern gint gdk_text_property_to_utf8_list_for_display(GdkDisplay
*
        display,
        GdkAtom encoding,
        gint format,
        const gchar * text,
        gint length,
        gchar * **list);

extern void gdk_threads_enter(void);
extern void gdk_threads_init(void);
extern void gdk_threads_leave(void);
extern GCallback gdk_threads_lock;
extern void gdk_threads_set_lock_functions(GCallback enter_fn,
        GCallback leave_fn);

```

```

extern GCallback gdk_threads_unlock;
extern guint gdk_unicode_to_keyval(guint32 wc);
extern gboolean gdk_utf8_to_compound_text(const gchar * str,
                                           GdkAtom * encoding,
                                           gint * format, gchar * *ctext,
                                           gint * length);
extern gboolean gdk_utf8_to_compound_text_for_display(GdkDisplay *
display,
                                           const gchar * str,
                                           GdkAtom * encoding,
                                           gint * format,
                                           gchar * *ctext,
                                           gint * length);

extern gchar *gdk_utf8_to_string_target(const gchar * str);
extern GType gdk_visibility_state_get_type(void);
extern GdkVisual *gdk_visual_get_best(void);
extern gint gdk_visual_get_best_depth(void);
extern GdkVisualType gdk_visual_get_best_type(void);
extern GdkVisual *gdk_visual_get_best_with_both(gint depth,
                                           GdkVisualType visual_type);
extern GdkVisual *gdk_visual_get_best_with_depth(gint depth);
extern GdkVisual *gdk_visual_get_best_with_type(GdkVisualType
visual_type);
extern GdkScreen *gdk_visual_get_screen(GdkVisual * visual);
extern GdkVisual *gdk_visual_get_system(void);
extern GType gdk_visual_get_type(void);
extern GType gdk_visual_type_get_type(void);
extern void gdk_window_add_filter(GdkWindow * window,
                                GdkFilterFunc function, gpointer data);
extern GdkWindow *gdk_window_at_pointer(gint * win_x, gint * win_y);
extern GType gdk_window_attributes_type_get_type(void);
extern void gdk_window_begin_move_drag(GdkWindow * window, gint
button,
                                gint root_x, gint root_y,
                                guint32 timestamp);
extern void gdk_window_begin_paint_rect(GdkWindow * window,
                                const GdkRectangle * rectangle);
extern void gdk_window_begin_paint_region(GdkWindow * window,
                                const GdkRegion * region);
extern void gdk_window_begin_resize_drag(GdkWindow * window,
                                GdkWindowEdge edge, gint button,
                                gint root_x, gint root_y,
                                guint32 timestamp);
extern GType gdk_window_class_get_type(void);
extern void gdk_window_clear(GdkWindow * window);
extern void gdk_window_clear_area(GdkWindow * window, gint x, gint
y,
                                gint width, gint height);
extern void gdk_window_clear_area_e(GdkWindow * window, gint x,
gint y,
                                gint width, gint height);
extern void gdk_window_configure_finished(GdkWindow * window);
extern void gdk_window_constrain_size(GdkGeometry * geometry, guint
flags,
                                gint width, gint height,
                                gint * new_width, gint *
new_height);
extern void gdk_window_deiconify(GdkWindow * window);
extern void gdk_window_destroy(GdkWindow * window);
extern GType gdk_window_edge_get_type(void);
extern void gdk_window_enable_synchronized_configure(GdkWindow *
window);
extern void gdk_window_end_paint(GdkWindow * window);
extern void gdk_window_focus(GdkWindow * window, guint32 timestamp);
extern GdkWindow *gdk_window_foreign_new(GdkNativeWindow anid);

```

```

extern GdkWindow *gdk_window_foreign_new_for_display(GdkDisplay *
display,
                                                    GdkNativeWindow anid);
extern void gdk_window_freeze_updates(GdkWindow * window);
extern void gdk_window_fullscreen(GdkWindow * window);
extern GList *gdk_window_get_children(GdkWindow * window);
extern gboolean gdk_window_get_decorations(GdkWindow * window,
                                           GdkWMDecoration * decorations);
extern GdkEventMask gdk_window_get_events(GdkWindow * window);
extern void gdk_window_get_frame_extents(GdkWindow * window,
                                           GdkRectangle * rect);
extern void gdk_window_get_geometry(GdkWindow * window, gint * x,
gint * y,
                                           gint * width, gint * height,
                                           gint * depth);
extern GdkWindow *gdk_window_get_group(GdkWindow * window);
extern void gdk_window_get_internal_paint_info(GdkWindow * window,
GdkDrawable *
*real_drawable,
gint * x_offset,
gint * y_offset);
extern gint gdk_window_get_origin(GdkWindow * window, gint * x,
gint * y);
extern GdkWindow *gdk_window_get_parent(GdkWindow * window);
extern GdkWindow *gdk_window_get_pointer(GdkWindow * window, gint
* x,
                                           gint * y, GdkModifierType *
mask);
extern void gdk_window_get_position(GdkWindow * window, gint * x,
gint * y);
extern void gdk_window_get_root_origin(GdkWindow * window, gint *
x,
gint * y);
extern GdkWindowState gdk_window_get_state(GdkWindow * window);
extern GdkWindow *gdk_window_get_toplevel(GdkWindow * window);
extern GList *gdk_window_get_toplevels(void);
extern GdkWindowTypeHint gdk_window_get_type_hint(GdkWindow *
window);
extern GdkRegion *gdk_window_get_update_area(GdkWindow * window);
extern void gdk_window_get_user_data(GdkWindow * window, gpointer
* data);
extern GdkWindowType gdk_window_get_window_type(GdkWindow *
window);
extern void gdk_window_hide(GdkWindow * window);
extern GType gdk_window_hints_get_type(void);
extern void gdk_window_iconify(GdkWindow * window);
extern void gdk_window_input_shape_combine_mask(GdkWindow * window,
GdkBitmap * mask, gint x,
gint y);
extern void gdk_window_input_shape_combine_region(GdkWindow *
window,
                                           const GdkRegion *
shape_region,
                                           gint offset_x,
                                           gint offset_y);
extern void gdk_window_invalidate_maybe_recurse(GdkWindow * window,
const GdkRegion * region,
gboolean(*child_func)
(GdkWindow *, gpointer),
gpointer user_data);
extern void gdk_window_invalidate_rect(GdkWindow * window,
const GdkRectangle * rect,
gboolean invalidate_children);
extern void gdk_window_invalidate_region(GdkWindow * window,
const GdkRegion * region,
gboolean invalidate_children);

```

```

extern gboolean gdk_window_is_viewable(GdkWindow * window);
extern gboolean gdk_window_is_visible(GdkWindow * window);
extern GdkWindow *gdk_window_lookup(GdkNativeWindow anid);
extern GdkWindow *gdk_window_lookup_for_display(GdkDisplay *
display,
GdkNativeWindow anid);
extern void gdk_window_lower(GdkWindow * window);
extern void gdk_window_maximize(GdkWindow * window);
extern void gdk_window_merge_child_input_shapes(GdkWindow *
window);
extern void gdk_window_merge_child_shapes(GdkWindow * window);
extern void gdk_window_move(GdkWindow * window, gint x, gint y);
extern void gdk_window_move_region(GdkWindow * window,
const GdkRegion * region, gint dx,
gint dy);
extern void gdk_window_move_resize(GdkWindow * window, gint x, gint
y,
gint width, gint height);
extern GdkWindow *gdk_window_new(GdkWindow * parent,
GdkWindowAttr * attributes,
gint attributes_mask);
extern GType gdk_window_object_get_type(void);
extern GList *gdk_window_peek_children(GdkWindow * window);
extern void gdk_window_process_all_updates(void);
extern void gdk_window_process_updates(GdkWindow * window,
gboolean update_children);
extern void gdk_window_raise(GdkWindow * window);
extern void gdk_window_register_dnd(GdkWindow * window);
extern void gdk_window_remove_filter(GdkWindow * window,
GdkFilterFunc function,
gpointer data);
extern void gdk_window_reparent(GdkWindow * window, GdkWindow *
new_parent,
gint x, gint y);
extern void gdk_window_resize(GdkWindow * window, gint width, gint
height);
extern void gdk_window_scroll(GdkWindow * window, gint dx, gint dy);
extern void gdk_window_set_accept_focus(GdkWindow * window,
gboolean accept_focus);
extern void gdk_window_set_back_pixmap(GdkWindow * window,
GdkPixmap * pixmap,
gboolean parent_relative);
extern void gdk_window_set_background(GdkWindow * window,
const GdkColor * color);
extern void gdk_window_set_child_input_shapes(GdkWindow * window);
extern void gdk_window_set_child_shapes(GdkWindow * window);
extern void gdk_window_set_cursor(GdkWindow * window, GdkCursor *
cursor);
extern void gdk_window_set_debug_updates(gboolean setting);
extern void gdk_window_set_decorations(GdkWindow * window,
GdkWMDecoration decorations);
extern void gdk_window_set_events(GdkWindow * window,
GdkEventMask event_mask);
extern void gdk_window_set_focus_on_map(GdkWindow * window,
gboolean focus_on_map);
extern void gdk_window_set_functions(GdkWindow * window,
GdkWMFunction functions);
extern void gdk_window_set_geometry_hints(GdkWindow * window,
const GdkGeometry * geometry,
GdkWindowHints geom_mask);
extern void gdk_window_set_group(GdkWindow * window, GdkWindow *
leader);
extern void gdk_window_set_icon(GdkWindow * window,
GdkWindow * icon_window,
GdkPixmap * pixmap, GdkBitmap * mask);

```

```

extern void gdk_window_set_icon_list(GdkWindow * window, GList *
pixbufs);
extern void gdk_window_set_icon_name(GdkWindow * window,
const gchar * name);
extern void gdk_window_set_keep_above(GdkWindow * window,
gboolean setting);
extern void gdk_window_set_keep_below(GdkWindow * window,
gboolean setting);
extern void gdk_window_set_modal_hint(GdkWindow * window, gboolean
modal);
extern void gdk_window_set_override_redirect(GdkWindow * window,
gboolean override_redirect);
extern void gdk_window_set_role(GdkWindow * window, const gchar *
role);
extern void gdk_window_set_skip_pager_hint(GdkWindow * window,
gboolean skips_pager);
extern void gdk_window_set_skip_taskbar_hint(GdkWindow * window,
gboolean skips_taskbar);
extern gboolean gdk_window_set_static_gravities(GdkWindow * window,
gboolean use_static);
extern void gdk_window_set_title(GdkWindow * window, const gchar *
title);
extern void gdk_window_set_transient_for(GdkWindow * window,
GdkWindow * parent);
extern void gdk_window_set_type_hint(GdkWindow * window,
GdkWindowTypeHint hint);
extern void gdk_window_set_urgency_hint(GdkWindow * window,
gboolean urgent);
extern void gdk_window_set_user_data(GdkWindow * window,
gpointer user_data);
extern void gdk_window_shape_combine_mask(GdkWindow * window,
GdkBitmap * mask, gint x,
gint y);
extern void gdk_window_shape_combine_region(GdkWindow * window,
const GdkRegion *
shape_region,
gint offset_x, gint offset_y);
extern void gdk_window_show(GdkWindow * window);
extern void gdk_window_show_unraised(GdkWindow * window);
extern GType gdk_window_state_get_type(void);
extern void gdk_window_stick(GdkWindow * window);
extern void gdk_window_thaw_updates(GdkWindow * window);
extern GType gdk_window_type_get_type(void);
extern GType gdk_window_type_hint_get_type(void);
extern void gdk_window_unfullscreen(GdkWindow * window);
extern void gdk_window_unmaximize(GdkWindow * window);
extern void gdk_window_unstick(GdkWindow * window);
extern void gdk_window_withdraw(GdkWindow * window);
extern GType gdk_wm_decoration_get_type(void);
extern GType gdk_wm_function_get_type(void);

```

17.30.2 gtk-2.0/gdk/gdkcairo.h

```

extern cairo_t *gdk_cairo_create(GdkDrawable * drawable);
extern void gdk_cairo_rectangle(cairo_t * cr,
const GdkRectangle * rectangle);
extern void gdk_cairo_region(cairo_t * cr, const GdkRegion *
region);
extern void gdk_cairo_set_source_color(cairo_t * cr,
const GdkColor * color);
extern void gdk_cairo_set_source_pixbuf(cairo_t * cr,
const GdkPixbuf * pixbuf,
double pixbuf_x, double pixbuf_y);
extern void gdk_cairo_set_source_pixmap(cairo_t * cr, GdkPixmap *
pixmap,

```

```
double pixmap_x, double pixmap_y);
```

17.30.3 gtk-2.0/gdk/gdkkeysyms.h

```
#define GDK_space      0x020
#define GDK_exclam     0x021
#define GDK_quotedbl   0x022
#define GDK_numbersign 0x023
#define GDK_dollar     0x024
#define GDK_percent    0x025
#define GDK_ampersand   0x026
#define GDK_apostrophe 0x027
#define GDK_quoteright 0x027
#define GDK_parenleft  0x028
#define GDK_parenright 0x029
#define GDK_asterisk    0x02a
#define GDK_plus        0x02b
#define GDK_comma       0x02c
#define GDK_minus       0x02d
#define GDK_period      0x02e
#define GDK_slash       0x02f
#define GDK_0           0x030
#define GDK_1           0x031
#define GDK_2           0x032
#define GDK_3           0x033
#define GDK_4           0x034
#define GDK_5           0x035
#define GDK_6           0x036
#define GDK_7           0x037
#define GDK_8           0x038
#define GDK_9           0x039
#define GDK_colon       0x03a
#define GDK_semicolon   0x03b
#define GDK_less        0x03c
#define GDK_equal       0x03d
#define GDK_greater     0x03e
#define GDK_question    0x03f
#define GDK_at          0x040
#define GDK_A           0x041
#define GDK_B           0x042
#define GDK_C           0x043
#define GDK_D           0x044
#define GDK_E           0x045
#define GDK_F           0x046
#define GDK_G           0x047
#define GDK_H           0x048
#define GDK_I           0x049
#define GDK_J           0x04a
#define GDK_K           0x04b
#define GDK_L           0x04c
#define GDK_M           0x04d
#define GDK_N           0x04e
#define GDK_O           0x04f
#define GDK_P           0x050
#define GDK_Q           0x051
#define GDK_R           0x052
#define GDK_S           0x053
#define GDK_T           0x054
#define GDK_U           0x055
#define GDK_V           0x056
#define GDK_W           0x057
#define GDK_X           0x058
#define GDK_Y           0x059
#define GDK_Z           0x05a
#define GDK_bracketleft 0x05b
```



```

#define GDK_backslash 0x05c
#define GDK_bracketright 0x05d
#define GDK_asciicircum 0x05e
#define GDK_underscore 0x05f
#define GDK_grave 0x060
#define GDK_quoteleft 0x060
#define GDK_a 0x061
#define GDK_b 0x062
#define GDK_c 0x063
#define GDK_d 0x064
#define GDK_e 0x065
#define GDK_f 0x066
#define GDK_g 0x067
#define GDK_h 0x068
#define GDK_i 0x069
#define GDK_j 0x06a
#define GDK_k 0x06b
#define GDK_l 0x06c
#define GDK_m 0x06d
#define GDK_n 0x06e
#define GDK_o 0x06f
#define GDK_p 0x070
#define GDK_q 0x071
#define GDK_r 0x072
#define GDK_s 0x073
#define GDK_t 0x074
#define GDK_u 0x075
#define GDK_v 0x076
#define GDK_w 0x077
#define GDK_x 0x078
#define GDK_y 0x079
#define GDK_z 0x07a
#define GDK_braceleft 0x07b
#define GDK_bar 0x07c
#define GDK_braceright 0x07d
#define GDK_asciitilde 0x07e
#define GDK_nobreakspace 0x0a0
#define GDK_exclamdown 0x0a1
#define GDK_cent 0x0a2
#define GDK_sterling 0x0a3
#define GDK_currency 0x0a4
#define GDK_yen 0x0a5
#define GDK_brokenbar 0x0a6
#define GDK_section 0x0a7
#define GDK_diaeresis 0x0a8
#define GDK_copyright 0x0a9
#define GDK_ordfeminine 0x0aa
#define GDK_guillemotleft 0x0ab
#define GDK_notsign 0x0ac
#define GDK_hyphen 0x0ad
#define GDK_registered 0x0ae
#define GDK_macron 0x0af
#define GDK_degree 0x0b0
#define GDK_plusminus 0x0b1
#define GDK_twosuperior 0x0b2
#define GDK_threesuperior 0x0b3
#define GDK_acute 0x0b4
#define GDK_mu 0x0b5
#define GDK_paragraph 0x0b6
#define GDK_periodcentered 0x0b7
#define GDK_cedilla 0x0b8
#define GDK_onesuperior 0x0b9
#define GDK_masculine 0x0ba
#define GDK_guillemotright 0x0bb
#define GDK_onequarter 0x0bc
#define GDK_onehalf 0x0bd

```

```

#define GDK_threequarters 0x0be
#define GDK_questiondown 0x0bf
#define GDK_Agrave 0x0c0
#define GDK_Aacute 0x0c1
#define GDK_Acircumflex 0x0c2
#define GDK_Atilde 0x0c3
#define GDK_Adiaeresis 0x0c4
#define GDK_Aring 0x0c5
#define GDK_AE 0x0c6
#define GDK_Ccedilla 0x0c7
#define GDK_Egrave 0x0c8
#define GDK_Eacute 0x0c9
#define GDK_Ecircumflex 0x0ca
#define GDK_Ediaeresis 0x0cb
#define GDK_Igrave 0x0cc
#define GDK_Iacute 0x0cd
#define GDK_Icircumflex 0x0ce
#define GDK_Idiaeresis 0x0cf
#define GDK_ETH 0x0d0
#define GDK_Eth 0x0d0
#define GDK_Ntilde 0x0d1
#define GDK_Ograve 0x0d2
#define GDK_Oacute 0x0d3
#define GDK_Ocircumflex 0x0d4
#define GDK_Otilde 0x0d5
#define GDK_Odiaeresis 0x0d6
#define GDK_multiply 0x0d7
#define GDK_Ooblique 0x0d8
#define GDK_Ugrave 0x0d9
#define GDK_Uacute 0x0da
#define GDK_Ucircumflex 0x0db
#define GDK_Udiaeresis 0x0dc
#define GDK_Yacute 0x0dd
#define GDK_THORN 0x0de
#define GDK_Thorn 0x0de
#define GDK_ssharp 0x0df
#define GDK_agrave 0x0e0
#define GDK_aacute 0x0e1
#define GDK_acircumflex 0x0e2
#define GDK_atilde 0x0e3
#define GDK_adiaeresis 0x0e4
#define GDK_aring 0x0e5
#define GDK_ae 0x0e6
#define GDK_ccedilla 0x0e7
#define GDK_egrave 0x0e8
#define GDK_eacute 0x0e9
#define GDK_ecircumflex 0x0ea
#define GDK_ediaeresis 0x0eb
#define GDK_igrave 0x0ec
#define GDK_iacute 0x0ed
#define GDK_icircumflex 0x0ee
#define GDK_idiaeresis 0x0ef
#define GDK_eth 0x0f0
#define GDK_ntilde 0x0f1
#define GDK_ograve 0x0f2
#define GDK_oacute 0x0f3
#define GDK_ocircumflex 0x0f4
#define GDK_otilde 0x0f5
#define GDK_odiaeresis 0x0f6
#define GDK_division 0x0f7
#define GDK_oslash 0x0f8
#define GDK_ugrave 0x0f9
#define GDK_uacute 0x0fa
#define GDK_ucircumflex 0x0fb
#define GDK_udiaeresis 0x0fc
#define GDK_yacute 0x0fd

```

```

#define GDK_thorn          0x0fe
#define GDK_ydiaeresis    0x0ff
#define GDK_OE            0x13bc
#define GDK_oe            0x13bd
#define GDK_Ydiaeresis    0x13be
#define GDK_Aogonek       0x1a1
#define GDK_breve         0x1a2
#define GDK_Lstroke       0x1a3
#define GDK_Lcaron        0x1a5
#define GDK_Sacute        0x1a6
#define GDK_Scaron        0x1a9
#define GDK_Scedilla      0x1aa
#define GDK_Tcaron        0x1ab
#define GDK_Zacute        0x1ac
#define GDK_Zcaron        0x1ae
#define GDK_Zabovedot     0x1af
#define GDK_aogonek       0x1b1
#define GDK_ogonek        0x1b2
#define GDK_lstroke       0x1b3
#define GDK_lcaron        0x1b5
#define GDK_sacute        0x1b6
#define GDK_caron         0x1b7
#define GDK_scaron        0x1b9
#define GDK_scedilla      0x1ba
#define GDK_tcaron        0x1bb
#define GDK_zacute        0x1bc
#define GDK_doubleacute    0x1bd
#define GDK_zcaron        0x1be
#define GDK_zabovedot     0x1bf
#define GDK_Racute        0x1c0
#define GDK_Abreve        0x1c3
#define GDK_Lacute        0x1c5
#define GDK_Cacute        0x1c6
#define GDK_Ccaron        0x1c8
#define GDK_Eogonek       0x1ca
#define GDK_Ecaron        0x1cc
#define GDK_Dcaron        0x1cd
#define GDK_Dstroke       0x1d0
#define GDK_Nacute        0x1d1
#define GDK_Ncaron        0x1d2
#define GDK_Odoubleacute  0x1d5
#define GDK_Rcaron        0x1d8
#define GDK_Uring         0x1d9
#define GDK_Udoubleacute  0x1db
#define GDK_Tcedilla      0x1de
#define GDK_racute        0x1e0
#define GDK_abreve        0x1e3
#define GDK_lacute        0x1e5
#define GDK_cacute        0x1e6
#define GDK_ccaron        0x1e8
#define GDK_eogonek       0x1ea
#define GDK_ecaron        0x1ec
#define GDK_dcaron        0x1ef
#define GDK_dstroke       0x1f0
#define GDK_nacute        0x1f1
#define GDK_ncaron        0x1f2
#define GDK_odoubleacute  0x1f5
#define GDK_rcaron        0x1f8
#define GDK_uring         0x1f9
#define GDK_udoubleacute  0x1fb
#define GDK_tcedilla      0x1fe
#define GDK_abovedot      0x1ff
#define GDK_EcuSign       0x20a0
#define GDK_ColonSign     0x20a1
#define GDK_CruzeiroSign  0x20a2
#define GDK_FFrancSign    0x20a3

```

```

#define GDK_LiraSign      0x20a4
#define GDK_MillSign      0x20a5
#define GDK_NairaSign     0x20a6
#define GDK_PesetaSign    0x20a7
#define GDK_RupeeSign     0x20a8
#define GDK_WonSign       0x20a9
#define GDK_NewSheqelSign 0x20aa
#define GDK_DongSign      0x20ab
#define GDK_EuroSign      0x20ac
#define GDK_Hstroke       0x2a1
#define GDK_Hcircumflex   0x2a6
#define GDK_Iabovedot     0x2a9
#define GDK_Gbreve        0x2ab
#define GDK_Jcircumflex   0x2ac
#define GDK_hstroke       0x2b1
#define GDK_hcircumflex   0x2b6
#define GDK_idotless      0x2b9
#define GDK_gbreve        0x2bb
#define GDK_jcircumflex   0x2bc
#define GDK_Cabovedot     0x2c5
#define GDK_Ccircumflex   0x2c6
#define GDK_Gabovedot     0x2d5
#define GDK_Gcircumflex   0x2d8
#define GDK_Ubreve        0x2dd
#define GDK_Scircumflex   0x2de
#define GDK_cabovedot     0x2e5
#define GDK_ccircumflex   0x2e6
#define GDK_gabovedot     0x2f5
#define GDK_gcircumflex   0x2f8
#define GDK_ubreve        0x2fd
#define GDK_scircumflex   0x2fe
#define GDK_kappa         0x3a2
#define GDK_kra            0x3a2
#define GDK_Rcedilla       0x3a3
#define GDK_Itilde         0x3a5
#define GDK_Lcedilla       0x3a6
#define GDK_Emacron        0x3aa
#define GDK_Gcedilla       0x3ab
#define GDK_Tslash         0x3ac
#define GDK_rcedilla       0x3b3
#define GDK_ityilde        0x3b5
#define GDK_lcedilla       0x3b6
#define GDK_emacron        0x3ba
#define GDK_gcedilla       0x3bb
#define GDK_tslash         0x3bc
#define GDK_ENG            0x3bd
#define GDK_eng            0x3bf
#define GDK_Amacron        0x3c0
#define GDK_Iogonek        0x3c7
#define GDK_Eabovedot     0x3cc
#define GDK_Imacron        0x3cf
#define GDK_Ncedilla       0x3d1
#define GDK_Omacron        0x3d2
#define GDK_Kcedilla       0x3d3
#define GDK_Uogonek        0x3d9
#define GDK_Utilde         0x3dd
#define GDK_Umacron        0x3de
#define GDK_amacron        0x3e0
#define GDK_iogonek        0x3e7
#define GDK_eabovedot     0x3ec
#define GDK_imacron        0x3ef
#define GDK_ncedilla       0x3f1
#define GDK_omacron        0x3f2
#define GDK_kcedilla       0x3f3
#define GDK_uogonek        0x3f9
#define GDK_utilde         0x3fd

```

```

#define GDK_umacron      0x3fe
#define GDK_overline     0x47e
#define GDK_kana_fullstop 0x4a1
#define GDK_kana_openingbracket 0x4a2
#define GDK_kana_closingbracket 0x4a3
#define GDK_kana_comma   0x4a4
#define GDK_kana_conjunctive 0x4a5
#define GDK_kana_middledot 0x4a5
#define GDK_kana_WO      0x4a6
#define GDK_kana_a       0x4a7
#define GDK_kana_i       0x4a8
#define GDK_kana_u       0x4a9
#define GDK_kana_e       0x4aa
#define GDK_kana_o       0x4ab
#define GDK_kana_ya      0x4ac
#define GDK_kana_yu      0x4ad
#define GDK_kana_yo      0x4ae
#define GDK_kana_tsu     0x4af
#define GDK_kana_tu      0x4af
#define GDK_prolongedsound 0x4b0
#define GDK_kana_A       0x4b1
#define GDK_kana_I       0x4b2
#define GDK_kana_U       0x4b3
#define GDK_kana_E       0x4b4
#define GDK_kana_O       0x4b5
#define GDK_kana_KA      0x4b6
#define GDK_kana_KI      0x4b7
#define GDK_kana_KU      0x4b8
#define GDK_kana_KE      0x4b9
#define GDK_kana_KO      0x4ba
#define GDK_kana_SA      0x4bb
#define GDK_kana_SHI     0x4bc
#define GDK_kana_SU      0x4bd
#define GDK_kana_SE      0x4be
#define GDK_kana_SO      0x4bf
#define GDK_kana_TA      0x4c0
#define GDK_kana_CHI     0x4c1
#define GDK_kana_TI      0x4c1
#define GDK_kana_TSU     0x4c2
#define GDK_kana_TU      0x4c2
#define GDK_kana_TB      0x4c3
#define GDK_kana_TO      0x4c4
#define GDK_kana_NA      0x4c5
#define GDK_kana_NI      0x4c6
#define GDK_kana_NU      0x4c7
#define GDK_kana_NE      0x4c8
#define GDK_kana_NO      0x4c9
#define GDK_kana_HA      0x4ca
#define GDK_kana_HI      0x4cb
#define GDK_kana_FU      0x4cc
#define GDK_kana_HU      0x4cc
#define GDK_kana_HE      0x4cd
#define GDK_kana_HO      0x4ce
#define GDK_kana_MA      0x4cf
#define GDK_kana_MI      0x4d0
#define GDK_kana_MU      0x4d1
#define GDK_kana_ME      0x4d2
#define GDK_kana_MO      0x4d3
#define GDK_kana_YA      0x4d4
#define GDK_kana_YU      0x4d5
#define GDK_kana_YO      0x4d6
#define GDK_kana_RA      0x4d7
#define GDK_kana_RI      0x4d8
#define GDK_kana_RU      0x4d9
#define GDK_kana_RE      0x4da
#define GDK_kana_RO      0x4db

```

```

#define GDK_kana_WA      0x4dc
#define GDK_kana_N       0x4dd
#define GDK_voicedsound 0x4de
#define GDK_semivoicedsound 0x4df
#define GDK_Arabic_comma 0x5ac
#define GDK_Arabic_semicolon 0x5bb
#define GDK_Arabic_question_mark 0x5bf
#define GDK_Arabic_hamza 0x5c1
#define GDK_Arabic_maddaonalef 0x5c2
#define GDK_Arabic_hamzaonalef 0x5c3
#define GDK_Arabic_hamzaonwaw 0x5c4
#define GDK_Arabic_hamzaunderalef 0x5c5
#define GDK_Arabic_hamzaonyeh 0x5c6
#define GDK_Arabic_alef 0x5c7
#define GDK_Arabic_beh 0x5c8
#define GDK_Arabic_tehmarbuta 0x5c9
#define GDK_Arabic_teh 0x5ca
#define GDK_Arabic_theh 0x5cb
#define GDK_Arabic_jeem 0x5cc
#define GDK_Arabic_hah 0x5cd
#define GDK_Arabic_khah 0x5ce
#define GDK_Arabic_dal 0x5cf
#define GDK_Arabic_thal 0x5d0
#define GDK_Arabic_ra 0x5d1
#define GDK_Arabic_zain 0x5d2
#define GDK_Arabic_seen 0x5d3
#define GDK_Arabic_sheen 0x5d4
#define GDK_Arabic_sad 0x5d5
#define GDK_Arabic_dad 0x5d6
#define GDK_Arabic_tah 0x5d7
#define GDK_Arabic_zah 0x5d8
#define GDK_Arabic_ain 0x5d9
#define GDK_Arabic_ghain 0x5da
#define GDK_Arabic_tatweel 0x5e0
#define GDK_Arabic_feh 0x5e1
#define GDK_Arabic_qaf 0x5e2
#define GDK_Arabic_kaf 0x5e3
#define GDK_Arabic_lam 0x5e4
#define GDK_Arabic_meem 0x5e5
#define GDK_Arabic_noon 0x5e6
#define GDK_Arabic_ha 0x5e7
#define GDK_Arabic_heh 0x5e7
#define GDK_Arabic_waw 0x5e8
#define GDK_Arabic_alefmaksura 0x5e9
#define GDK_Arabic_yeh 0x5ea
#define GDK_Arabic_fathatan 0x5eb
#define GDK_Arabic_dammatan 0x5ec
#define GDK_Arabic_kasratan 0x5ed
#define GDK_Arabic_fatha 0x5ee
#define GDK_Arabic_damma 0x5ef
#define GDK_Arabic_kasra 0x5f0
#define GDK_Arabic_shadda 0x5f1
#define GDK_Arabic_sukun 0x5f2
#define GDK_Serbian_dje 0x6a1
#define GDK_Macedonia_gje 0x6a2
#define GDK_Cyrillic_io 0x6a3
#define GDK_Ukrainian_ie 0x6a4
#define GDK_Ukrainian_je 0x6a4
#define GDK_Macedonia_dse 0x6a5
#define GDK_Ukrainian_i 0x6a6
#define GDK_Ukrainian_i 0x6a6
#define GDK_Ukrainian_yi 0x6a7
#define GDK_Ukrainian_yi 0x6a7
#define GDK_Cyrillic_je 0x6a8
#define GDK_Serbian_je 0x6a8
#define GDK_Cyrillic_lje 0x6a9

```

```

#define GDK_Serbian_lje 0x6a9
#define GDK_Cyrillic_nje 0x6aa
#define GDK_Serbian_nje 0x6aa
#define GDK_Serbian_tshe 0x6ab
#define GDK_Macedonia_kje 0x6ac
#define GDK_Ukrainian_ghe_with_upturn 0x6ad
#define GDK_Byelorussian_shortu 0x6ae
#define GDK_Cyrillic_dzhe 0x6af
#define GDK_Serbian_dze 0x6af
#define GDK_numerosign 0x6b0
#define GDK_Serbian_DJE 0x6b1
#define GDK_Macedonia_GJE 0x6b2
#define GDK_Cyrillic_IO 0x6b3
#define GDK_Ukrainian_IE 0x6b4
#define GDK_Ukrainian_JE 0x6b4
#define GDK_Macedonia_DSE 0x6b5
#define GDK_Ukrainian_I 0x6b6
#define GDK_Ukrainian_I 0x6b6
#define GDK_Ukrainian_YI 0x6b7
#define GDK_Ukrainian_YI 0x6b7
#define GDK_Cyrillic_JE 0x6b8
#define GDK_Serbian_JE 0x6b8
#define GDK_Cyrillic_LJE 0x6b9
#define GDK_Serbian_LJE 0x6b9
#define GDK_Cyrillic_NJE 0x6ba
#define GDK_Serbian_NJE 0x6ba
#define GDK_Serbian_TSHE 0x6bb
#define GDK_Macedonia_KJE 0x6bc
#define GDK_Ukrainian_GHE_WITH_UPTURN 0x6bd
#define GDK_Byelorussian_SHORTU 0x6be
#define GDK_Cyrillic_DZHE 0x6bf
#define GDK_Serbian_DZE 0x6bf
#define GDK_Cyrillic_yu 0x6c0
#define GDK_Cyrillic_a 0x6c1
#define GDK_Cyrillic_be 0x6c2
#define GDK_Cyrillic_tse 0x6c3
#define GDK_Cyrillic_de 0x6c4
#define GDK_Cyrillic_ie 0x6c5
#define GDK_Cyrillic_ef 0x6c6
#define GDK_Cyrillic_ghe 0x6c7
#define GDK_Cyrillic_ha 0x6c8
#define GDK_Cyrillic_i 0x6c9
#define GDK_Cyrillic_shorti 0x6ca
#define GDK_Cyrillic_ka 0x6cb
#define GDK_Cyrillic_el 0x6cc
#define GDK_Cyrillic_em 0x6cd
#define GDK_Cyrillic_en 0x6ce
#define GDK_Cyrillic_o 0x6cf
#define GDK_Cyrillic_pe 0x6d0
#define GDK_Cyrillic_ya 0x6d1
#define GDK_Cyrillic_er 0x6d2
#define GDK_Cyrillic_es 0x6d3
#define GDK_Cyrillic_te 0x6d4
#define GDK_Cyrillic_u 0x6d5
#define GDK_Cyrillic_zhe 0x6d6
#define GDK_Cyrillic_ve 0x6d7
#define GDK_Cyrillic_softsign 0x6d8
#define GDK_Cyrillic_yeru 0x6d9
#define GDK_Cyrillic_ze 0x6da
#define GDK_Cyrillic_sha 0x6db
#define GDK_Cyrillic_e 0x6dc
#define GDK_Cyrillic_shcha 0x6dd
#define GDK_Cyrillic_che 0x6de
#define GDK_Cyrillic_hardsign 0x6df
#define GDK_Cyrillic_YU 0x6e0
#define GDK_Cyrillic_A 0x6e1

```

```

#define GDK_Cyrillic_BE 0x6e2
#define GDK_Cyrillic_TSE 0x6e3
#define GDK_Cyrillic_DE 0x6e4
#define GDK_Cyrillic_IE 0x6e5
#define GDK_Cyrillic_EF 0x6e6
#define GDK_Cyrillic_GHE 0x6e7
#define GDK_Cyrillic_HA 0x6e8
#define GDK_Cyrillic_I 0x6e9
#define GDK_Cyrillic_SHORTI 0x6ea
#define GDK_Cyrillic_KA 0x6eb
#define GDK_Cyrillic_EL 0x6ec
#define GDK_Cyrillic_EM 0x6ed
#define GDK_Cyrillic_EN 0x6ee
#define GDK_Cyrillic_O 0x6ef
#define GDK_Cyrillic_PE 0x6f0
#define GDK_Cyrillic_YA 0x6f1
#define GDK_Cyrillic_ER 0x6f2
#define GDK_Cyrillic_ES 0x6f3
#define GDK_Cyrillic_TE 0x6f4
#define GDK_Cyrillic_U 0x6f5
#define GDK_Cyrillic_ZHE 0x6f6
#define GDK_Cyrillic_VE 0x6f7
#define GDK_Cyrillic_SOFTSIGN 0x6f8
#define GDK_Cyrillic_YERU 0x6f9
#define GDK_Cyrillic_ZE 0x6fa
#define GDK_Cyrillic_SHA 0x6fb
#define GDK_Cyrillic_E 0x6fc
#define GDK_Cyrillic_SHCHA 0x6fd
#define GDK_Cyrillic_CHE 0x6fe
#define GDK_Cyrillic_HARDSIGN 0x6ff
#define GDK_Greek_ALPHAaccent 0x7a1
#define GDK_Greek_EPSILONaccent 0x7a2
#define GDK_Greek_ETAaccent 0x7a3
#define GDK_Greek_IOTAaccent 0x7a4
#define GDK_Greek_IOTAdieresis 0x7a5
#define GDK_Greek_OMICRONaccent 0x7a7
#define GDK_Greek_UPSILONaccent 0x7a8
#define GDK_Greek_UPSILONdieresis 0x7a9
#define GDK_Greek_OMEGAaccent 0x7ab
#define GDK_Greek_accentdieresis 0x7ae
#define GDK_Greek_horizbar 0x7af
#define GDK_Greek_alphaaccent 0x7b1
#define GDK_Greek_epsilonaccent 0x7b2
#define GDK_Greek_etaaccent 0x7b3
#define GDK_Greek_iotaaccent 0x7b4
#define GDK_Greek_iotadieresis 0x7b5
#define GDK_Greek_iotaaccentdieresis 0x7b6
#define GDK_Greek_omicronaccent 0x7b7
#define GDK_Greek_upsilonaccent 0x7b8
#define GDK_Greek_upsilondieresis 0x7b9
#define GDK_Greek_upsilonaccentdieresis 0x7ba
#define GDK_Greek_omegaaccent 0x7bb
#define GDK_Greek_ALPHA 0x7c1
#define GDK_Greek_BETA 0x7c2
#define GDK_Greek_GAMMA 0x7c3
#define GDK_Greek_DELTA 0x7c4
#define GDK_Greek_EPSILON 0x7c5
#define GDK_Greek_ZETA 0x7c6
#define GDK_Greek_ETA 0x7c7
#define GDK_Greek_THETA 0x7c8
#define GDK_Greek_IOTA 0x7c9
#define GDK_Greek_KAPPA 0x7ca
#define GDK_Greek_LAMBDA 0x7cb
#define GDK_Greek_LAMDA 0x7cb
#define GDK_Greek_MU 0x7cc
#define GDK_Greek_NU 0x7cd

```



```

#define GDK_Greek_XI      0x7ce
#define GDK_Greek_OMICRON 0x7cf
#define GDK_Greek_PI      0x7d0
#define GDK_Greek_RHO     0x7d1
#define GDK_Greek_SIGMA   0x7d2
#define GDK_Greek_TAU     0x7d4
#define GDK_Greek_UPSILON 0x7d5
#define GDK_Greek_PHI     0x7d6
#define GDK_Greek_CHI     0x7d7
#define GDK_Greek_PSI     0x7d8
#define GDK_Greek_OMEGA   0x7d9
#define GDK_Greek_alpha   0x7e1
#define GDK_Greek_beta    0x7e2
#define GDK_Greek_gamma   0x7e3
#define GDK_Greek_delta   0x7e4
#define GDK_Greek_epsilon 0x7e5
#define GDK_Greek_zeta    0x7e6
#define GDK_Greek_eta     0x7e7
#define GDK_Greek_theta   0x7e8
#define GDK_Greek_iota    0x7e9
#define GDK_Greek_kappa   0x7ea
#define GDK_Greek_lambda  0x7eb
#define GDK_Greek_lamda   0x7eb
#define GDK_Greek_mu      0x7ec
#define GDK_Greek_nu      0x7ed
#define GDK_Greek_xi      0x7ee
#define GDK_Greek_omicron 0x7ef
#define GDK_Greek_pi      0x7f0
#define GDK_Greek_rho     0x7f1
#define GDK_Greek_sigma   0x7f2
#define GDK_Greek_finalsmallsigma 0x7f3
#define GDK_Greek_tau     0x7f4
#define GDK_Greek_upsilon 0x7f5
#define GDK_Greek_phi     0x7f6
#define GDK_Greek_chi     0x7f7
#define GDK_Greek_psi     0x7f8
#define GDK_Greek_omega   0x7f9
#define GDK_leftradical   0x8a1
#define GDK_topleftradical 0x8a2
#define GDK_horizconnector 0x8a3
#define GDK_topintegral   0x8a4
#define GDK_botintegral   0x8a5
#define GDK_vertconnector 0x8a6
#define GDK_topleftsqbracket 0x8a7
#define GDK_botleftsqbracket 0x8a8
#define GDK_toprightsqbracket 0x8a9
#define GDK_botrightsqbracket 0x8aa
#define GDK_topleftparens 0x8ab
#define GDK_botleftparens 0x8ac
#define GDK_toprightparens 0x8ad
#define GDK_botrightparens 0x8ae
#define GDK_leftmiddlecurlybrace 0x8af
#define GDK_rightmiddlecurlybrace 0x8b0
#define GDK_topleftsummation 0x8b1
#define GDK_botleftsummation 0x8b2
#define GDK_topvertsummationconnector 0x8b3
#define GDK_botvertsummationconnector 0x8b4
#define GDK_toprightsummation 0x8b5
#define GDK_botrightsummation 0x8b6
#define GDK_rightmiddlesummation 0x8b7
#define GDK_lessthanequal 0x8bc
#define GDK_notequal      0x8bd
#define GDK_greaterthanequal 0x8be
#define GDK_integral      0x8bf
#define GDK_therefore     0x8c0
#define GDK_variation     0x8c1

```

```

#define GDK_infinity      0x8c2
#define GDK_nabla         0x8c5
#define GDK_approximate  0x8c8
#define GDK_similarequal  0x8c9
#define GDK_ifonlyif      0x8cd
#define GDK_implies       0x8ce
#define GDK_identical     0x8cf
#define GDK_radical       0x8d6
#define GDK_includedin    0x8da
#define GDK_includes      0x8db
#define GDK_intersection   0x8dc
#define GDK_union         0x8dd
#define GDK_logicaland    0x8de
#define GDK_logicalor     0x8df
#define GDK_partialderivative 0x8ef
#define GDK_function      0x8f6
#define GDK_leftarrow     0x8fb
#define GDK_uparrow       0x8fc
#define GDK_rightarrow    0x8fd
#define GDK_downarrow     0x8fe
#define GDK_blank         0x9df
#define GDK_soliddiamond  0x9e0
#define GDK_checkerboard  0x9e1
#define GDK_ht            0x9e2
#define GDK_ff            0x9e3
#define GDK_cr            0x9e4
#define GDK_lf            0x9e5
#define GDK_nl            0x9e8
#define GDK_vt            0x9e9
#define GDK_lowrightcorner 0x9ea
#define GDK_uprightcorner 0x9eb
#define GDK_uleftcorner   0x9ec
#define GDK_lowleftcorner 0x9ed
#define GDK_crossinglines 0x9ee
#define GDK_horizlinescan1 0x9ef
#define GDK_horizlinescan3 0x9f0
#define GDK_horizlinescan5 0x9f1
#define GDK_horizlinescan7 0x9f2
#define GDK_horizlinescan9 0x9f3
#define GDK_leftt         0x9f4
#define GDK_rightt        0x9f5
#define GDK_bott          0x9f6
#define GDK_topt          0x9f7
#define GDK_vertbar       0x9f8
#define GDK_emspace       0xaa1
#define GDK_enspace       0xaa2
#define GDK_em3space      0xaa3
#define GDK_em4space      0xaa4
#define GDK_digitspace    0xaa5
#define GDK_punctspace    0xaa6
#define GDK_thinspace     0xaa7
#define GDK_hairspace     0xaa8
#define GDK_emdash        0xaa9
#define GDK_endash        0xaaa
#define GDK_signifblank   0xaac
#define GDK_ellipsis      0xaae
#define GDK_doubbaselinedot 0xaaf
#define GDK_onethird      0xab0
#define GDK_twothirds     0xab1
#define GDK_onefifth      0xab2
#define GDK_twofifths     0xab3
#define GDK_threefifths   0xab4
#define GDK_fourfifths    0xab5
#define GDK_onesixth      0xab6
#define GDK_fivesixths    0xab7
#define GDK_careof        0xab8

```

```

#define GDK_figdash      0xab
#define GDK_leftanglebracket  0xabc
#define GDK_decimalpoint  0xabd
#define GDK_rightanglebracket  0xabe
#define GDK_marker      0xabf
#define GDK_oneeighth    0xac3
#define GDK_threeeighths  0xac4
#define GDK_fiveeighths  0xac5
#define GDK_seveneighths  0xac6
#define GDK_trademark    0xac9
#define GDK_signaturemark  0xaca
#define GDK_trademarkincircle  0xacb
#define GDK_leftopentriangle  0xacc
#define GDK_rightopentriangle  0xacd
#define GDK_emopencircle  0xace
#define GDK_emopenrectangle  0xacf
#define GDK_leftsinglequotemark  0xad0
#define GDK_rightsinglequotemark  0xad1
#define GDK_lefthdoublequotemark  0xad2
#define GDK_righthdoublequotemark  0xad3
#define GDK_prescription  0xad4
#define GDK_minutes      0xad6
#define GDK_seconds      0xad7
#define GDK_latincross   0xad9
#define GDK_hexagram     0xada
#define GDK_filledrectbullet  0xadb
#define GDK_filledlefttribullet  0xadc
#define GDK_filledrighttribullet  0xadd
#define GDK_emfilledcircle  0xade
#define GDK_emfilledrect  0xadf
#define GDK_enopencircbullet  0xae0
#define GDK_enopensquarebullet  0xae1
#define GDK_openrectbullet  0xae2
#define GDK_opentribulletup  0xae3
#define GDK_opentribulletdown  0xae4
#define GDK_openstar      0xae5
#define GDK_enfilledcircbullet  0xae6
#define GDK_enfilledsqbullet  0xae7
#define GDK_filledtribulletup  0xae8
#define GDK_filledtribulletdown  0xae9
#define GDK_leftpointer  0xaea
#define GDK_rightpointer  0xaeb
#define GDK_club          0xaec
#define GDK_diamond       0xaed
#define GDK_heart         0xae
#define GDK_maltesecross  0xaf0
#define GDK_dagger        0xaf1
#define GDK_doubledagger  0xaf2
#define GDK_checkmark     0xaf3
#define GDK_ballotcross   0xaf4
#define GDK_musicalsharp  0xaf5
#define GDK_musicalflat  0xaf6
#define GDK_malesymbol    0xaf7
#define GDK_femalesymbol  0xaf8
#define GDK_telephone     0xaf9
#define GDK_telephonerecorder  0afa
#define GDK_phonographcopyright  0afb
#define GDK_caret         0afc
#define GDK_singlelowquotemark  0afd
#define GDK_doublelowquotemark  0afe
#define GDK_cursor        0aff
#define GDK_leftcaret     0xba3
#define GDK_rightcaret    0xba6
#define GDK_downcaret     0xba8
#define GDK_upcaret       0xba9
#define GDK_overbar       0xbc0

```

```

#define GDK_downtack 0xbc2
#define GDK_upshoe 0xbc3
#define GDK_downstile 0xbc4
#define GDK_underbar 0xbc6
#define GDK_jot 0xbca
#define GDK_quad 0xbcc
#define GDK_uptack 0xbce
#define GDK_circle 0xbcf
#define GDK_upstile 0xbd3
#define GDK_downshoe 0xbd6
#define GDK_rightshoe 0xbd8
#define GDK_leftshoe 0xbda
#define GDK_lefttack 0xbdc
#define GDK_righttack 0xbfc
#define GDK_hebrew_doublelowline 0xcdf
#define GDK_hebrew_aleph 0xce0
#define GDK_hebrew_bet 0xce1
#define GDK_hebrew_beth 0xce1
#define GDK_hebrew_gimel 0xce2
#define GDK_hebrew_gimmel 0xce2
#define GDK_hebrew_dalet 0xce3
#define GDK_hebrew_daleth 0xce3
#define GDK_hebrew_he 0xce4
#define GDK_hebrew_waw 0xce5
#define GDK_hebrew_zain 0xce6
#define GDK_hebrew_zayin 0xce6
#define GDK_hebrew_chet 0xce7
#define GDK_hebrew_het 0xce7
#define GDK_hebrew_tet 0xce8
#define GDK_hebrew_teth 0xce8
#define GDK_hebrew_yod 0xce9
#define GDK_hebrew_finalkaph 0xcea
#define GDK_hebrew_kaph 0xceb
#define GDK_hebrew_lamed 0xcec
#define GDK_hebrew_finalmem 0xc ed
#define GDK_hebrew_mem 0xcee
#define GDK_hebrew_finalnun 0xcef
#define GDK_hebrew_nun 0xcf0
#define GDK_hebrew_samech 0xcf1
#define GDK_hebrew_samekh 0xcf1
#define GDK_hebrew_ayin 0xcf2
#define GDK_hebrew_finalpe 0xcf3
#define GDK_hebrew_pe 0xcf4
#define GDK_hebrew_finalzade 0xcf5
#define GDK_hebrew_finalzadi 0xcf5
#define GDK_hebrew_zade 0xcf6
#define GDK_hebrew_zadi 0xcf6
#define GDK_hebrew_kuf 0xcf7
#define GDK_hebrew_qoph 0xcf7
#define GDK_hebrew_resh 0xcf8
#define GDK_hebrew_shin 0xcf9
#define GDK_hebrew_taf 0xcfa
#define GDK_hebrew_taw 0xcfa
#define GDK_Thai_kokai 0xda1
#define GDK_Thai_khokhai 0xda2
#define GDK_Thai_khokhuat 0xda3
#define GDK_Thai_khokhwai 0xda4
#define GDK_Thai_khokhon 0xda5
#define GDK_Thai_khorakhang 0xda6
#define GDK_Thai_ngongu 0xda7
#define GDK_Thai_chochan 0xda8
#define GDK_Thai_choching 0xda9
#define GDK_Thai_chochang 0xdaa
#define GDK_Thai_soso 0xdab
#define GDK_Thai_chochoe 0xdac
#define GDK_Thai_yoying 0xdad

```

```

#define GDK_Thai_dochada 0xdae
#define GDK_Thai_topatak 0xdaf
#define GDK_Thai_thothan 0xdb0
#define GDK_Thai_thonangmontho 0xdb1
#define GDK_Thai_thophuthao 0xdb2
#define GDK_Thai_nonen 0xdb3
#define GDK_Thai_dodek 0xdb4
#define GDK_Thai_totao 0xdb5
#define GDK_Thai_thothung 0xdb6
#define GDK_Thai_thothahan 0xdb7
#define GDK_Thai_thothong 0xdb8
#define GDK_Thai_nonu 0xdb9
#define GDK_Thai_bobaimai 0xdba
#define GDK_Thai_popla 0xdbb
#define GDK_Thai_phophung 0xdbc
#define GDK_Thai_fofa 0xdbd
#define GDK_Thai_phophan 0xdbe
#define GDK_Thai_fofan 0dbf
#define GDK_Thai_phosamphao 0xdc0
#define GDK_Thai_moma 0xdc1
#define GDK_Thai_yoyak 0xdc2
#define GDK_Thai_rorua 0xdc3
#define GDK_Thai_ru 0xdc4
#define GDK_Thai_loling 0xdc5
#define GDK_Thai_lu 0xdc6
#define GDK_Thai_wowaen 0xdc7
#define GDK_Thai_sosala 0xdc8
#define GDK_Thai_sorusi 0xdc9
#define GDK_Thai_sosua 0xdca
#define GDK_Thai_hohip 0xdcb
#define GDK_Thai_lochula 0xdcc
#define GDK_Thai_oang 0xcd
#define GDK_Thai_honokhuk 0xdce
#define GDK_Thai_paiyannoi 0xdcf
#define GDK_Thai_saraa 0xdd0
#define GDK_Thai_maihanakat 0xdd1
#define GDK_Thai_saraaa 0xdd2
#define GDK_Thai_saraam 0xdd3
#define GDK_Thai_sarai 0xdd4
#define GDK_Thai_saraii 0xdd5
#define GDK_Thai_saraue 0xdd6
#define GDK_Thai_sarauee 0xdd7
#define GDK_Thai_sarau 0xdd8
#define GDK_Thai_sarauu 0xdd9
#define GDK_Thai_phinthu 0xdda
#define GDK_Thai_maihanakat_maitho 0xdde
#define GDK_Thai_baht 0xddf
#define GDK_Thai_sarae 0xde0
#define GDK_Thai_saraae 0xde1
#define GDK_Thai_sarao 0xde2
#define GDK_Thai_saraaimaimuan 0xde3
#define GDK_Thai_saraaimaimalai 0xde4
#define GDK_Thai_lakkhangyao 0xde5
#define GDK_Thai_maiyamok 0xde6
#define GDK_Thai_maitaikhu 0xde7
#define GDK_Thai_maiek 0xde8
#define GDK_Thai_maitho 0xde9
#define GDK_Thai_maitri 0xdea
#define GDK_Thai_maichattawa 0xdeb
#define GDK_Thai_thanthakhat 0xdec
#define GDK_Thai_nikhahit 0xded
#define GDK_Thai_leksun 0xdf0
#define GDK_Thai_leknung 0xdf1
#define GDK_Thai_leksong 0xdf2
#define GDK_Thai_leksam 0xdf3
#define GDK_Thai_leksi 0xdf4

```

```

#define GDK_Thai_lekha 0xdf5
#define GDK_Thai_lekhok 0xdf6
#define GDK_Thai_lekchet 0xdf7
#define GDK_Thai_lekpaet 0xdf8
#define GDK_Thai_lekkao 0xdf9
#define GDK_Hangul_Kiyeog 0xea1
#define GDK_Hangul_SsangKiyeog 0xea2
#define GDK_Hangul_KiyeogSios 0xea3
#define GDK_Hangul_Nieun 0xea4
#define GDK_Hangul_NieunJieuj 0xea5
#define GDK_Hangul_NieunHieuh 0xea6
#define GDK_Hangul_Dikeud 0xea7
#define GDK_Hangul_SsangDikeud 0xea8
#define GDK_Hangul_Rieul 0xea9
#define GDK_Hangul_RieulKiyeog 0xaaa
#define GDK_Hangul_RieulMieum 0xaab
#define GDK_Hangul_RieulPieub 0xaac
#define GDK_Hangul_RieulSios 0xaad
#define GDK_Hangul_RieulTieut 0xaea
#define GDK_Hangul_RieulPhieuf 0xaea
#define GDK_Hangul_RieulHieuh 0xeb0
#define GDK_Hangul_Mieum 0xeb1
#define GDK_Hangul_Pieub 0xeb2
#define GDK_Hangul_SsangPieub 0xeb3
#define GDK_Hangul_PieubSios 0xeb4
#define GDK_Hangul_Sios 0xeb5
#define GDK_Hangul_SsangSios 0xeb6
#define GDK_Hangul_Ieung 0xeb7
#define GDK_Hangul_Jieuj 0xeb8
#define GDK_Hangul_SsangJieuj 0xeb9
#define GDK_Hangul_Cieuc 0xeba
#define GDK_Hangul_Khieug 0xebb
#define GDK_Hangul_Tieut 0xebc
#define GDK_Hangul_Phieuf 0xebd
#define GDK_Hangul_Hieuh 0xebf
#define GDK_Hangul_A 0xebf
#define GDK_Hangul_AE 0xec0
#define GDK_Hangul_YA 0xec1
#define GDK_Hangul_YAE 0xec2
#define GDK_Hangul_EO 0xec3
#define GDK_Hangul_E 0xec4
#define GDK_Hangul_YEO 0xec5
#define GDK_Hangul_YE 0xec6
#define GDK_Hangul_O 0xec7
#define GDK_Hangul_WA 0xec8
#define GDK_Hangul_WAE 0xec9
#define GDK_Hangul_OE 0xeca
#define GDK_Hangul_YO 0xecb
#define GDK_Hangul_U 0xecc
#define GDK_Hangul_WEO 0xecd
#define GDK_Hangul_WE 0xece
#define GDK_Hangul_WI 0xecf
#define GDK_Hangul_YU 0xed0
#define GDK_Hangul_EU 0xed1
#define GDK_Hangul_YI 0xed2
#define GDK_Hangul_I 0xed3
#define GDK_Hangul_J_Kiyeog 0xed4
#define GDK_Hangul_J_SsangKiyeog 0xed5
#define GDK_Hangul_J_KiyeogSios 0xed6
#define GDK_Hangul_J_Nieun 0xed7
#define GDK_Hangul_J_NieunJieuj 0xed8
#define GDK_Hangul_J_NieunHieuh 0xed9
#define GDK_Hangul_J_Dikeud 0xeda
#define GDK_Hangul_J_Rieul 0xedb
#define GDK_Hangul_J_RieulKiyeog 0xedc
#define GDK_Hangul_J_RieulMieum 0xedd

```

```

#define GDK_Hangul_J_RieulPieub 0xede
#define GDK_Hangul_J_RieulSios 0xedf
#define GDK_Hangul_J_RieulTieut 0xee0
#define GDK_Hangul_J_RieulPhieuf 0xee1
#define GDK_Hangul_J_RieulHieuh 0xee2
#define GDK_Hangul_J_Mieum 0xee3
#define GDK_Hangul_J_Pieub 0xee4
#define GDK_Hangul_J_PieubSios 0xee5
#define GDK_Hangul_J_Sios 0xee6
#define GDK_Hangul_J_SsangSios 0xee7
#define GDK_Hangul_J_Ieung 0xee8
#define GDK_Hangul_J_Jieuj 0xee9
#define GDK_Hangul_J_Cieuc 0xeea
#define GDK_Hangul_J_Khieug 0xeeb
#define GDK_Hangul_J_Tieut 0xeec
#define GDK_Hangul_J_Phieuf 0xeed
#define GDK_Hangul_J_Hieuh 0xeee
#define GDK_Hangul_RieulYeorinHieuh 0xeef
#define GDK_Hangul_SunkyeongeumMieum 0xef0
#define GDK_Hangul_SunkyeongeumPieub 0xef1
#define GDK_Hangul_PanSios 0xef2
#define GDK_Hangul_KkogjiDalrinIeung 0xef3
#define GDK_Hangul_SunkyeongeumPhieuf 0xef4
#define GDK_Hangul_YeorinHieuh 0xef5
#define GDK_Hangul_AraeA 0xef6
#define GDK_Hangul_AraeAE 0xef7
#define GDK_Hangul_J_PanSios 0xef8
#define GDK_Hangul_J_KkogjiDalrinIeung 0xef9
#define GDK_Hangul_J_YeorinHieuh 0xefa
#define GDK_Korean_Won 0xeff
#define GDK_3270_Duplicate 0xFD01
#define GDK_3270_FieldMark 0xFD02
#define GDK_3270_Right2 0xFD03
#define GDK_3270_Left2 0xFD04
#define GDK_3270_BackTab 0xFD05
#define GDK_3270_EraseEOF 0xFD06
#define GDK_3270_EraseInput 0xFD07
#define GDK_3270_Reset 0xFD08
#define GDK_3270_Quit 0xFD09
#define GDK_3270_PA1 0xFD0A
#define GDK_3270_PA2 0xFD0B
#define GDK_3270_PA3 0xFD0C
#define GDK_3270_Test 0xFD0D
#define GDK_3270_Attn 0xFD0E
#define GDK_3270_CursorBlink 0xFD0F
#define GDK_3270_AltCursor 0xFD10
#define GDK_3270_KeyClick 0xFD11
#define GDK_3270_Jump 0xFD12
#define GDK_3270_Ident 0xFD13
#define GDK_3270_Rule 0xFD14
#define GDK_3270_Copy 0xFD15
#define GDK_3270_Play 0xFD16
#define GDK_3270_Setup 0xFD17
#define GDK_3270_Record 0xFD18
#define GDK_3270_ChangeScreen 0xFD19
#define GDK_3270_DeleteWord 0xFD1A
#define GDK_3270_ExSelect 0xFD1B
#define GDK_3270_CursorSelect 0xFD1C
#define GDK_3270_PrintScreen 0xFD1D
#define GDK_3270_Enter 0xFD1E
#define GDK_ISO_Lock 0xFE01
#define GDK_ISO_Level2_Latch 0xFE02
#define GDK_ISO_Level3_Shift 0xFE03
#define GDK_ISO_Level3_Latch 0xFE04
#define GDK_ISO_Level3_Lock 0xFE05
#define GDK_ISO_Group_Latch 0xFE06

```



```

#define GDK_ISO_Group_Lock      0xFE07
#define GDK_ISO_Next_Group      0xFE08
#define GDK_ISO_Next_Group_Lock 0xFE09
#define GDK_ISO_Prev_Group      0xFE0A
#define GDK_ISO_Prev_Group_Lock 0xFE0B
#define GDK_ISO_First_Group     0xFE0C
#define GDK_ISO_First_Group_Lock 0xFE0D
#define GDK_ISO_Last_Group      0xFE0E
#define GDK_ISO_Last_Group_Lock 0xFE0F
#define GDK_ISO_Left_Tab        0xFE20
#define GDK_ISO_Move_Line_Up    0xFE21
#define GDK_ISO_Move_Line_Down  0xFE22
#define GDK_ISO_Partial_Line_Up  0xFE23
#define GDK_ISO_Partial_Line_Down 0xFE24
#define GDK_ISO_Partial_Space_Left 0xFE25
#define GDK_ISO_Partial_Space_Right 0xFE26
#define GDK_ISO_Set_Margin_Left 0xFE27
#define GDK_ISO_Set_Margin_Right 0xFE28
#define GDK_ISO_Release_Margin_Left 0xFE29
#define GDK_ISO_Release_Margin_Right 0xFE2A
#define GDK_ISO_Release_Both_Margins 0xFE2B
#define GDK_ISO_Fast_Cursor_Left 0xFE2C
#define GDK_ISO_Fast_Cursor_Right 0xFE2D
#define GDK_ISO_Fast_Cursor_Up 0xFE2E
#define GDK_ISO_Fast_Cursor_Down 0xFE2F
#define GDK_ISO_Continuous_Underline 0xFE30
#define GDK_ISO_Discontinuous_Underline 0xFE31
#define GDK_ISO_Emphasize      0xFE32
#define GDK_ISO_Center_Object   0xFE33
#define GDK_ISO_Enter           0xFE34
#define GDK_dead_grave          0xFE50
#define GDK_dead_acute          0xFE51
#define GDK_dead_circumflex     0xFE52
#define GDK_dead_tilde          0xFE53
#define GDK_dead_macron          0xFE54
#define GDK_dead_breve          0xFE55
#define GDK_dead_abovedot       0xFE56
#define GDK_dead_diaeresis      0xFE57
#define GDK_dead_aoovering      0xFE58
#define GDK_dead_doubleacute     0xFE59
#define GDK_dead_caron          0xFE5A
#define GDK_dead_cedilla        0xFE5B
#define GDK_dead_ogonek         0xFE5C
#define GDK_dead_iota           0xFE5D
#define GDK_dead_voiced_sound    0xFE5E
#define GDK_dead_semivoiced_sound 0xFE5F
#define GDK_dead_belowdot       0xFE60
#define GDK_dead_hook           0xFE61
#define GDK_dead_horn           0xFE62
#define GDK_AccessX_Enable      0xFE70
#define GDK_AccessX_Feedback_Enable 0xFE71
#define GDK_RepeatKeys_Enable   0xFE72
#define GDK_SlowKeys_Enable     0xFE73
#define GDK_BounceKeys_Enable   0xFE74
#define GDK_StickyKeys_Enable   0xFE75
#define GDK_MouseKeys_Enable    0xFE76
#define GDK_MouseKeys_Accel_Enable 0xFE77
#define GDK_Overlay1_Enable     0xFE78
#define GDK_Overlay2_Enable     0xFE79
#define GDK_AudibleBell_Enable  0xFE7A
#define GDK_First_Virtual_Screen 0xFED0
#define GDK_Prev_Virtual_Screen 0xFED1
#define GDK_Next_Virtual_Screen 0xFED2
#define GDK_Last_Virtual_Screen 0xFED4
#define GDK_Terminate_Server    0xFED5
#define GDK_Pointer_Left        0xFEE0

```



```

#define GDK_Pointer_Right      0xFEE1
#define GDK_Pointer_Up 0xFEE2
#define GDK_Pointer_Down      0xFEE3
#define GDK_Pointer_UpLeft    0xFEE4
#define GDK_Pointer_UpRight   0xFEE5
#define GDK_Pointer_DownLeft  0xFEE6
#define GDK_Pointer_DownRight 0xFEE7
#define GDK_Pointer_Button_Dflt 0xFEE8
#define GDK_Pointer_Button1   0xFEE9
#define GDK_Pointer_Button2   0xFEEA
#define GDK_Pointer_Button3   0xFEEB
#define GDK_Pointer_Button4   0xFEEC
#define GDK_Pointer_Button5   0xFEED
#define GDK_Pointer_DblClick_Dflt 0xFEEE
#define GDK_Pointer_DblClick1 0xFEEF
#define GDK_Pointer_DblClick2 0xFEFO
#define GDK_Pointer_DblClick3 0xFEf1
#define GDK_Pointer_DblClick4 0xFEf2
#define GDK_Pointer_DblClick5 0xFEf3
#define GDK_Pointer_Drag_Dflt  0xFEf4
#define GDK_Pointer_Drag1     0xFEf5
#define GDK_Pointer_Drag2     0xFEf6
#define GDK_Pointer_Drag3     0xFEf7
#define GDK_Pointer_Drag4     0xFEf8
#define GDK_Pointer_EnableKeys 0xFEf9
#define GDK_Pointer_Accelerate 0xFEFA
#define GDK_Pointer_DfltBtnNext 0xFEfB
#define GDK_Pointer_DfltBtnPrev 0xFEfC
#define GDK_Pointer_Drag5     0xFEfD
#define GDK_BackSpace 0xFF08
#define GDK_Tab 0xFF09
#define GDK_Linefeed 0xFF0A
#define GDK_Clear 0xFF0B
#define GDK_Return 0xFF0D
#define GDK_Pause 0xFF13
#define GDK_Scroll_Lock 0xFF14
#define GDK_Sys_Req 0xFF15
#define GDK_Escape 0xFF1B
#define GDK_Multi_key 0xFF20
#define GDK_Kanji 0xFF21
#define GDK_Muhenkan 0xFF22
#define GDK_Henkan 0xFF23
#define GDK_Henkan_Mode 0xFF23
#define GDK_Romaji 0xFF24
#define GDK_Hiragana 0xFF25
#define GDK_Katakana 0xFF26
#define GDK_Hiragana_Katakana 0xFF27
#define GDK_Zenkaku 0xFF28
#define GDK_Hankaku 0xFF29
#define GDK_Zenkaku_Hankaku 0xFF2A
#define GDK_Touroku 0xFF2B
#define GDK_Massyo 0xFF2C
#define GDK_Kana_Lock 0xFF2D
#define GDK_Kana_Shift 0xFF2E
#define GDK_Eisu_Shift 0xFF2F
#define GDK_Eisu_toggle 0xFF30
#define GDK_Hangul 0xff31
#define GDK_Hangul_Start 0xff32
#define GDK_Hangul_End 0xff33
#define GDK_Hangul_Hanja 0xff34
#define GDK_Hangul_Jamo 0xff35
#define GDK_Hangul_Romaja 0xff36
#define GDK_Codeinput 0xFF37
#define GDK_Hangul_Codeinput 0xff37
#define GDK_Kanji_Bangou 0xFF37
#define GDK_Hangul_Jeonja 0xff38

```

```

#define GDK_Hangul_Banja      0xff39
#define GDK_Hangul_PreHanja   0xff3a
#define GDK_Hangul_PostHanja  0xff3b
#define GDK_Hangul_SingleCandidate 0xff3c
#define GDK_SingleCandidate   0xFF3C
#define GDK_Hangul_MultipleCandidate 0xff3d
#define GDK_MultipleCandidate 0xFF3D
#define GDK_Zen_Koho          0xFF3D
#define GDK_Hangul_PreviousCandidate 0xff3e
#define GDK_Mae_Koho          0xFF3E
#define GDK_PreviousCandidate 0xFF3E
#define GDK_Hangul_Special    0xff3f
#define GDK_Home              0xFF50
#define GDK_Left              0xFF51
#define GDK_Up                0xFF52
#define GDK_Right             0xFF53
#define GDK_Down              0xFF54
#define GDK_Page_Up           0xFF55
#define GDK_Prior             0xFF55
#define GDK_Next              0xFF56
#define GDK_Page_Down         0xFF56
#define GDK_End               0xFF57
#define GDK_Begin             0xFF58
#define GDK_Select            0xFF60
#define GDK_Print             0xFF61
#define GDK_Execute           0xFF62
#define GDK_Insert            0xFF63
#define GDK_Undo              0xFF65
#define GDK_Redo              0xFF66
#define GDK_Menu              0xFF67
#define GDK_Find              0xFF68
#define GDK_Cancel            0xFF69
#define GDK_Help              0xFF6A
#define GDK_Break             0xFF6B
#define GDK_Arabic_switch     0xFF7E
#define GDK_Greek_switch     0xFF7E
#define GDK_Hangul_switch     0xFF7E
#define GDK_Hebrew_switch     0xFF7E
#define GDK_ISO_Group_Shift   0xFF7E
#define GDK_Mode_switch       0xFF7E
#define GDK_kana_switch       0xFF7E
#define GDK_script_switch     0xFF7E
#define GDK_Num_Lock          0xFF7F
#define GDK_KP_Space          0xFF80
#define GDK_KP_Tab            0xFF89
#define GDK_KP_Enter          0xFF8D
#define GDK_KP_F1             0xFF91
#define GDK_KP_F2             0xFF92
#define GDK_KP_F3             0xFF93
#define GDK_KP_F4             0xFF94
#define GDK_KP_Home           0xFF95
#define GDK_KP_Left           0xFF96
#define GDK_KP_Up             0xFF97
#define GDK_KP_Right          0xFF98
#define GDK_KP_Down           0xFF99
#define GDK_KP_Page_Up        0xFF9A
#define GDK_KP_Prior          0xFF9A
#define GDK_KP_Next           0xFF9B
#define GDK_KP_Page_Down      0xFF9B
#define GDK_KP_End            0xFF9C
#define GDK_KP_Begin          0xFF9D
#define GDK_KP_Insert         0xFF9E
#define GDK_KP_Delete         0xFF9F
#define GDK_KP_Multiply        0xFFAA
#define GDK_KP_Add             0xFFAB
#define GDK_KP_Separator      0xFFAC

```

```

#define GDK_KP_Subtract 0xFFAD
#define GDK_KP_Decimal 0xFFAE
#define GDK_KP_Divide 0xFFAF
#define GDK_KP_0 0xFFB0
#define GDK_KP_1 0xFFB1
#define GDK_KP_2 0xFFB2
#define GDK_KP_3 0xFFB3
#define GDK_KP_4 0xFFB4
#define GDK_KP_5 0xFFB5
#define GDK_KP_6 0xFFB6
#define GDK_KP_7 0xFFB7
#define GDK_KP_8 0xFFB8
#define GDK_KP_9 0xFFB9
#define GDK_KP_Equal 0xFFBD
#define GDK_F1 0xFFBE
#define GDK_F2 0xFFBF
#define GDK_F3 0xFFC0
#define GDK_F4 0xFFC1
#define GDK_F5 0xFFC2
#define GDK_F6 0xFFC3
#define GDK_F7 0xFFC4
#define GDK_F8 0xFFC5
#define GDK_F9 0xFFC6
#define GDK_F10 0xFFC7
#define GDK_F11 0xFFC8
#define GDK_L1 0xFFC8
#define GDK_F12 0xFFC9
#define GDK_L2 0xFFC9
#define GDK_F13 0xFFCA
#define GDK_L3 0xFFCA
#define GDK_F14 0xFFCB
#define GDK_L4 0xFFCB
#define GDK_F15 0xFFCC
#define GDK_L5 0xFFCC
#define GDK_F16 0xFFCD
#define GDK_L6 0xFFCD
#define GDK_F17 0xFFCE
#define GDK_L7 0xFFCE
#define GDK_F18 0xFFCF
#define GDK_L8 0xFFCF
#define GDK_F19 0xFFD0
#define GDK_L9 0xFFD0
#define GDK_F20 0xFFD1
#define GDK_L10 0xFFD1
#define GDK_F21 0xFFD2
#define GDK_R1 0xFFD2
#define GDK_F22 0xFFD3
#define GDK_R2 0xFFD3
#define GDK_F23 0xFFD4
#define GDK_R3 0xFFD4
#define GDK_F24 0xFFD5
#define GDK_R4 0xFFD5
#define GDK_F25 0xFFD6
#define GDK_R5 0xFFD6
#define GDK_F26 0xFFD7
#define GDK_R6 0xFFD7
#define GDK_F27 0xFFD8
#define GDK_R7 0xFFD8
#define GDK_F28 0xFFD9
#define GDK_R8 0xFFD9
#define GDK_F29 0xFFDA
#define GDK_R9 0xFFDA
#define GDK_F30 0xFFDB
#define GDK_R10 0xFFDB
#define GDK_F31 0xFFDC
#define GDK_R11 0xFFDC

```

```

#define GDK_F32 0xFFDD
#define GDK_R12 0xFFDD
#define GDK_F33 0xFFDE
#define GDK_R13 0xFFDE
#define GDK_F34 0xFFDF
#define GDK_R14 0xFFDF
#define GDK_F35 0xFFE0
#define GDK_R15 0xFFE0
#define GDK_Shift_L 0xFFE1
#define GDK_Shift_R 0xFFE2
#define GDK_Control_L 0xFFE3
#define GDK_Control_R 0xFFE4
#define GDK_Caps_Lock 0xFFE5
#define GDK_Shift_Lock 0xFFE6
#define GDK_Meta_L 0xFFE7
#define GDK_Meta_R 0xFFE8
#define GDK_Alt_L 0xFFE9
#define GDK_Alt_R 0xFFEA
#define GDK_Super_L 0xFFEB
#define GDK_Super_R 0xFFEC
#define GDK_Hyper_L 0xFFED
#define GDK_Hyper_R 0xFFEE
#define GDK_Delete 0xFFFF
#define GDK_VoidSymbol 0xFFFFFFFF
#define GDK_Greek_IOTAadieresis GDK_Greek_IOTAadieresis

```

17.30.4 gtk-2.0/gdk/gdkx.h

```

#define GDK_WINDOW_DESTROYED(d) \
    (((GdkWindowObject*) (GDK_WINDOW (d)))->destroyed)
#define GDK_WINDOW_TYPE(d) \
    (((GdkWindowObject*) (GDK_WINDOW (d)))->window_type)
#define GDK_COLORMAP_XCOLORMAP(cmap) \
    (gdk_x11_colormap_get_xcolormap (cmap))
#define GDK_DISPLAY_XDISPLAY(display) \
    (gdk_x11_display_get_xdisplay (display))
#define GDK_SCREEN_XDISPLAY(screen) \
    (gdk_x11_display_get_xdisplay (gdk_screen_get_display
    (screen)))
#define GDK_PIXMAP_XDISPLAY(win) \
    (gdk_x11_drawable_get_xdisplay (((GdkPixmapObject *)win)-
>impl))
#define GDK_WINDOW_XDISPLAY(win) \
    (gdk_x11_drawable_get_xdisplay (((GdkWindowObject *)win)-
>impl))
#define GDK_SCREEN_XNUMBER(screen) \
    (gdk_x11_screen_get_screen_number (screen))
#define GDK_PARENT_RELATIVE_BG ((GdkPixmap *)1L)
#define GDK_NO_BG ((GdkPixmap *)2L)
#define GDK_COLORMAP_XDISPLAY(cmap) \
    (gdk_x11_colormap_get_xdisplay (cmap))
#define GDK_CURSOR_XCURSOR(cursor) \
    (gdk_x11_cursor_get_xcursor
    (cursor))
#define GDK_CURSOR_XDISPLAY(cursor) \
    (gdk_x11_cursor_get_xdisplay (cursor))
#define GDK_DRAWABLE_XDISPLAY(win) \
    (gdk_x11_drawable_get_xdisplay (win))
#define GDK_DRAWABLE_XID(win) \
    (gdk_x11_drawable_get_xid (win))
#define GDK_PIXMAP_XID(win) \
    (gdk_x11_drawable_get_xid (win))
#define GDK_WINDOW_XID(win) \
    (gdk_x11_drawable_get_xid (win))
#define GDK_WINDOW_XWINDOW(win) \
    (gdk_x11_drawable_get_xid (win))
#define GDK_GC_XDISPLAY(gc) \
    (gdk_x11_gc_get_xdisplay (gc))
#define GDK_GC_XGC(gc) \
    (gdk_x11_gc_get_xgc (gc))
#define GDK_ROOT_WINDOW() \
    (gdk_x11_get_default_root_xwindow
    ())

```

```

#define GDK_IMAGE_XDISPLAY(image)      (gdk_x11_image_get_xdisplay
(image))
#define GDK_IMAGE_XIMAGE(image) (gdk_x11_image_get_ximage (image))
#define GDK_SCREEN_XSCREEN(screen)      (gdk_x11_screen_get_xscreen
(screen))
#define GDK_VISUAL_XVISUAL(visual)      (gdk_x11_visual_get_xvisual
(visual))
#define GDK_DISPLAY()    gdk_display

extern Display *gdk_display;
extern gboolean gdk_net_wm_supports(GdkAtom property);
extern void gdk_synthesize_window_state(GdkWindow * window,
                                         GdkWindowState unset_flags,
                                         GdkWindowState set_flags);
extern void gdk_window_destroy_notify(GdkWindow * window);
extern Atom gdk_x11_atom_to_xatom(GdkAtom atom);
extern Atom gdk_x11_atom_to_xatom_for_display(GdkDisplay * display,
                                              GdkAtom atom);
extern GdkColormap *gdk_x11_colormap_foreign_new(GdkVisual *
visual,
                                                  Colormap xcolormap);
extern Colormap gdk_x11_colormap_get_xcolormap(GdkColormap *
colormap);
extern Display *gdk_x11_colormap_get_xdisplay(GdkColormap *
colormap);
extern Cursor gdk_x11_cursor_get_xcursor(GdkCursor * cursor);
extern Display *gdk_x11_cursor_get_xdisplay(GdkCursor * cursor);
extern guint32 gdk_x11_display_get_user_time(GdkDisplay * display);
extern Display *gdk_x11_display_get_xdisplay(GdkDisplay * display);
extern void gdk_x11_display_grab(GdkDisplay * display);
extern void gdk_x11_display_set_cursor_theme(GdkDisplay * display,
                                              const gchar * theme,
                                              const gint size);
extern void gdk_x11_display_ungrab(GdkDisplay * display);
extern Display *gdk_x11_drawable_get_xdisplay(GdkDrawable *
drawable);
extern XID gdk_x11_drawable_get_xid(GdkDrawable * drawable);
extern Display *gdk_x11_gc_get_xdisplay(GdkGC * gc);
extern GC gdk_x11_gc_get_xgc(GdkGC * gc);
extern Window gdk_x11_get_default_root_xwindow(void);
extern gint gdk_x11_get_default_screen(void);
extern Display *gdk_x11_get_default_xdisplay(void);
extern guint32 gdk_x11_get_server_time(GdkWindow * window);
extern Atom gdk_x11_get_xatom_by_name(const gchar * atom_name);
extern Atom gdk_x11_get_xatom_by_name_for_display(GdkDisplay *
display,
                                                  const gchar * atom_name);
extern const gchar *gdk_x11_get_xatom_name(Atom xatom);
extern const gchar *gdk_x11_get_xatom_name_for_display(GdkDisplay
*
display,
                                                  Atom xatom);
extern void gdk_x11_grab_server(void);
extern Display *gdk_x11_image_get_xdisplay(GdkImage * image);
extern XImage *gdk_x11_image_get_ximage(GdkImage * image);
extern GdkDisplay *gdk_x11_lookup_xdisplay(Display * xdisplay);
extern void gdk_x11_register_standard_event_type(GdkDisplay *
display,
                                                  gint event_base,
                                                  gint n_events);
extern int gdk_x11_screen_get_screen_number(GdkScreen * screen);
extern const char
*gdk_x11_screen_get_window_manager_name(GdkScreen *
screen);
extern Screen *gdk_x11_screen_get_xscreen(GdkScreen * screen);
extern GdkVisual *gdk_x11_screen_lookup_visual(GdkScreen * screen,

```

```

VisualID xvisualid);
extern gboolean gdk_x11_screen_supports_net_wm_hint(GdkScreen *
screen,
GdkAtom property);
extern void gdk_x11_ungrab_server(void);
extern Visual *gdk_x11_visual_get_xvisual(GdkVisual * visual);
extern void gdk_x11_window_move_to_current_desktop(GdkWindow *
window);
extern void gdk_x11_window_set_user_time(GdkWindow * window,
guint32 timestamp);
extern GdkAtom gdk_x11_xatom_to_atom(Atom xatom);
extern GdkAtom gdk_x11_xatom_to_atom_for_display(GdkDisplay *
display,
Atom xatom);
extern gpointer gdk_xid_table_lookup(XID xid);
extern gpointer gdk_xid_table_lookup_for_display(GdkDisplay *
display,
XID xid);
extern GdkVisual *gdkx_visual_get(VisualID xvisualid);

```

17.31 Interfaces for libgtk-x11-2.0

Table 17-122 defines the library name and shared object name for the libgtk-x11-2.0 library

Table 17-122 libgtk-x11-2.0 Definition

Library:	libgtk-x11-2.0
SONAME:	libgtk-x11-2.0.so.0

The behavior of the interfaces in this library is specified by the following specifications:

[Gobject 2.32] Gobject 2.32 Reference Manual
[Gtk 2.10] Gtk+ 2.10.14 Reference Manual
[LSB] This Specification

17.31.1 GTK main Widgets library

17.31.1.1 Interfaces for GTK main Widgets library

An LSB conforming implementation shall provide the generic functions for GTK main Widgets library specified in Table 17-123, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-123 libgtk-x11-2.0 - GTK main Widgets library Function Interfaces

gtk_about_dialog_get_artists [Gtk 2.10]	gtk_about_dialog_get_authors [Gtk 2.10]
gtk_about_dialog_get_comments [Gtk 2.10]	gtk_about_dialog_get_copyright [Gtk 2.10]
gtk_about_dialog_get_documenters [Gtk 2.10]	gtk_about_dialog_get_license [Gtk 2.10]
gtk_about_dialog_get_logo [Gtk 2.10]	gtk_about_dialog_get_logo_icon_name [Gtk 2.10]
gtk_about_dialog_get_name [Gtk 2.10]	gtk_about_dialog_get_translator_credits [Gtk 2.10]

gtk_about_dialog_get_type [Gobject 2.32]	gtk_about_dialog_get_version [Gtk 2.10]
gtk_about_dialog_get_website [Gtk 2.10]	gtk_about_dialog_get_website_label [Gtk 2.10]
gtk_about_dialog_get_wrap_license [Gtk 2.10]	gtk_about_dialog_new [Gtk 2.10]
gtk_about_dialog_set_artists [Gtk 2.10]	gtk_about_dialog_set_authors [Gtk 2.10]
gtk_about_dialog_set_comments [Gtk 2.10]	gtk_about_dialog_set_copyright [Gtk 2.10]
gtk_about_dialog_set_documenters [Gtk 2.10]	gtk_about_dialog_set_email_hook [Gtk 2.10]
gtk_about_dialog_set_license [Gtk 2.10]	gtk_about_dialog_set_logo [Gtk 2.10]
gtk_about_dialog_set_logo_icon_name [Gtk 2.10]	gtk_about_dialog_set_name [Gtk 2.10]
gtk_about_dialog_set_translator_credits [Gtk 2.10]	gtk_about_dialog_set_url_hook [Gtk 2.10]
gtk_about_dialog_set_version [Gtk 2.10]	gtk_about_dialog_set_website [Gtk 2.10]
gtk_about_dialog_set_website_label [Gtk 2.10]	gtk_about_dialog_set_wrap_license [Gtk 2.10]
gtk_accel_flags_get_type [Gobject 2.32]	gtk_accel_group_activate [Gtk 2.10]
gtk_accel_group_connect [Gtk 2.10]	gtk_accel_group_connect_by_path [Gtk 2.10]
gtk_accel_group_disconnect [Gtk 2.10]	gtk_accel_group_disconnect_key [Gtk 2.10]
gtk_accel_group_find [Gtk 2.10]	gtk_accel_group_from_accel_closure [Gtk 2.10]
gtk_accel_group_get_type [Gobject 2.32]	gtk_accel_group_lock [Gtk 2.10]
gtk_accel_group_new [Gtk 2.10]	gtk_accel_group_query [Gtk 2.10]
gtk_accel_group_unlock [Gtk 2.10]	gtk_accel_groups_activate [Gtk 2.10]
gtk_accel_groups_from_object [Gtk 2.10]	gtk_accel_label_get_accel_widget [Gtk 2.10]
gtk_accel_label_get_accel_width [Gtk 2.10]	gtk_accel_label_get_type [Gobject 2.32]
gtk_accel_label_new [Gtk 2.10]	gtk_accel_label_refetch [Gtk 2.10]
gtk_accel_label_set_accel_closure [Gtk 2.10]	gtk_accel_label_set_accel_widget [Gtk 2.10]

gtk_accel_map_add_entry [Gtk 2.10]	gtk_accel_map_add_filter [Gtk 2.10]
gtk_accel_map_change_entry [Gtk 2.10]	gtk_accel_map_foreach [Gtk 2.10]
gtk_accel_map_foreach_unfiltered [Gtk 2.10]	gtk_accel_map_get [Gtk 2.10]
gtk_accel_map_get_type [Gobject 2.32]	gtk_accel_map_load [Gtk 2.10]
gtk_accel_map_load_fd [Gtk 2.10]	gtk_accel_map_load_scanner [Gtk 2.10]
gtk_accel_map_lock_path [Gtk 2.10]	gtk_accel_map_lookup_entry [Gtk 2.10]
gtk_accel_map_save [Gtk 2.10]	gtk_accel_map_save_fd [Gtk 2.10]
gtk_accel_map_unlock_path [Gtk 2.10]	gtk_accelerator_get_default_mod_mask [Gtk 2.10]
gtk_accelerator_get_label [Gtk 2.10]	gtk_accelerator_name [Gtk 2.10]
gtk_accelerator_parse [Gtk 2.10]	gtk_accelerator_set_default_mod_mask [Gtk 2.10]
gtk_accelerator_valid [Gtk 2.10]	gtk_accessible_connect_widget_destroyed [Gtk 2.10]
gtk_accessible_get_type [Gobject 2.32]	gtk_action_activate [Gtk 2.10]
gtk_action_block_activate_from [Gtk 2.10]	gtk_action_connect_accelerator [Gtk 2.10]
gtk_action_connect_proxy [Gtk 2.10]	gtk_action_create_icon [Gtk 2.10]
gtk_action_create_menu_item [Gtk 2.10]	gtk_action_create_tool_item [Gtk 2.10]
gtk_action_disconnect_accelerator [Gtk 2.10]	gtk_action_disconnect_proxy [Gtk 2.10]
gtk_action_get_accel_closure [Gtk 2.10]	gtk_action_get_accel_path [Gtk 2.10]
gtk_action_get_name [Gtk 2.10]	gtk_action_get_proxies [Gtk 2.10]
gtk_action_get_sensitive [Gtk 2.10]	gtk_action_get_type [Gobject 2.32]
gtk_action_get_visible [Gtk 2.10]	gtk_action_group_add_action [Gtk 2.10]
gtk_action_group_add_action_with_accel [Gtk 2.10]	gtk_action_group_add_actions [Gtk 2.10]
gtk_action_group_add_actions_full [Gtk 2.10]	gtk_action_group_add_radio_actions [Gtk 2.10]
gtk_action_group_add_radio_actions_full [Gtk 2.10]	gtk_action_group_add_toggle_actions [Gtk 2.10]

gtk_action_group_add_toggle_actions_full [Gtk 2.10]	gtk_action_group_get_action [Gtk 2.10]
gtk_action_group_get_name [Gtk 2.10]	gtk_action_group_get_sensitive [Gtk 2.10]
gtk_action_group_get_type [Gobject 2.32]	gtk_action_group_get_visible [Gtk 2.10]
gtk_action_group_list_actions [Gtk 2.10]	gtk_action_group_new [Gtk 2.10]
gtk_action_group_remove_action [Gtk 2.10]	gtk_action_group_set_sensitive [Gtk 2.10]
gtk_action_group_set_translate_func [Gtk 2.10]	gtk_action_group_set_translation_domain [Gtk 2.10]
gtk_action_group_set_visible [Gtk 2.10]	gtk_action_group_translate_string [Gtk 2.10]
gtk_action_is_sensitive [Gtk 2.10]	gtk_action_is_visible [Gtk 2.10]
gtk_action_new [Gtk 2.10]	gtk_action_set_accel_group [Gtk 2.10]
gtk_action_set_accel_path [Gtk 2.10]	gtk_action_set_sensitive [Gtk 2.10]
gtk_action_set_visible [Gtk 2.10]	gtk_action_unblock_activate_from [Gtk 2.10]
gtk_adjustment_changed [Gtk 2.10]	gtk_adjustment_clamp_page [Gtk 2.10]
gtk_adjustment_get_type [Gobject 2.32]	gtk_adjustment_get_value [Gtk 2.10]
gtk_adjustment_new [Gtk 2.10]	gtk_adjustment_set_value [Gtk 2.10]
gtk_adjustment_value_changed [Gtk 2.10]	gtk_alignment_get_padding [Gtk 2.10]
gtk_alignment_get_type [Gobject 2.32]	gtk_alignment_new [Gtk 2.10]
gtk_alignment_set [Gtk 2.10]	gtk_alignment_set_padding [Gtk 2.10]
gtk_alternative_dialog_button_order [Gtk 2.10]	gtk_anchor_type_get_type [Gobject 2.32]
gtk_arg_flags_get_type [Gobject 2.32]	gtk_arrow_get_type [Gobject 2.32]
gtk_arrow_new [Gtk 2.10]	gtk_arrow_set [Gtk 2.10]
gtk_arrow_type_get_type [Gobject 2.32]	gtk_aspect_frame_get_type [Gobject 2.32]
gtk_aspect_frame_new [Gtk 2.10]	gtk_aspect_frame_set [Gtk 2.10]
gtk_assistant_add_action_widget [Gtk 2.10]	gtk_assistant_append_page [Gtk 2.10]

gtk_assistant_get_current_page [Gtk 2.10]	gtk_assistant_get_n_pages [Gtk 2.10]
gtk_assistant_get_nth_page [Gtk 2.10]	gtk_assistant_get_page_complete [Gtk 2.10]
gtk_assistant_get_page_header_image [Gtk 2.10]	gtk_assistant_get_page_side_image [Gtk 2.10]
gtk_assistant_get_page_title [Gtk 2.10]	gtk_assistant_get_page_type [Gtk 2.10]
gtk_assistant_get_type [Gtk 2.10]	gtk_assistant_insert_page [Gtk 2.10]
gtk_assistant_new [Gtk 2.10]	gtk_assistant_page_type_get_type [Gtk 2.10]
gtk_assistant_prepend_page [Gtk 2.10]	gtk_assistant_remove_action_widget [Gtk 2.10]
gtk_assistant_set_current_page [Gtk 2.10]	gtk_assistant_set_forward_page_function [Gtk 2.10]
gtk_assistant_set_page_complete [Gtk 2.10]	gtk_assistant_set_page_header_image [Gtk 2.10]
gtk_assistant_set_page_side_image [Gtk 2.10]	gtk_assistant_set_page_title [Gtk 2.10]
gtk_assistant_set_page_type [Gtk 2.10]	gtk_assistant_update_buttons_state [Gtk 2.10]
gtk_attach_options_get_type [GObject 2.32]	gtk_bin_get_child [Gtk 2.10]
gtk_bin_get_type [GObject 2.32]	gtk_binding_entry_add_signal [Gtk 2.10]
gtk_binding_entry_add_signal [Gtk 2.10]	gtk_binding_entry_clear [Gtk 2.10]
gtk_binding_entry_remove [Gtk 2.10]	gtk_binding_parse_binding [Gtk 2.10]
gtk_binding_set_activate [Gtk 2.10]	gtk_binding_set_add_path [Gtk 2.10]
gtk_binding_set_by_class [Gtk 2.10]	gtk_binding_set_find [Gtk 2.10]
gtk_binding_set_new [Gtk 2.10]	gtk_bindings_activate [Gtk 2.10]
gtk_bindings_activate_event [Gtk 2.10]	gtk_border_copy [Gtk 2.10]
gtk_border_free [Gtk 2.10]	gtk_border_get_type [GObject 2.32]
gtk_box_get_homogeneous [Gtk 2.10]	gtk_box_get_spacing [Gtk 2.10]
gtk_box_get_type [GObject 2.32]	gtk_box_pack_end [Gtk 2.10]
gtk_box_pack_end_defaults [Gtk 2.10]	gtk_box_pack_start [Gtk 2.10]

gtk_box_pack_start_defaults [Gtk 2.10]	gtk_box_query_child_packing [Gtk 2.10]
gtk_box_reorder_child [Gtk 2.10]	gtk_box_set_child_packing [Gtk 2.10]
gtk_box_set_homogeneous [Gtk 2.10]	gtk_box_set_spacing [Gtk 2.10]
gtk_button_box_get_child_secondary [Gtk 2.10]	gtk_button_box_get_layout [Gtk 2.10]
gtk_button_box_get_type [Gobject 2.32]	gtk_button_box_set_child_secondary [Gtk 2.10]
gtk_button_box_set_layout [Gtk 2.10]	gtk_button_box_style_get_type [Gobject 2.32]
gtk_button_clicked [Gtk 2.10]	gtk_button_enter [Gtk 2.10]
gtk_button_get_alignment [Gtk 2.10]	gtk_button_get_focus_on_click [Gtk 2.10]
gtk_button_get_image [LSB]	gtk_button_get_image_position [Gtk 2.10]
gtk_button_get_label [Gtk 2.10]	gtk_button_get_relief [Gtk 2.10]
gtk_button_get_type [Gobject 2.32]	gtk_button_get_use_stock [Gtk 2.10]
gtk_button_get_use_underline [Gtk 2.10]	gtk_button_leave [Gtk 2.10]
gtk_button_new [Gtk 2.10]	gtk_button_new_from_stock [Gtk 2.10]
gtk_button_new_with_label [Gtk 2.10]	gtk_button_new_with_mnemonic [Gtk 2.10]
gtk_button_pressed [Gtk 2.10]	gtk_button_released [Gtk 2.10]
gtk_button_set_alignment [Gtk 2.10]	gtk_button_set_focus_on_click [Gtk 2.10]
gtk_button_set_image [LSB]	gtk_button_set_image_position [Gtk 2.10]
gtk_button_set_label [Gtk 2.10]	gtk_button_set_relief [Gtk 2.10]
gtk_button_set_use_stock [Gtk 2.10]	gtk_button_set_use_underline [Gtk 2.10]
gtk_buttons_type_get_type [Gobject 2.32]	gtk_calendar_clear_marks [Gtk 2.10]
gtk_calendar_display_options_get_type [Gobject 2.32]	gtk_calendar_get_date [Gtk 2.10]
gtk_calendar_get_display_options [Gtk 2.10]	gtk_calendar_get_type [Gobject 2.32]
gtk_calendar_mark_day [Gtk 2.10]	gtk_calendar_new [Gtk 2.10]
gtk_calendar_select_day [Gtk 2.10]	gtk_calendar_select_month [Gtk 2.10]

gtk_calendar_set_display_options [Gtk 2.10]	gtk_calendar_unmark_day [Gtk 2.10]
gtk_cell_editable_editing_done [Gtk 2.10]	gtk_cell_editable_get_type [Gobject 2.32]
gtk_cell_editable_remove_widget [Gtk 2.10]	gtk_cell_editable_start_editing [Gtk 2.10]
gtk_cell_layout_add_attribute [Gtk 2.10]	gtk_cell_layout_clear [Gtk 2.10]
gtk_cell_layout_clear_attributes [Gtk 2.10]	gtk_cell_layout_get_type [Gobject 2.32]
gtk_cell_layout_pack_end [Gtk 2.10]	gtk_cell_layout_pack_start [Gtk 2.10]
gtk_cell_layout_reorder [Gtk 2.10]	gtk_cell_layout_set_attributes [Gtk 2.10]
gtk_cell_layout_set_cell_data_func [Gtk 2.10]	gtk_cell_renderer_accel_get_type [Gtk 2.10]
gtk_cell_renderer_accel_mode_get_type [Gtk 2.10]	gtk_cell_renderer_accel_new [Gtk 2.10]
gtk_cell_renderer_activate [Gtk 2.10]	gtk_cell_renderer_combo_get_type [Gobject 2.32]
gtk_cell_renderer_combo_new [Gtk 2.10]	gtk_cell_renderer_get_fixed_size [Gtk 2.10]
gtk_cell_renderer_get_size [Gtk 2.10]	gtk_cell_renderer_get_type [Gobject 2.32]
gtk_cell_renderer_mode_get_type [Gobject 2.32]	gtk_cell_renderer_pixbuf_get_type [Gobject 2.32]
gtk_cell_renderer_pixbuf_new [Gtk 2.10]	gtk_cell_renderer_progress_get_type [Gobject 2.32]
gtk_cell_renderer_progress_new [Gtk 2.10]	gtk_cell_renderer_render [Gtk 2.10]
gtk_cell_renderer_set_fixed_size [Gtk 2.10]	gtk_cell_renderer_spin_get_type [Gtk 2.10]
gtk_cell_renderer_spin_new [Gtk 2.10]	gtk_cell_renderer_start_editing [Gtk 2.10]
gtk_cell_renderer_state_get_type [Gobject 2.32]	gtk_cell_renderer_stop_editing [Gtk 2.10]
gtk_cell_renderer_text_get_type [Gobject 2.32]	gtk_cell_renderer_text_new [Gtk 2.10]
gtk_cell_renderer_text_set_fixed_height_from_font [Gtk 2.10]	gtk_cell_renderer_toggle_get_active [Gtk 2.10]
gtk_cell_renderer_toggle_get_radio [Gtk 2.10]	gtk_cell_renderer_toggle_get_type [Gobject 2.32]

gtk_cell_renderer_toggle_new [Gtk 2.10]	gtk_cell_renderer_toggle_set_active [Gtk 2.10]
gtk_cell_renderer_toggle_set_radio [Gtk 2.10]	gtk_cell_view_get_cell_renderers [Gtk 2.10]
gtk_cell_view_get_displayed_row [Gtk 2.10]	gtk_cell_view_get_size_of_row [Gtk 2.10]
gtk_cell_view_get_type [Gobject 2.32]	gtk_cell_view_new [Gtk 2.10]
gtk_cell_view_new_with_markup [Gtk 2.10]	gtk_cell_view_new_with_pixbuf [Gtk 2.10]
gtk_cell_view_new_with_text [Gtk 2.10]	gtk_cell_view_set_background_color [Gtk 2.10]
gtk_cell_view_set_displayed_row [Gtk 2.10]	gtk_cell_view_set_model [Gtk 2.10]
gtk_check_button_get_type [Gobject 2.32]	gtk_check_button_new [Gtk 2.10]
gtk_check_button_new_with_label [Gtk 2.10]	gtk_check_button_new_with_mnemonic [Gtk 2.10]
gtk_check_menu_item_get_active [Gtk 2.10]	gtk_check_menu_item_get_draw_as_radio [Gtk 2.10]
gtk_check_menu_item_get_inconsistent [Gtk 2.10]	gtk_check_menu_item_get_type [Gobject 2.32]
gtk_check_menu_item_new [Gtk 2.10]	gtk_check_menu_item_new_with_label [Gtk 2.10]
gtk_check_menu_item_new_with_mnemonic [Gtk 2.10]	gtk_check_menu_item_set_active [Gtk 2.10]
gtk_check_menu_item_set_draw_as_radio [Gtk 2.10]	gtk_check_menu_item_set_inconsistent [Gtk 2.10]
gtk_check_menu_item_toggled [Gtk 2.10]	gtk_check_version [Gtk 2.10]
gtk_clipboard_clear [Gtk 2.10]	gtk_clipboard_get [Gtk 2.10]
gtk_clipboard_get_display [Gtk 2.10]	gtk_clipboard_get_for_display [Gtk 2.10]
gtk_clipboard_get_owner [Gtk 2.10]	gtk_clipboard_get_type [Gobject 2.32]
gtk_clipboard_request_contents [Gtk 2.10]	gtk_clipboard_request_image [Gtk 2.10]
gtk_clipboard_request_rich_text [Gtk 2.10]	gtk_clipboard_request_targets [Gtk 2.10]
gtk_clipboard_request_text [Gtk 2.10]	gtk_clipboard_set_can_store [Gtk 2.10]
gtk_clipboard_set_image [Gtk 2.10]	gtk_clipboard_set_text [Gtk 2.10]

gtk_clipboard_set_with_data [Gtk 2.10]	gtk_clipboard_set_with_owner [Gtk 2.10]
gtk_clipboard_store [Gtk 2.10]	gtk_clipboard_wait_for_contents [Gtk 2.10]
gtk_clipboard_wait_for_image [Gtk 2.10]	gtk_clipboard_wait_for_rich_text [Gtk 2.10]
gtk_clipboard_wait_for_targets [Gtk 2.10]	gtk_clipboard_wait_for_text [Gtk 2.10]
gtk_clipboard_wait_is_image_available [Gtk 2.10]	gtk_clipboard_wait_is_rich_text_available [Gtk 2.10]
gtk_clipboard_wait_is_target_available [Gtk 2.10]	gtk_clipboard_wait_is_text_available [Gtk 2.10]
gtk_color_button_get_alpha [Gtk 2.10]	gtk_color_button_get_color [Gtk 2.10]
gtk_color_button_get_title [Gtk 2.10]	gtk_color_button_get_type [Gobject 2.32]
gtk_color_button_get_use_alpha [Gtk 2.10]	gtk_color_button_new [Gtk 2.10]
gtk_color_button_new_with_color [Gtk 2.10]	gtk_color_button_set_alpha [Gtk 2.10]
gtk_color_button_set_color [Gtk 2.10]	gtk_color_button_set_title [Gtk 2.10]
gtk_color_button_set_use_alpha [Gtk 2.10]	gtk_color_selection_dialog_get_type [Gobject 2.32]
gtk_color_selection_dialog_new [Gtk 2.10]	gtk_color_selection_get_current_alpha [Gtk 2.10]
gtk_color_selection_get_current_color [Gtk 2.10]	gtk_color_selection_get_has_opacity_control [Gtk 2.10]
gtk_color_selection_get_has_palette [Gtk 2.10]	gtk_color_selection_get_previous_alpha [Gtk 2.10]
gtk_color_selection_get_previous_color [Gtk 2.10]	gtk_color_selection_get_type [Gobject 2.32]
gtk_color_selection_is_adjusting [Gtk 2.10]	gtk_color_selection_new [Gtk 2.10]
gtk_color_selection_palette_from_string [Gtk 2.10]	gtk_color_selection_palette_to_string [Gtk 2.10]
gtk_color_selection_set_change_palette_with_screen_hook [Gtk 2.10]	gtk_color_selection_set_current_alpha [Gtk 2.10]
gtk_color_selection_set_current_color [Gtk 2.10]	gtk_color_selection_set_has_opacity_control [Gtk 2.10]
gtk_color_selection_set_has_palette [Gtk 2.10]	gtk_color_selection_set_previous_alpha [Gtk 2.10]

gtk_color_selection_set_previous_color [Gtk 2.10]	gtk_combo_box_append_text [Gtk 2.10]
gtk_combo_box_entry_get_text_column [Gtk 2.10]	gtk_combo_box_entry_get_type [GObject 2.32]
gtk_combo_box_entry_new [Gtk 2.10]	gtk_combo_box_entry_new_text [Gtk 2.10]
gtk_combo_box_entry_new_with_model [Gtk 2.10]	gtk_combo_box_entry_set_text_column [Gtk 2.10]
gtk_combo_box_get_active [Gtk 2.10]	gtk_combo_box_get_active_iter [Gtk 2.10]
gtk_combo_box_get_active_text [Gtk 2.10]	gtk_combo_box_get_add_tearoffs [Gtk 2.10]
gtk_combo_box_get_column_span_column [Gtk 2.10]	gtk_combo_box_get_focus_on_click [Gtk 2.10]
gtk_combo_box_get_model [Gtk 2.10]	gtk_combo_box_get_popup_accessible [Gtk 2.10]
gtk_combo_box_get_row_separator_func [Gtk 2.10]	gtk_combo_box_get_row_span_column [Gtk 2.10]
gtk_combo_box_get_title [Gtk 2.10]	gtk_combo_box_get_type [GObject 2.32]
gtk_combo_box_get_wrap_width [Gtk 2.10]	gtk_combo_box_insert_text [Gtk 2.10]
gtk_combo_box_new [Gtk 2.10]	gtk_combo_box_new_text [Gtk 2.10]
gtk_combo_box_new_with_model [Gtk 2.10]	gtk_combo_box_popdown [Gtk 2.10]
gtk_combo_box_popup [Gtk 2.10]	gtk_combo_box_prepend_text [Gtk 2.10]
gtk_combo_box_remove_text [Gtk 2.10]	gtk_combo_box_set_active [Gtk 2.10]
gtk_combo_box_set_active_iter [Gtk 2.10]	gtk_combo_box_set_add_tearoffs [Gtk 2.10]
gtk_combo_box_set_column_span_column [Gtk 2.10]	gtk_combo_box_set_focus_on_click [Gtk 2.10]
gtk_combo_box_set_model [Gtk 2.10]	gtk_combo_box_set_row_separator_func [Gtk 2.10]
gtk_combo_box_set_row_span_column [Gtk 2.10]	gtk_combo_box_set_title [Gtk 2.10]
gtk_combo_box_set_wrap_width [Gtk 2.10]	gtk_combo_get_type [GObject 2.32]
gtk_container_add [Gtk 2.10]	gtk_container_add_with_properties [Gtk 2.10]

gtk_container_check_resize [Gtk 2.10]	gtk_container_child_get [Gtk 2.10]
gtk_container_child_get_property [Gtk 2.10]	gtk_container_child_get_valist [Gtk 2.10]
gtk_container_child_set [Gtk 2.10]	gtk_container_child_set_property [Gtk 2.10]
gtk_container_child_set_valist [Gtk 2.10]	gtk_container_child_type [Gtk 2.10]
gtk_container_class_find_child_property [Gtk 2.10]	gtk_container_class_install_child_property [Gtk 2.10]
gtk_container_class_list_child_properties [Gtk 2.10]	gtk_container_forall [Gtk 2.10]
gtk_container_foreach [Gtk 2.10]	gtk_container_get_border_width [Gtk 2.10]
gtk_container_get_children [Gtk 2.10]	gtk_container_get_focus_chain [Gtk 2.10]
gtk_container_get_focus_hadjustment [Gtk 2.10]	gtk_container_get_focus_vadjustment [Gtk 2.10]
gtk_container_get_resize_mode [Gtk 2.10]	gtk_container_get_type [Gobject 2.32]
gtk_container_propagate_expose [Gtk 2.10]	gtk_container_remove [Gtk 2.10]
gtk_container_resize_children [Gtk 2.10]	gtk_container_set_border_width [Gtk 2.10]
gtk_container_set_focus_chain [Gtk 2.10]	gtk_container_set_focus_child [Gtk 2.10]
gtk_container_set_focus_hadjustment [Gtk 2.10]	gtk_container_set_focus_vadjustment [Gtk 2.10]
gtk_container_set_reallocate_redraws [Gtk 2.10]	gtk_container_set_resize_mode [Gtk 2.10]
gtk_container_unset_focus_chain [Gtk 2.10]	gtk_corner_type_get_type [Gobject 2.32]
gtk_curve_get_type [Gobject 2.32]	gtk_curve_get_vector [Gtk 2.10]
gtk_curve_new [Gtk 2.10]	gtk_curve_reset [Gtk 2.10]
gtk_curve_set_curve_type [Gtk 2.10]	gtk_curve_set_gamma [Gtk 2.10]
gtk_curve_set_range [Gtk 2.10]	gtk_curve_set_vector [Gtk 2.10]
gtk_curve_type_get_type [Gobject 2.32]	gtk_debug_flag_get_type [Gobject 2.32]
gtk_delete_type_get_type [Gobject 2.32]	gtk_dest_defaults_get_type [Gobject 2.32]

gtk_dialog_add_action_widget [Gtk 2.10]	gtk_dialog_add_button [Gtk 2.10]
gtk_dialog_add_buttons [Gtk 2.10]	gtk_dialog_flags_get_type [Gobject 2.32]
gtk_dialog_get_has_separator [Gtk 2.10]	gtk_dialog_get_response_for_widget [Gtk 2.10]
gtk_dialog_get_type [Gobject 2.32]	gtk_dialog_new [Gtk 2.10]
gtk_dialog_new_with_buttons [Gtk 2.10]	gtk_dialog_response [Gtk 2.10]
gtk_dialog_run [Gtk 2.10]	gtk_dialog_set_alternative_button_order [Gtk 2.10]
gtk_dialog_set_alternative_button_order_from_array [Gtk 2.10]	gtk_dialog_set_default_response [Gtk 2.10]
gtk_dialog_set_has_separator [Gtk 2.10]	gtk_dialog_set_response_sensitive [Gtk 2.10]
gtk_direction_type_get_type [Gobject 2.32]	gtk_disable_setlocale [Gtk 2.10]
gtk_drag_begin [Gtk 2.10]	gtk_drag_check_threshold [Gtk 2.10]
gtk_drag_dest_add_image_targets [Gtk 2.10]	gtk_drag_dest_add_text_targets [Gtk 2.10]
gtk_drag_dest_add_uri_targets [Gtk 2.10]	gtk_drag_dest_find_target [Gtk 2.10]
gtk_drag_dest_get_target_list [Gtk 2.10]	gtk_drag_dest_get_track_motion [Gtk 2.10]
gtk_drag_dest_set [Gtk 2.10]	gtk_drag_dest_set_proxy [Gtk 2.10]
gtk_drag_dest_set_target_list [Gtk 2.10]	gtk_drag_dest_set_track_motion [Gtk 2.10]
gtk_drag_dest_unset [Gtk 2.10]	gtk_drag_finish [Gtk 2.10]
gtk_drag_get_data [Gtk 2.10]	gtk_drag_get_source_widget [Gtk 2.10]
gtk_drag_highlight [Gtk 2.10]	gtk_drag_set_icon_default [Gtk 2.10]
gtk_drag_set_icon_name [Gtk 2.10]	gtk_drag_set_icon_pixbuf [Gtk 2.10]
gtk_drag_set_icon_pixmap [Gtk 2.10]	gtk_drag_set_icon_stock [Gtk 2.10]
gtk_drag_set_icon_widget [Gtk 2.10]	gtk_drag_source_add_image_targets [Gtk 2.10]
gtk_drag_source_add_text_targets [Gtk 2.10]	gtk_drag_source_add_uri_targets [Gtk 2.10]
gtk_drag_source_get_target_list [Gtk 2.10]	gtk_drag_source_set [Gtk 2.10]
gtk_drag_source_set_icon [Gtk 2.10]	gtk_drag_source_set_icon_name [Gtk 2.10]

gtk_drag_source_set_icon_pixbuf [Gtk 2.10]	gtk_drag_source_set_icon_stock [Gtk 2.10]
gtk_drag_source_set_target_list [Gtk 2.10]	gtk_drag_source_unset [Gtk 2.10]
gtk_drag_unhighlight [Gtk 2.10]	gtk_draw_insertion_cursor [Gtk 2.10]
gtk_drawing_area_get_type [Gobject 2.32]	gtk_drawing_area_new [Gtk 2.10]
gtk_editable_copy_clipboard [Gtk 2.10]	gtk_editable_cut_clipboard [Gtk 2.10]
gtk_editable_delete_selection [Gtk 2.10]	gtk_editable_delete_text [Gtk 2.10]
gtk_editable_get_chars [Gtk 2.10]	gtk_editable_get_editable [Gtk 2.10]
gtk_editable_get_position [Gtk 2.10]	gtk_editable_get_selection_bounds [Gtk 2.10]
gtk_editable_get_type [Gobject 2.32]	gtk_editable_insert_text [Gtk 2.10]
gtk_editable_paste_clipboard [Gtk 2.10]	gtk_editable_select_region [Gtk 2.10]
gtk_editable_set_editable [Gtk 2.10]	gtk_editable_set_position [Gtk 2.10]
gtk_entry_completion_complete [Gtk 2.10]	gtk_entry_completion_delete_action [Gtk 2.10]
gtk_entry_completion_get_entry [Gtk 2.10]	gtk_entry_completion_get_inline_completion [Gtk 2.10]
gtk_entry_completion_get_minimum_key_length [Gtk 2.10]	gtk_entry_completion_get_model [Gtk 2.10]
gtk_entry_completion_get_popup_completion [Gtk 2.10]	gtk_entry_completion_get_popup_set_width [Gtk 2.10]
gtk_entry_completion_get_popup_single_match [Gtk 2.10]	gtk_entry_completion_get_text_column [Gtk 2.10]
gtk_entry_completion_get_type [Gobject 2.32]	gtk_entry_completion_insert_action_markup [Gtk 2.10]
gtk_entry_completion_insert_action_text [Gtk 2.10]	gtk_entry_completion_insert_prefix [Gtk 2.10]
gtk_entry_completion_new [Gtk 2.10]	gtk_entry_completion_set_inline_completion [Gtk 2.10]
gtk_entry_completion_set_match_function [Gtk 2.10]	gtk_entry_completion_set_minimum_key_length [Gtk 2.10]
gtk_entry_completion_set_model [Gtk 2.10]	gtk_entry_completion_set_popup_completion [Gtk 2.10]
gtk_entry_completion_set_popup_set_width [Gtk 2.10]	gtk_entry_completion_set_popup_single_match [Gtk 2.10]

gtk_entry_completion_set_text_column [Gtk 2.10]	gtk_entry_get_activates_default [Gtk 2.10]
gtk_entry_get_alignment [Gtk 2.10]	gtk_entry_get_completion [Gtk 2.10]
gtk_entry_get_has_frame [Gtk 2.10]	gtk_entry_get_inner_border [Gtk 2.10]
gtk_entry_get_invisible_char [Gtk 2.10]	gtk_entry_get_layout [Gtk 2.10]
gtk_entry_get_layout_offsets [Gtk 2.10]	gtk_entry_get_max_length [Gtk 2.10]
gtk_entry_get_text [Gtk 2.10]	gtk_entry_get_type [Gobject 2.32]
gtk_entry_get_visibility [Gtk 2.10]	gtk_entry_get_width_chars [Gtk 2.10]
gtk_entry_layout_index_to_text_index [Gtk 2.10]	gtk_entry_new [Gtk 2.10]
gtk_entry_set_activates_default [Gtk 2.10]	gtk_entry_set_alignment [Gtk 2.10]
gtk_entry_set_completion [Gtk 2.10]	gtk_entry_set_has_frame [Gtk 2.10]
gtk_entry_set_inner_border [Gtk 2.10]	gtk_entry_set_invisible_char [Gtk 2.10]
gtk_entry_set_max_length [Gtk 2.10]	gtk_entry_set_text [Gtk 2.10]
gtk_entry_set_visibility [Gtk 2.10]	gtk_entry_set_width_chars [Gtk 2.10]
gtk_entry_text_index_to_layout_index [Gtk 2.10]	gtk_enumerate_printers [Gtk 2.10]
gtk_event_box_get_above_child [Gtk 2.10]	gtk_event_box_get_type [Gobject 2.32]
gtk_event_box_get_visible_window [Gtk 2.10]	gtk_event_box_new [Gtk 2.10]
gtk_event_box_set_above_child [Gtk 2.10]	gtk_event_box_set_visible_window [Gtk 2.10]
gtk_events_pending [Gtk 2.10]	gtk_expander_get_expanded [Gtk 2.10]
gtk_expander_get_label [Gtk 2.10]	gtk_expander_get_label_widget [Gtk 2.10]
gtk_expander_get_spacing [Gtk 2.10]	gtk_expander_get_type [Gobject 2.32]
gtk_expander_get_use_markup [Gtk 2.10]	gtk_expander_get_use_underline [Gtk 2.10]
gtk_expander_new [Gtk 2.10]	gtk_expander_new_with_mnemonic [Gtk 2.10]
gtk_expander_set_expanded [Gtk 2.10]	gtk_expander_set_label [Gtk 2.10]

gtk_expander_set_label_widget [Gtk 2.10]	gtk_expander_set_spacing [Gtk 2.10]
gtk_expander_set_use_markup [Gtk 2.10]	gtk_expander_set_use_underline [Gtk 2.10]
gtk_expander_style_get_type [Gobject 2.32]	gtk_false [Gtk 2.10]
gtk_file_chooser_action_get_type [Gobject 2.32]	gtk_file_chooser_add_filter [Gtk 2.10]
gtk_file_chooser_add_shortcut_folder [Gtk 2.10]	gtk_file_chooser_add_shortcut_folder_uri [Gtk 2.10]
gtk_file_chooser_button_get_focus_on_click [Gtk 2.10]	gtk_file_chooser_button_get_title [Gtk 2.10]
gtk_file_chooser_button_get_type [Gobject 2.32]	gtk_file_chooser_button_get_width_chars [Gtk 2.10]
gtk_file_chooser_button_new [Gtk 2.10]	gtk_file_chooser_button_new_with_backend [Gtk 2.10]
gtk_file_chooser_button_new_with_dialog [Gtk 2.10]	gtk_file_chooser_button_set_focus_on_click [Gtk 2.10]
gtk_file_chooser_button_set_title [Gtk 2.10]	gtk_file_chooser_button_set_width_chars [Gtk 2.10]
gtk_file_chooser_dialog_get_type [Gobject 2.32]	gtk_file_chooser_dialog_new [Gtk 2.10]
gtk_file_chooser_dialog_new_with_backend [Gtk 2.10]	gtk_file_chooser_error_get_type [Gobject 2.32]
gtk_file_chooser_error_quark [Gtk 2.10]	gtk_file_chooser_get_action [Gtk 2.10]
gtk_file_chooser_get_current_folder [Gtk 2.10]	gtk_file_chooser_get_current_folder_uri [Gtk 2.10]
gtk_file_chooser_get_do_overwrite_confirmation [Gtk 2.10]	gtk_file_chooser_get_extra_widget [Gtk 2.10]
gtk_file_chooser_get_filename [Gtk 2.10]	gtk_file_chooser_get_filenames [Gtk 2.10]
gtk_file_chooser_get_filter [Gtk 2.10]	gtk_file_chooser_get_local_only [Gtk 2.10]
gtk_file_chooser_get_preview_filename [Gtk 2.10]	gtk_file_chooser_get_preview_uri [Gtk 2.10]
gtk_file_chooser_get_preview_widget [Gtk 2.10]	gtk_file_chooser_get_preview_widget_active [Gtk 2.10]
gtk_file_chooser_get_select_multiple [Gtk 2.10]	gtk_file_chooser_get_show_hidden [Gtk 2.10]
gtk_file_chooser_get_type [Gobject 2.32]	gtk_file_chooser_get_uri [Gtk 2.10]

gtk_file_chooser_get_uris [Gtk 2.10]	gtk_file_chooser_get_use_preview_label [Gtk 2.10]
gtk_file_chooser_list_filters [Gtk 2.10]	gtk_file_chooser_list_shortcut_folder_uris [Gtk 2.10]
gtk_file_chooser_list_shortcut_folders [Gtk 2.10]	gtk_file_chooser_remove_filter [Gtk 2.10]
gtk_file_chooser_remove_shortcut_folder [Gtk 2.10]	gtk_file_chooser_remove_shortcut_folder_uri [Gtk 2.10]
gtk_file_chooser_select_all [Gtk 2.10]	gtk_file_chooser_select_filename [Gtk 2.10]
gtk_file_chooser_select_uri [Gtk 2.10]	gtk_file_chooser_set_action [Gtk 2.10]
gtk_file_chooser_set_current_folder [Gtk 2.10]	gtk_file_chooser_set_current_folder_uri [Gtk 2.10]
gtk_file_chooser_set_current_name [Gtk 2.10]	gtk_file_chooser_set_do_overwrite_confirmation [Gtk 2.10]
gtk_file_chooser_set_extra_widget [Gtk 2.10]	gtk_file_chooser_set_filename [Gtk 2.10]
gtk_file_chooser_set_filter [Gtk 2.10]	gtk_file_chooser_set_local_only [Gtk 2.10]
gtk_file_chooser_set_preview_widget [Gtk 2.10]	gtk_file_chooser_set_preview_widget_active [Gtk 2.10]
gtk_file_chooser_set_select_multiple [Gtk 2.10]	gtk_file_chooser_set_show_hidden [Gtk 2.10]
gtk_file_chooser_set_uri [Gtk 2.10]	gtk_file_chooser_set_use_preview_label [Gtk 2.10]
gtk_file_chooser_unselect_all [Gtk 2.10]	gtk_file_chooser_unselect_filename [Gtk 2.10]
gtk_file_chooser_unselect_uri [Gtk 2.10]	gtk_file_chooser_widget_get_type [Gobject 2.32]
gtk_file_chooser_widget_new [Gtk 2.10]	gtk_file_chooser_widget_new_with_backend [Gtk 2.10]
gtk_file_filter_add_custom [Gtk 2.10]	gtk_file_filter_add_mime_type [Gtk 2.10]
gtk_file_filter_add_pattern [Gtk 2.10]	gtk_file_filter_add_pixbuf_formats [Gtk 2.10]
gtk_file_filter_filter [Gtk 2.10]	gtk_file_filter_flags_get_type [Gobject 2.32]
gtk_file_filter_get_name [Gtk 2.10]	gtk_file_filter_get_needed [Gtk 2.10]
gtk_file_filter_get_type [Gobject 2.32]	gtk_file_filter_new [Gtk 2.10]
gtk_file_filter_set_name [Gtk 2.10]	gtk_file_selection_complete [Gtk 2.10]

gtk_file_selection_get_filename [Gtk 2.10]	gtk_file_selection_get_select_multiple [Gtk 2.10]
gtk_file_selection_get_selections [Gtk 2.10]	gtk_file_selection_get_type [Gobject 2.32]
gtk_file_selection_hide_fileop_buttons [Gtk 2.10]	gtk_file_selection_new [Gtk 2.10]
gtk_file_selection_set_filename [Gtk 2.10]	gtk_file_selection_set_select_multiple [Gtk 2.10]
gtk_file_selection_show_fileop_buttons [Gtk 2.10]	gtk_fixed_get_has_window [Gtk 2.10]
gtk_fixed_get_type [Gobject 2.32]	gtk_fixed_move [Gtk 2.10]
gtk_fixed_new [Gtk 2.10]	gtk_fixed_put [Gtk 2.10]
gtk_fixed_set_has_window [Gtk 2.10]	gtk_font_button_get_font_name [Gtk 2.10]
gtk_font_button_get_show_size [Gtk 2.10]	gtk_font_button_get_show_style [Gtk 2.10]
gtk_font_button_get_title [Gtk 2.10]	gtk_font_button_get_type [Gobject 2.32]
gtk_font_button_get_use_font [Gtk 2.10]	gtk_font_button_get_use_size [Gtk 2.10]
gtk_font_button_new [Gtk 2.10]	gtk_font_button_new_with_font [Gtk 2.10]
gtk_font_button_set_font_name [Gtk 2.10]	gtk_font_button_set_show_size [Gtk 2.10]
gtk_font_button_set_show_style [Gtk 2.10]	gtk_font_button_set_title [Gtk 2.10]
gtk_font_button_set_use_font [Gtk 2.10]	gtk_font_button_set_use_size [Gtk 2.10]
gtk_font_selection_dialog_get_font_name [Gtk 2.10]	gtk_font_selection_dialog_get_preview_text [Gtk 2.10]
gtk_font_selection_dialog_get_type [Gobject 2.32]	gtk_font_selection_dialog_new [Gtk 2.10]
gtk_font_selection_dialog_set_font_name [Gtk 2.10]	gtk_font_selection_dialog_set_preview_text [Gtk 2.10]
gtk_font_selection_get_font_name [Gtk 2.10]	gtk_font_selection_get_preview_text [Gtk 2.10]
gtk_font_selection_get_type [Gobject 2.32]	gtk_font_selection_new [Gtk 2.10]
gtk_font_selection_set_font_name [Gtk 2.10]	gtk_font_selection_set_preview_text [Gtk 2.10]
gtk_frame_get_label [Gtk 2.10]	gtk_frame_get_label_align [Gtk 2.10]

gtk_frame_get_label_widget [Gtk 2.10]	gtk_frame_get_shadow_type [Gtk 2.10]
gtk_frame_get_type [Gobject 2.32]	gtk_frame_new [Gtk 2.10]
gtk_frame_set_label [Gtk 2.10]	gtk_frame_set_label_align [Gtk 2.10]
gtk_frame_set_label_widget [Gtk 2.10]	gtk_frame_set_shadow_type [Gtk 2.10]
gtk_gamma_curve_get_type [Gobject 2.32]	gtk_gamma_curve_new [Gtk 2.10]
gtk_gc_get [Gtk 2.10]	gtk_gc_release [Gtk 2.10]
gtk_get_current_event [Gtk 2.10]	gtk_get_current_event_state [Gtk 2.10]
gtk_get_current_event_time [Gtk 2.10]	gtk_get_default_language [Gtk 2.10]
gtk_get_event_widget [Gtk 2.10]	gtk_get_option_group [Gtk 2.10]
gtk_grab_add [Gtk 2.10]	gtk_grab_get_current [Gtk 2.10]
gtk_grab_remove [Gtk 2.10]	gtk_handle_box_get_handle_position [Gtk 2.10]
gtk_handle_box_get_shadow_type [Gtk 2.10]	gtk_handle_box_get_snap_edge [Gtk 2.10]
gtk_handle_box_get_type [Gobject 2.32]	gtk_handle_box_new [Gtk 2.10]
gtk_handle_box_set_handle_position [Gtk 2.10]	gtk_handle_box_set_shadow_type [Gtk 2.10]
gtk_handle_box_set_snap_edge [Gtk 2.10]	gtk_hbox_get_type [Gobject 2.32]
gtk_hbox_new [Gtk 2.10]	gtk_hbutton_box_get_type [Gobject 2.32]
gtk_hbutton_box_new [Gtk 2.10]	gtk_hpaned_get_type [Gobject 2.32]
gtk_hpaned_new [Gtk 2.10]	gtk_hruler_get_type [Gobject 2.32]
gtk_hruler_new [Gtk 2.10]	gtk_hscale_get_type [Gobject 2.32]
gtk_hscale_new [Gtk 2.10]	gtk_hscale_new_with_range [Gtk 2.10]
gtk_hscrollbar_get_type [Gobject 2.32]	gtk_hscrollbar_new [Gtk 2.10]
gtk_hseparator_get_type [Gobject 2.32]	gtk_hseparator_new [Gtk 2.10]
gtk_icon_factory_add [Gtk 2.10]	gtk_icon_factory_add_default [Gtk 2.10]
gtk_icon_factory_get_type [Gobject 2.32]	gtk_icon_factory_lookup [Gtk 2.10]

gtk_icon_factory_lookup_default [Gtk 2.10]	gtk_icon_factory_new [Gtk 2.10]
gtk_icon_factory_remove_default [Gtk 2.10]	gtk_icon_info_copy [Gtk 2.10]
gtk_icon_info_free [Gtk 2.10]	gtk_icon_info_get_attach_points [Gtk 2.10]
gtk_icon_info_get_base_size [Gtk 2.10]	gtk_icon_info_get_builtin_pixbuf [Gtk 2.10]
gtk_icon_info_get_display_name [Gtk 2.10]	gtk_icon_info_get_embedded_rect [Gtk 2.10]
gtk_icon_info_get_filename [Gtk 2.10]	gtk_icon_info_get_type [Gobject 2.32]
gtk_icon_info_load_icon [Gtk 2.10]	gtk_icon_info_set_raw_coordinates [Gtk 2.10]
gtk_icon_lookup_flags_get_type [Gobject 2.32]	gtk_icon_set_add_source [Gtk 2.10]
gtk_icon_set_copy [Gtk 2.10]	gtk_icon_set_get_sizes [Gtk 2.10]
gtk_icon_set_get_type [Gobject 2.32]	gtk_icon_set_new [Gtk 2.10]
gtk_icon_set_new_from_pixbuf [Gtk 2.10]	gtk_icon_set_ref [Gtk 2.10]
gtk_icon_set_render_icon [Gtk 2.10]	gtk_icon_set_unref [Gtk 2.10]
gtk_icon_size_from_name [Gtk 2.10]	gtk_icon_size_get_name [Gtk 2.10]
gtk_icon_size_get_type [Gobject 2.32]	gtk_icon_size_lookup [Gtk 2.10]
gtk_icon_size_lookup_for_settings [Gtk 2.10]	gtk_icon_size_register [Gtk 2.10]
gtk_icon_size_register_alias [Gtk 2.10]	gtk_icon_source_copy [Gtk 2.10]
gtk_icon_source_free [Gtk 2.10]	gtk_icon_source_get_direction [Gtk 2.10]
gtk_icon_source_get_direction_wildcarded [Gtk 2.10]	gtk_icon_source_get_filename [Gtk 2.10]
gtk_icon_source_get_icon_name [Gtk 2.10]	gtk_icon_source_get_pixbuf [Gtk 2.10]
gtk_icon_source_get_size [Gtk 2.10]	gtk_icon_source_get_size_wildcarded [Gtk 2.10]
gtk_icon_source_get_state [Gtk 2.10]	gtk_icon_source_get_state_wildcarded [Gtk 2.10]
gtk_icon_source_get_type [Gobject 2.32]	gtk_icon_source_new [Gtk 2.10]
gtk_icon_source_set_direction [Gtk 2.10]	gtk_icon_source_set_direction_wildcarded [Gtk 2.10]

gtk_icon_source_set_filename [Gtk 2.10]	gtk_icon_source_set_icon_name [Gtk 2.10]
gtk_icon_source_set_pixbuf [Gtk 2.10]	gtk_icon_source_set_size [Gtk 2.10]
gtk_icon_source_set_size_wildcard [Gtk 2.10]	gtk_icon_source_set_state [Gtk 2.10]
gtk_icon_source_set_state_wildcard [Gtk 2.10]	gtk_icon_theme_add_builtin_icon [Gtk 2.10]
gtk_icon_theme_append_search_path [Gtk 2.10]	gtk_icon_theme_error_get_type [Gobject 2.32]
gtk_icon_theme_error_quark [Gtk 2.10]	gtk_icon_theme_get_default [Gtk 2.10]
gtk_icon_theme_get_example_icon_name [Gtk 2.10]	gtk_icon_theme_get_for_screen [Gtk 2.10]
gtk_icon_theme_get_icon_sizes [Gtk 2.10]	gtk_icon_theme_get_search_path [Gtk 2.10]
gtk_icon_theme_get_type [Gobject 2.32]	gtk_icon_theme_has_icon [Gtk 2.10]
gtk_icon_theme_list_icons [Gtk 2.10]	gtk_icon_theme_load_icon [Gtk 2.10]
gtk_icon_theme_lookup_icon [Gtk 2.10]	gtk_icon_theme_new [Gtk 2.10]
gtk_icon_theme_prepend_search_path [Gtk 2.10]	gtk_icon_theme_rescan_if_needed [Gtk 2.10]
gtk_icon_theme_set_custom_theme [Gtk 2.10]	gtk_icon_theme_set_screen [Gtk 2.10]
gtk_icon_theme_set_search_path [Gtk 2.10]	gtk_icon_view_create_drag_icon [Gtk 2.10]
gtk_icon_view_enable_model_drag_dest [Gtk 2.10]	gtk_icon_view_enable_model_drag_source [Gtk 2.10]
gtk_icon_view_get_column_spacing [Gtk 2.10]	gtk_icon_view_get_columns [Gtk 2.10]
gtk_icon_view_get_cursor [Gtk 2.10]	gtk_icon_view_get_dest_item_at_pos [Gtk 2.10]
gtk_icon_view_get_drag_dest_item [Gtk 2.10]	gtk_icon_view_get_item_at_pos [Gtk 2.10]
gtk_icon_view_get_item_width [Gtk 2.10]	gtk_icon_view_get_margin [Gtk 2.10]
gtk_icon_view_get_markup_column [Gtk 2.10]	gtk_icon_view_get_model [Gtk 2.10]
gtk_icon_view_get_orientation [Gtk 2.10]	gtk_icon_view_get_path_at_pos [Gtk 2.10]

gtk_icon_view_get_pixbuf_column [Gtk 2.10]	gtk_icon_view_get_reorderable [Gtk 2.10]
gtk_icon_view_get_row_spacing [Gtk 2.10]	gtk_icon_view_get_selected_items [Gtk 2.10]
gtk_icon_view_get_selection_mode [Gtk 2.10]	gtk_icon_view_get_spacing [Gtk 2.10]
gtk_icon_view_get_text_column [Gtk 2.10]	gtk_icon_view_get_type [Gobject 2.32]
gtk_icon_view_get_visible_range [Gtk 2.10]	gtk_icon_view_item_activated [Gtk 2.10]
gtk_icon_view_new [Gtk 2.10]	gtk_icon_view_new_with_model [Gtk 2.10]
gtk_icon_view_path_is_selected [Gtk 2.10]	gtk_icon_view_scroll_to_path [Gtk 2.10]
gtk_icon_view_select_all [Gtk 2.10]	gtk_icon_view_select_path [Gtk 2.10]
gtk_icon_view_selected_foreach [Gtk 2.10]	gtk_icon_view_set_column_spacing [Gtk 2.10]
gtk_icon_view_set_columns [Gtk 2.10]	gtk_icon_view_set_cursor [Gtk 2.10]
gtk_icon_view_set_drag_dest_item [Gtk 2.10]	gtk_icon_view_set_item_width [Gtk 2.10]
gtk_icon_view_set_margin [Gtk 2.10]	gtk_icon_view_set_markup_column [Gtk 2.10]
gtk_icon_view_set_model [Gtk 2.10]	gtk_icon_view_set_orientation [Gtk 2.10]
gtk_icon_view_set_pixbuf_column [Gtk 2.10]	gtk_icon_view_set_reorderable [Gtk 2.10]
gtk_icon_view_set_row_spacing [Gtk 2.10]	gtk_icon_view_set_selection_mode [Gtk 2.10]
gtk_icon_view_set_spacing [Gtk 2.10]	gtk_icon_view_set_text_column [Gtk 2.10]
gtk_icon_view_unselect_all [Gtk 2.10]	gtk_icon_view_unselect_path [Gtk 2.10]
gtk_icon_view_unset_model_drag_dest [Gtk 2.10]	gtk_icon_view_unset_model_drag_source [Gtk 2.10]
gtk_identifier_get_type [Gobject 2.32]	gtk_im_context_delete_surrounding [Gtk 2.10]
gtk_im_context_filter_keypress [Gtk 2.10]	gtk_im_context_focus_in [Gtk 2.10]
gtk_im_context_focus_out [Gtk 2.10]	gtk_im_context_get_preedit_string [Gtk 2.10]

gtk_im_context_get_surrounding [Gtk 2.10]	gtk_im_context_get_type [Gobject 2.32]
gtk_im_context_reset [Gtk 2.10]	gtk_im_context_set_client_window [Gtk 2.10]
gtk_im_context_set_cursor_location [Gtk 2.10]	gtk_im_context_set_surrounding [Gtk 2.10]
gtk_im_context_set_use_preedit [Gtk 2.10]	gtk_im_context_simple_add_table [Gtk 2.10]
gtk_im_context_simple_get_type [Gobject 2.32]	gtk_im_context_simple_new [Gtk 2.10]
gtk_im_multicontext_append_menus [Gtk 2.10]	gtk_im_multicontext_get_type [Gobject 2.32]
gtk_im_multicontext_new [Gtk 2.10]	gtk_im_preedit_style_get_type [Gobject 2.32]
gtk_im_status_style_get_type [Gobject 2.32]	gtk_image_clear [Gtk 2.10]
gtk_image_get_animation [Gtk 2.10]	gtk_image_get_icon_name [Gtk 2.10]
gtk_image_get_icon_set [Gtk 2.10]	gtk_image_get_image [Gtk 2.10]
gtk_image_get_pixmap [Gtk 2.10]	gtk_image_get_pixel_size [Gtk 2.10]
gtk_image_get_pixmap [Gtk 2.10]	gtk_image_get_stock [Gtk 2.10]
gtk_image_get_storage_type [Gtk 2.10]	gtk_image_get_type [Gobject 2.32]
gtk_image_menu_item_get_image [Gtk 2.10]	gtk_image_menu_item_get_type [Gobject 2.32]
gtk_image_menu_item_new [Gtk 2.10]	gtk_image_menu_item_new_from_stock [Gtk 2.10]
gtk_image_menu_item_new_with_label [Gtk 2.10]	gtk_image_menu_item_new_with_mnemonic [Gtk 2.10]
gtk_image_menu_item_set_image [Gtk 2.10]	gtk_image_new [Gtk 2.10]
gtk_image_new_from_animation [Gtk 2.10]	gtk_image_new_from_file [Gtk 2.10]
gtk_image_new_from_icon_name [Gtk 2.10]	gtk_image_new_from_icon_set [Gtk 2.10]
gtk_image_new_from_image [Gtk 2.10]	gtk_image_new_from_pixmap [Gtk 2.10]
gtk_image_new_from_pixmap [Gtk 2.10]	gtk_image_new_from_stock [Gtk 2.10]
gtk_image_set_from_animation [Gtk 2.10]	gtk_image_set_from_file [Gtk 2.10]

gtk_image_set_from_icon_name [Gtk 2.10]	gtk_image_set_from_icon_set [Gtk 2.10]
gtk_image_set_from_image [Gtk 2.10]	gtk_image_set_from_pixbuf [Gtk 2.10]
gtk_image_set_from_pixmap [Gtk 2.10]	gtk_image_set_from_stock [Gtk 2.10]
gtk_image_set_pixel_size [Gtk 2.10]	gtk_image_type_get_type [Gobject 2.32]
gtk_init [Gtk 2.10]	gtk_init_add [Gtk 2.10]
gtk_init_check [Gtk 2.10]	gtk_init_with_args [Gtk 2.10]
gtk_input_dialog_get_type [Gobject 2.32]	gtk_input_dialog_new [Gtk 2.10]
gtk_invisible_get_screen [Gtk 2.10]	gtk_invisible_get_type [Gobject 2.32]
gtk_invisible_new [Gtk 2.10]	gtk_invisible_new_for_screen [Gtk 2.10]
gtk_invisible_set_screen [Gtk 2.10]	gtk_item_deselect [Gtk 2.10]
gtk_item_get_type [Gobject 2.32]	gtk_item_select [Gtk 2.10]
gtk_item_toggle [Gtk 2.10]	gtk_justification_get_type [Gobject 2.32]
gtk_key_snooper_install [Gtk 2.10]	gtk_key_snooper_remove [Gtk 2.10]
gtk_label_get_angle [Gtk 2.10]	gtk_label_get_attributes [Gtk 2.10]
gtk_label_get_ellipsize [Gtk 2.10]	gtk_label_get_justify [Gtk 2.10]
gtk_label_get_label [Gtk 2.10]	gtk_label_get_layout [Gtk 2.10]
gtk_label_get_layout_offsets [Gtk 2.10]	gtk_label_get_line_wrap [Gtk 2.10]
gtk_label_get_line_wrap_mode [Gtk 2.10]	gtk_label_get_max_width_chars [Gtk 2.10]
gtk_label_get_mnemonic_keyval [Gtk 2.10]	gtk_label_get_mnemonic_widget [Gtk 2.10]
gtk_label_get_selectable [Gtk 2.10]	gtk_label_get_selection_bounds [Gtk 2.10]
gtk_label_get_single_line_mode [Gtk 2.10]	gtk_label_get_text [Gtk 2.10]
gtk_label_get_type [Gobject 2.32]	gtk_label_get_use_markup [Gtk 2.10]
gtk_label_get_use_underline [Gtk 2.10]	gtk_label_get_width_chars [Gtk 2.10]
gtk_label_new [Gtk 2.10]	gtk_label_new_with_mnemonic [Gtk 2.10]
gtk_label_select_region [Gtk 2.10]	gtk_label_set_angle [Gtk 2.10]

gtk_label_set_attributes [Gtk 2.10]	gtk_label_set_ellipsize [Gtk 2.10]
gtk_label_set_justify [Gtk 2.10]	gtk_label_set_label [Gtk 2.10]
gtk_label_set_line_wrap [Gtk 2.10]	gtk_label_set_line_wrap_mode [Gtk 2.10]
gtk_label_set_markup [Gtk 2.10]	gtk_label_set_markup_with_mnemonic [Gtk 2.10]
gtk_label_set_max_width_chars [Gtk 2.10]	gtk_label_set_mnemonic_widget [Gtk 2.10]
gtk_label_set_pattern [Gtk 2.10]	gtk_label_set_selectable [Gtk 2.10]
gtk_label_set_single_line_mode [Gtk 2.10]	gtk_label_set_text [Gtk 2.10]
gtk_label_set_text_with_mnemonic [Gtk 2.10]	gtk_label_set_use_markup [Gtk 2.10]
gtk_label_set_use_underline [Gtk 2.10]	gtk_label_set_width_chars [Gtk 2.10]
gtk_layout_get_hadjustment [Gtk 2.10]	gtk_layout_get_size [Gtk 2.10]
gtk_layout_get_type [Gobject 2.32]	gtk_layout_get_vadjustment [Gtk 2.10]
gtk_layout_move [Gtk 2.10]	gtk_layout_new [Gtk 2.10]
gtk_layout_put [Gtk 2.10]	gtk_layout_set_hadjustment [Gtk 2.10]
gtk_layout_set_size [Gtk 2.10]	gtk_layout_set_vadjustment [Gtk 2.10]
gtk_link_button_get_type [Gtk 2.10]	gtk_link_button_get_uri [Gtk 2.10]
gtk_link_button_new [Gtk 2.10]	gtk_link_button_new_with_label [Gtk 2.10]
gtk_link_button_set_uri [Gtk 2.10]	gtk_link_button_set_uri_hook [Gtk 2.10]
gtk_list_store_append [Gtk 2.10]	gtk_list_store_clear [Gtk 2.10]
gtk_list_store_get_type [Gobject 2.32]	gtk_list_store_insert [Gtk 2.10]
gtk_list_store_insert_after [Gtk 2.10]	gtk_list_store_insert_before [Gtk 2.10]
gtk_list_store_insert_with_values [Gtk 2.10]	gtk_list_store_insert_with_valuesv [Gtk 2.10]
gtk_list_store_iter_is_valid [Gtk 2.10]	gtk_list_store_move_after [Gtk 2.10]
gtk_list_store_move_before [Gtk 2.10]	gtk_list_store_new [Gtk 2.10]
gtk_list_store_newv [Gtk 2.10]	gtk_list_store_prepend [Gtk 2.10]
gtk_list_store_remove [Gtk 2.10]	gtk_list_store_reorder [Gtk 2.10]

gtk_list_store_set [Gtk 2.10]	gtk_list_store_set_column_types [Gtk 2.10]
gtk_list_store_set_valist [Gtk 2.10]	gtk_list_store_set_value [Gtk 2.10]
gtk_list_store_swap [Gtk 2.10]	gtk_main [Gtk 2.10]
gtk_main_do_event [Gtk 2.10]	gtk_main_iteration [Gtk 2.10]
gtk_main_iteration_do [Gtk 2.10]	gtk_main_level [Gtk 2.10]
gtk_main_quit [Gtk 2.10]	gtk_match_type_get_type [Gobject 2.32]
gtk_menu_attach [Gtk 2.10]	gtk_menu_attach_to_widget [Gtk 2.10]
gtk_menu_bar_get_child_pack_direction [Gtk 2.10]	gtk_menu_bar_get_pack_direction [Gtk 2.10]
gtk_menu_bar_get_type [Gobject 2.32]	gtk_menu_bar_new [Gtk 2.10]
gtk_menu_bar_set_child_pack_direction [Gtk 2.10]	gtk_menu_bar_set_pack_direction [Gtk 2.10]
gtk_menu_detach [Gtk 2.10]	gtk_menu_direction_type_get_type [Gobject 2.32]
gtk_menu_get_accel_group [Gtk 2.10]	gtk_menu_get_active [Gtk 2.10]
gtk_menu_get_attach_widget [Gtk 2.10]	gtk_menu_get_for_attach_widget [Gtk 2.10]
gtk_menu_get_tearoff_state [Gtk 2.10]	gtk_menu_get_title [Gtk 2.10]
gtk_menu_get_type [Gobject 2.32]	gtk_menu_item_activate [Gtk 2.10]
gtk_menu_item_deselect [Gtk 2.10]	gtk_menu_item_get_right_justified [Gtk 2.10]
gtk_menu_item_get_submenu [Gtk 2.10]	gtk_menu_item_get_type [Gobject 2.32]
gtk_menu_item_new [Gtk 2.10]	gtk_menu_item_new_with_label [Gtk 2.10]
gtk_menu_item_new_with_mnemonic [Gtk 2.10]	gtk_menu_item_remove_submenu [Gtk 2.10]
gtk_menu_item_select [Gtk 2.10]	gtk_menu_item_set_accel_path [Gtk 2.10]
gtk_menu_item_set_right_justified [Gtk 2.10]	gtk_menu_item_set_submenu [Gtk 2.10]
gtk_menu_item_toggle_size_allocate [Gtk 2.10]	gtk_menu_item_toggle_size_request [Gtk 2.10]
gtk_menu_new [Gtk 2.10]	gtk_menu_popdown [Gtk 2.10]
gtk_menu_popup [Gtk 2.10]	gtk_menu_reorder_child [Gtk 2.10]

gtk_menu_reposition [Gtk 2.10]	gtk_menu_set_accel_group [Gtk 2.10]
gtk_menu_set_accel_path [Gtk 2.10]	gtk_menu_set_active [Gtk 2.10]
gtk_menu_set_monitor [Gtk 2.10]	gtk_menu_set_screen [Gtk 2.10]
gtk_menu_set_tearoff_state [Gtk 2.10]	gtk_menu_set_title [Gtk 2.10]
gtk_menu_shell_activate_item [Gtk 2.10]	gtk_menu_shell_append [Gtk 2.10]
gtk_menu_shell_cancel [Gtk 2.10]	gtk_menu_shell_deactivate [Gtk 2.10]
gtk_menu_shell_deselect [Gtk 2.10]	gtk_menu_shell_get_take_focus [Gtk 2.10]
gtk_menu_shell_get_type [Gobject 2.32]	gtk_menu_shell_insert [Gtk 2.10]
gtk_menu_shell_prepend [Gtk 2.10]	gtk_menu_shell_select_first [Gtk 2.10]
gtk_menu_shell_select_item [Gtk 2.10]	gtk_menu_shell_set_take_focus [Gtk 2.10]
gtk_menu_tool_button_get_menu [Gtk 2.10]	gtk_menu_tool_button_get_type [Gobject 2.32]
gtk_menu_tool_button_new [Gtk 2.10]	gtk_menu_tool_button_new_from_stock [Gtk 2.10]
gtk_menu_tool_button_set_arrow_tooltip [Gtk 2.10]	gtk_menu_tool_button_set_menu [Gtk 2.10]
gtk_message_dialog_format_secondary_markup [Gtk 2.10]	gtk_message_dialog_format_secondary_text [Gtk 2.10]
gtk_message_dialog_get_type [Gobject 2.32]	gtk_message_dialog_new [Gtk 2.10]
gtk_message_dialog_new_with_markup [Gtk 2.10]	gtk_message_dialog_set_image [Gtk 2.10]
gtk_message_dialog_set_markup [Gtk 2.10]	gtk_message_type_get_type [Gobject 2.32]
gtk_metric_type_get_type [Gobject 2.32]	gtk_misc_get_alignment [Gtk 2.10]
gtk_misc_get_padding [Gtk 2.10]	gtk_misc_get_type [Gobject 2.32]
gtk_misc_set_alignment [Gtk 2.10]	gtk_misc_set_padding [Gtk 2.10]
gtk_movement_step_get_type [Gobject 2.32]	gtk_notebook_append_page [Gtk 2.10]
gtk_notebook_append_page_menu [Gtk 2.10]	gtk_notebook_get_current_page [Gtk 2.10]
gtk_notebook_get_menu_label [Gtk 2.10]	gtk_notebook_get_menu_label_text [Gtk 2.10]

gtk_notebook_get_n_pages [Gtk 2.10]	gtk_notebook_get_nth_page [Gtk 2.10]
gtk_notebook_get_scrollable [Gtk 2.10]	gtk_notebook_get_show_border [Gtk 2.10]
gtk_notebook_get_show_tabs [Gtk 2.10]	gtk_notebook_get_tab_detachable [Gtk 2.10]
gtk_notebook_get_tab_label [Gtk 2.10]	gtk_notebook_get_tab_label_text [Gtk 2.10]
gtk_notebook_get_tab_pos [Gtk 2.10]	gtk_notebook_get_tab_reorderable [Gtk 2.10]
gtk_notebook_get_type [Gobject 2.32]	gtk_notebook_insert_page [Gtk 2.10]
gtk_notebook_insert_page_menu [Gtk 2.10]	gtk_notebook_new [Gtk 2.10]
gtk_notebook_next_page [Gtk 2.10]	gtk_notebook_page_num [Gtk 2.10]
gtk_notebook_popup_disable [Gtk 2.10]	gtk_notebook_popup_enable [Gtk 2.10]
gtk_notebook_prepend_page [Gtk 2.10]	gtk_notebook_prepend_page_menu [Gtk 2.10]
gtk_notebook_prev_page [Gtk 2.10]	gtk_notebook_query_tab_label_packing [Gtk 2.10]
gtk_notebook_remove_page [Gtk 2.10]	gtk_notebook_reorder_child [Gtk 2.10]
gtk_notebook_set_current_page [Gtk 2.10]	gtk_notebook_set_menu_label [Gtk 2.10]
gtk_notebook_set_menu_label_text [Gtk 2.10]	gtk_notebook_set_scrollable [Gtk 2.10]
gtk_notebook_set_show_border [Gtk 2.10]	gtk_notebook_set_show_tabs [Gtk 2.10]
gtk_notebook_set_tab_detachable [Gtk 2.10]	gtk_notebook_set_tab_label [Gtk 2.10]
gtk_notebook_set_tab_label_packing [Gtk 2.10]	gtk_notebook_set_tab_label_text [Gtk 2.10]
gtk_notebook_set_tab_pos [Gtk 2.10]	gtk_notebook_set_tab_reorderable [Gtk 2.10]
gtk_notebook_set_window_creation_hook [Gtk 2.10]	gtk_notebook_tab_get_type [Gobject 2.32]
gtk_object_destroy [Gtk 2.10]	gtk_object_flags_get_type [Gobject 2.32]
gtk_object_get_type [Gobject 2.32]	gtk_object_sink [Gtk 2.10]
gtk_orientation_get_type [Gobject 2.32]	gtk_pack_direction_get_type [Gobject 2.32]

gtk_pack_type_get_type [Gobject 2.32]	gtk_page_orientation_get_type [Gtk 2.10]
gtk_page_set_get_type [Gtk 2.10]	gtk_page_setup_copy [Gtk 2.10]
gtk_page_setup_get_bottom_margin [Gtk 2.10]	gtk_page_setup_get_left_margin [Gtk 2.10]
gtk_page_setup_get_orientation [Gtk 2.10]	gtk_page_setup_get_page_height [Gtk 2.10]
gtk_page_setup_get_page_width [Gtk 2.10]	gtk_page_setup_get_paper_height [Gtk 2.10]
gtk_page_setup_get_paper_size [Gtk 2.10]	gtk_page_setup_get_paper_width [Gtk 2.10]
gtk_page_setup_get_right_margin [Gtk 2.10]	gtk_page_setup_get_top_margin [Gtk 2.10]
gtk_page_setup_get_type [Gtk 2.10]	gtk_page_setup_new [Gtk 2.10]
gtk_page_setup_set_bottom_margin [Gtk 2.10]	gtk_page_setup_set_left_margin [Gtk 2.10]
gtk_page_setup_set_orientation [Gtk 2.10]	gtk_page_setup_set_paper_size [Gtk 2.10]
gtk_page_setup_set_paper_size_and_default_margins [Gtk 2.10]	gtk_page_setup_set_right_margin [Gtk 2.10]
gtk_page_setup_set_top_margin [Gtk 2.10]	gtk_page_setup_unix_dialog_get_page_setup [Gtk 2.10]
gtk_page_setup_unix_dialog_get_print_settings [Gtk 2.10]	gtk_page_setup_unix_dialog_get_type [Gtk 2.10]
gtk_page_setup_unix_dialog_new [Gtk 2.10]	gtk_page_setup_unix_dialog_set_page_setup [Gtk 2.10]
gtk_page_setup_unix_dialog_set_print_settings [Gtk 2.10]	gtk_paint_arrow [Gtk 2.10]
gtk_paint_box [Gtk 2.10]	gtk_paint_box_gap [Gtk 2.10]
gtk_paint_check [Gtk 2.10]	gtk_paint_diamond [Gtk 2.10]
gtk_paint_expander [Gtk 2.10]	gtk_paint_extension [Gtk 2.10]
gtk_paint_flat_box [Gtk 2.10]	gtk_paint_focus [Gtk 2.10]
gtk_paint_handle [Gtk 2.10]	gtk_paint_hline [Gtk 2.10]
gtk_paint_layout [Gtk 2.10]	gtk_paint_option [Gtk 2.10]
gtk_paint_polygon [Gtk 2.10]	gtk_paint_resize_grip [Gtk 2.10]
gtk_paint_shadow [Gtk 2.10]	gtk_paint_shadow_gap [Gtk 2.10]
gtk_paint_slider [Gtk 2.10]	gtk_paint_tab [Gtk 2.10]
gtk_paint_vline [Gtk 2.10]	gtk_paned_add1 [Gtk 2.10]
gtk_paned_add2 [Gtk 2.10]	gtk_paned_get_child1 [Gtk 2.10]

gtk_paned_get_child2 [Gtk 2.10]	gtk_paned_get_position [Gtk 2.10]
gtk_paned_get_type [Gobject 2.32]	gtk_paned_pack1 [Gtk 2.10]
gtk_paned_pack2 [Gtk 2.10]	gtk_paned_set_position [Gtk 2.10]
gtk_paper_size_copy [Gtk 2.10]	gtk_paper_size_free [Gtk 2.10]
gtk_paper_size_get_default [Gtk 2.10]	gtk_paper_size_get_default_bottom_margin [Gtk 2.10]
gtk_paper_size_get_default_left_margin [Gtk 2.10]	gtk_paper_size_get_default_right_margin [Gtk 2.10]
gtk_paper_size_get_default_top_margin [Gtk 2.10]	gtk_paper_size_get_display_name [Gtk 2.10]
gtk_paper_size_get_height [Gtk 2.10]	gtk_paper_size_get_name [Gtk 2.10]
gtk_paper_size_get_ppd_name [Gtk 2.10]	gtk_paper_size_get_type [Gtk 2.10]
gtk_paper_size_get_width [Gtk 2.10]	gtk_paper_size_is_custom [Gtk 2.10]
gtk_paper_size_is_equal [Gtk 2.10]	gtk_paper_size_new [Gtk 2.10]
gtk_paper_size_new_custom [Gtk 2.10]	gtk_paper_size_new_from_ppd [Gtk 2.10]
gtk_paper_size_set_size [Gtk 2.10]	gtk_parse_args [Gtk 2.10]
gtk_path_priority_type_get_type [Gobject 2.32]	gtk_path_type_get_type [Gobject 2.32]
gtk_plugin_construct [Gtk 2.10]	gtk_plugin_construct_for_display [Gtk 2.10]
gtk_plugin_get_id [Gtk 2.10]	gtk_plugin_get_type [Gobject 2.32]
gtk_plugin_new [Gtk 2.10]	gtk_plugin_new_for_display [Gtk 2.10]
gtk_policy_type_get_type [Gobject 2.32]	gtk_position_type_get_type [Gobject 2.32]
gtk_print_capabilities_get_type [Gtk 2.10]	gtk_print_context_create_pango_context [Gtk 2.10]
gtk_print_context_create_pango_layout [Gtk 2.10]	gtk_print_context_get_cairo_context [Gtk 2.10]
gtk_print_context_get_dpi_x [Gtk 2.10]	gtk_print_context_get_dpi_y [Gtk 2.10]
gtk_print_context_get_height [Gtk 2.10]	gtk_print_context_get_page_setup [Gtk 2.10]
gtk_print_context_get_pango_fontmap [Gtk 2.10]	gtk_print_context_get_type [Gtk 2.10]
gtk_print_context_get_width [Gtk 2.10]	gtk_print_context_set_cairo_context [Gtk 2.10]
gtk_print_duplex_get_type [Gtk 2.10]	gtk_print_error_get_type [Gtk 2.10]

gtk_print_error_quark [Gtk 2.10]	gtk_print_job_get_printer [Gtk 2.10]
gtk_print_job_get_settings [Gtk 2.10]	gtk_print_job_get_status [Gtk 2.10]
gtk_print_job_get_surface [Gtk 2.10]	gtk_print_job_get_title [Gtk 2.10]
gtk_print_job_get_track_print_status [Gtk 2.10]	gtk_print_job_get_type [Gtk 2.10]
gtk_print_job_new [Gtk 2.10]	gtk_print_job_send [Gtk 2.10]
gtk_print_job_set_source_file [Gtk 2.10]	gtk_print_job_set_track_print_status [Gtk 2.10]
gtk_print_operation_action_get_type [Gtk 2.10]	gtk_print_operation_cancel [Gtk 2.10]
gtk_print_operation_get_default_page_setup [Gtk 2.10]	gtk_print_operation_get_error [Gtk 2.10]
gtk_print_operation_get_print_settings [Gtk 2.10]	gtk_print_operation_get_status [Gtk 2.10]
gtk_print_operation_get_status_string [Gtk 2.10]	gtk_print_operation_get_type [Gtk 2.10]
gtk_print_operation_is_finished [Gtk 2.10]	gtk_print_operation_new [Gtk 2.10]
gtk_print_operation_preview_end_preview [Gtk 2.10]	gtk_print_operation_preview_get_type [Gtk 2.10]
gtk_print_operation_preview_is_selected [Gtk 2.10]	gtk_print_operation_preview_render_page [Gtk 2.10]
gtk_print_operation_result_get_type [Gtk 2.10]	gtk_print_operation_run [Gtk 2.10]
gtk_print_operation_set_allow_async [Gtk 2.10]	gtk_print_operation_set_current_page [Gtk 2.10]
gtk_print_operation_set_custom_tab_label [Gtk 2.10]	gtk_print_operation_set_default_page_setup [Gtk 2.10]
gtk_print_operation_set_export_filename [Gtk 2.10]	gtk_print_operation_set_job_name [Gtk 2.10]
gtk_print_operation_set_n_pages [Gtk 2.10]	gtk_print_operation_set_print_settings [Gtk 2.10]
gtk_print_operation_set_show_progress [Gtk 2.10]	gtk_print_operation_set_track_print_status [Gtk 2.10]
gtk_print_operation_set_unit [Gtk 2.10]	gtk_print_operation_set_use_full_page [Gtk 2.10]
gtk_print_pages_get_type [Gtk 2.10]	gtk_print_quality_get_type [Gtk 2.10]
gtk_print_run_page_setup_dialog [Gtk 2.10]	gtk_print_run_page_setup_dialog_async [Gtk 2.10]
gtk_print_settings_copy [Gtk 2.10]	gtk_print_settings_foreach [Gtk 2.10]

gtk_print_settings_get [Gtk 2.10]	gtk_print_settings_get_bool [Gtk 2.10]
gtk_print_settings_get_collate [Gtk 2.10]	gtk_print_settings_get_default_source [Gtk 2.10]
gtk_print_settings_get_dither [Gtk 2.10]	gtk_print_settings_get_double [Gtk 2.10]
gtk_print_settings_get_double_with_default [Gtk 2.10]	gtk_print_settings_get_duplex [Gtk 2.10]
gtk_print_settings_get_finishings [Gtk 2.10]	gtk_print_settings_get_int [Gtk 2.10]
gtk_print_settings_get_int_with_default [Gtk 2.10]	gtk_print_settings_get_length [Gtk 2.10]
gtk_print_settings_get_media_type [Gtk 2.10]	gtk_print_settings_get_n_copies [Gtk 2.10]
gtk_print_settings_get_number_up [Gtk 2.10]	gtk_print_settings_get_orientation [Gtk 2.10]
gtk_print_settings_get_output_bin [Gtk 2.10]	gtk_print_settings_get_page_ranges [Gtk 2.10]
gtk_print_settings_get_page_set [Gtk 2.10]	gtk_print_settings_get_paper_height [Gtk 2.10]
gtk_print_settings_get_paper_size [Gtk 2.10]	gtk_print_settings_get_paper_width [Gtk 2.10]
gtk_print_settings_get_print_pages [Gtk 2.10]	gtk_print_settings_get_printer [Gtk 2.10]
gtk_print_settings_get_quality [Gtk 2.10]	gtk_print_settings_get_resolution [Gtk 2.10]
gtk_print_settings_get_reverse [Gtk 2.10]	gtk_print_settings_get_scale [Gtk 2.10]
gtk_print_settings_get_type [Gtk 2.10]	gtk_print_settings_get_use_color [Gtk 2.10]
gtk_print_settings_has_key [Gtk 2.10]	gtk_print_settings_new [Gtk 2.10]
gtk_print_settings_set [Gtk 2.10]	gtk_print_settings_set_bool [Gtk 2.10]
gtk_print_settings_set_collate [Gtk 2.10]	gtk_print_settings_set_default_source [Gtk 2.10]
gtk_print_settings_set_dither [Gtk 2.10]	gtk_print_settings_set_double [Gtk 2.10]
gtk_print_settings_set_duplex [Gtk 2.10]	gtk_print_settings_set_finishings [Gtk 2.10]
gtk_print_settings_set_int [Gtk 2.10]	gtk_print_settings_set_length [Gtk 2.10]

gtk_print_settings_set_media_type [Gtk 2.10]	gtk_print_settings_set_n_copies [Gtk 2.10]
gtk_print_settings_set_number_up [Gtk 2.10]	gtk_print_settings_set_orientation [Gtk 2.10]
gtk_print_settings_set_output_bin [Gtk 2.10]	gtk_print_settings_set_page_ranges [Gtk 2.10]
gtk_print_settings_set_page_set [Gtk 2.10]	gtk_print_settings_set_paper_height [Gtk 2.10]
gtk_print_settings_set_paper_size [Gtk 2.10]	gtk_print_settings_set_paper_width [Gtk 2.10]
gtk_print_settings_set_print_pages [Gtk 2.10]	gtk_print_settings_set_printer [Gtk 2.10]
gtk_print_settings_set_quality [Gtk 2.10]	gtk_print_settings_set_resolution [Gtk 2.10]
gtk_print_settings_set_reverse [Gtk 2.10]	gtk_print_settings_set_scale [Gtk 2.10]
gtk_print_settings_set_use_color [Gtk 2.10]	gtk_print_settings_unset [Gtk 2.10]
gtk_print_status_get_type [Gtk 2.10]	gtk_print_unix_dialog_add_custom_tab [Gtk 2.10]
gtk_print_unix_dialog_get_current_page [Gtk 2.10]	gtk_print_unix_dialog_get_page_set up [Gtk 2.10]
gtk_print_unix_dialog_get_selected_printer [Gtk 2.10]	gtk_print_unix_dialog_get_settings [Gtk 2.10]
gtk_print_unix_dialog_get_type [Gtk 2.10]	gtk_print_unix_dialog_new [Gtk 2.10]
gtk_print_unix_dialog_set_current_page [Gtk 2.10]	gtk_print_unix_dialog_set_manual_capabilities [Gtk 2.10]
gtk_print_unix_dialog_set_page_setup [Gtk 2.10]	gtk_print_unix_dialog_set_settings [Gtk 2.10]
gtk_printer_accepts_pdf [Gtk 2.10]	gtk_printer_accepts_ps [Gtk 2.10]
gtk_printer_compare [Gtk 2.10]	gtk_printer_get_backend [Gtk 2.10]
gtk_printer_get_description [Gtk 2.10]	gtk_printer_get_icon_name [Gtk 2.10]
gtk_printer_get_job_count [Gtk 2.10]	gtk_printer_get_location [Gtk 2.10]
gtk_printer_get_name [Gtk 2.10]	gtk_printer_get_state_message [Gtk 2.10]
gtk_printer_get_type [Gtk 2.10]	gtk_printer_is_active [Gtk 2.10]
gtk_printer_is_default [Gtk 2.10]	gtk_printer_is_virtual [Gtk 2.10]
gtk_printer_new [Gtk 2.10]	gtk_progress_bar_get_ellipsize [Gtk 2.10]

gtk_progress_bar_get_fraction [Gtk 2.10]	gtk_progress_bar_get_orientation [Gtk 2.10]
gtk_progress_bar_get_pulse_step [Gtk 2.10]	gtk_progress_bar_get_text [Gtk 2.10]
gtk_progress_bar_get_type [Gobject 2.32]	gtk_progress_bar_new [Gtk 2.10]
gtk_progress_bar_orientation_get_type [Gobject 2.32]	gtk_progress_bar_pulse [Gtk 2.10]
gtk_progress_bar_set_ellipsize [Gtk 2.10]	gtk_progress_bar_set_fraction [Gtk 2.10]
gtk_progress_bar_set_orientation [Gtk 2.10]	gtk_progress_bar_set_pulse_step [Gtk 2.10]
gtk_progress_bar_set_text [Gtk 2.10]	gtk_progress_bar_style_get_type [Gobject 2.32]
gtk_propagate_event [Gtk 2.10]	gtk_quit_add [Gtk 2.10]
gtk_quit_add_destroy [Gtk 2.10]	gtk_quit_add_full [Gtk 2.10]
gtk_quit_remove [Gtk 2.10]	gtk_quit_remove_by_data [Gtk 2.10]
gtk_radio_action_get_current_value [Gtk 2.10]	gtk_radio_action_get_group [Gtk 2.10]
gtk_radio_action_get_type [Gobject 2.32]	gtk_radio_action_new [Gtk 2.10]
gtk_radio_action_set_current_value [Gtk 2.10]	gtk_radio_action_set_group [Gtk 2.10]
gtk_radio_button_get_group [Gtk 2.10]	gtk_radio_button_get_type [Gobject 2.32]
gtk_radio_button_new [Gtk 2.10]	gtk_radio_button_new_from_widget [Gtk 2.10]
gtk_radio_button_new_with_label [Gtk 2.10]	gtk_radio_button_new_with_label_from_widget [Gtk 2.10]
gtk_radio_button_new_with_mnemonic [Gtk 2.10]	gtk_radio_button_new_with_mnemonic_from_widget [Gtk 2.10]
gtk_radio_button_set_group [Gtk 2.10]	gtk_radio_menu_item_get_group [Gtk 2.10]
gtk_radio_menu_item_get_type [Gobject 2.32]	gtk_radio_menu_item_new [Gtk 2.10]
gtk_radio_menu_item_new_from_widget [Gtk 2.10]	gtk_radio_menu_item_new_with_label [Gtk 2.10]
gtk_radio_menu_item_new_with_label_from_widget [Gtk 2.10]	gtk_radio_menu_item_new_with_mnemonic [Gtk 2.10]
gtk_radio_menu_item_new_with_mnemonic_from_widget [Gtk 2.10]	gtk_radio_menu_item_set_group [Gtk 2.10]

gtk_radio_tool_button_get_group [Gtk 2.10]	gtk_radio_tool_button_get_type [Gobject 2.32]
gtk_radio_tool_button_new [Gtk 2.10]	gtk_radio_tool_button_new_from_stock [Gtk 2.10]
gtk_radio_tool_button_new_from_widget [Gtk 2.10]	gtk_radio_tool_button_new_with_stock_from_widget [Gtk 2.10]
gtk_radio_tool_button_set_group [Gtk 2.10]	gtk_range_get_adjustment [Gtk 2.10]
gtk_range_get_inverted [Gtk 2.10]	gtk_range_get_lower_stepper_sensitivity [Gtk 2.10]
gtk_range_get_type [Gobject 2.32]	gtk_range_get_update_policy [Gtk 2.10]
gtk_range_get_upper_stepper_sensitivity [Gtk 2.10]	gtk_range_get_value [Gtk 2.10]
gtk_range_set_adjustment [Gtk 2.10]	gtk_range_set_increments [Gtk 2.10]
gtk_range_set_inverted [Gtk 2.10]	gtk_range_set_lower_stepper_sensitivity [Gtk 2.10]
gtk_range_set_range [Gtk 2.10]	gtk_range_set_update_policy [Gtk 2.10]
gtk_range_set_upper_stepper_sensitivity [Gtk 2.10]	gtk_range_set_value [Gtk 2.10]
gtk_rc_add_default_file [Gtk 2.10]	gtk_rc_find_module_in_path [Gtk 2.10]
gtk_rc_find_pixmap_in_path [Gtk 2.10]	gtk_rc_flags_get_type [Gobject 2.32]
gtk_rc_get_default_files [Gtk 2.10]	gtk_rc_get_im_module_file [Gtk 2.10]
gtk_rc_get_im_module_path [Gtk 2.10]	gtk_rc_get_module_dir [Gtk 2.10]
gtk_rc_get_style [Gtk 2.10]	gtk_rc_get_style_by_paths [Gtk 2.10]
gtk_rc_get_theme_dir [Gtk 2.10]	gtk_rc_parse [Gtk 2.10]
gtk_rc_parse_color [Gtk 2.10]	gtk_rc_parse_priority [Gtk 2.10]
gtk_rc_parse_state [Gtk 2.10]	gtk_rc_parse_string [Gtk 2.10]
gtk_rc_property_parse_border [Gtk 2.10]	gtk_rc_property_parse_color [Gtk 2.10]
gtk_rc_property_parse_enum [Gtk 2.10]	gtk_rc_property_parse_flags [Gtk 2.10]
gtk_rc_property_parse_requisition [Gtk 2.10]	gtk_rc_reparse_all [Gtk 2.10]
gtk_rc_reparse_all_for_settings [Gtk 2.10]	gtk_rc_reset_styles [Gtk 2.10]

gtk_rc_scanner_new [Gtk 2.10]	gtk_rc_set_default_files [Gtk 2.10]
gtk_rc_style_copy [Gtk 2.10]	gtk_rc_style_get_type [Gobject 2.32]
gtk_rc_style_new [Gtk 2.10]	gtk_rc_style_ref [Gtk 2.10]
gtk_rc_style_unref [Gtk 2.10]	gtk_rc_token_type_get_type [Gobject 2.32]
gtk_recent_chooser_add_filter [Gtk 2.10]	gtk_recent_chooser_dialog_get_type [Gtk 2.10]
gtk_recent_chooser_dialog_new [Gtk 2.10]	gtk_recent_chooser_dialog_new_for_manager [Gtk 2.10]
gtk_recent_chooser_error_get_type [Gtk 2.10]	gtk_recent_chooser_error_quark [Gtk 2.10]
gtk_recent_chooser_get_current_item [Gtk 2.10]	gtk_recent_chooser_get_current_uri [Gtk 2.10]
gtk_recent_chooser_get_filter [Gtk 2.10]	gtk_recent_chooser_get_items [Gtk 2.10]
gtk_recent_chooser_get_limit [Gtk 2.10]	gtk_recent_chooser_get_local_only [Gtk 2.10]
gtk_recent_chooser_get_select_multiple [Gtk 2.10]	gtk_recent_chooser_get_show_icons [Gtk 2.10]
gtk_recent_chooser_get_show_not_found [Gtk 2.10]	gtk_recent_chooser_get_show_private [Gtk 2.10]
gtk_recent_chooser_get_show_tips [Gtk 2.10]	gtk_recent_chooser_get_sort_type [Gtk 2.10]
gtk_recent_chooser_get_type [Gtk 2.10]	gtk_recent_chooser_get_uris [Gtk 2.10]
gtk_recent_chooser_list_filters [Gtk 2.10]	gtk_recent_chooser_menu_get_show_numbers [Gtk 2.10]
gtk_recent_chooser_menu_get_type [Gtk 2.10]	gtk_recent_chooser_menu_new [Gtk 2.10]
gtk_recent_chooser_menu_new_for_manager [Gtk 2.10]	gtk_recent_chooser_menu_set_show_numbers [Gtk 2.10]
gtk_recent_chooser_remove_filter [Gtk 2.10]	gtk_recent_chooser_select_all [Gtk 2.10]
gtk_recent_chooser_select_uri [Gtk 2.10]	gtk_recent_chooser_set_current_uri [Gtk 2.10]
gtk_recent_chooser_set_filter [Gtk 2.10]	gtk_recent_chooser_set_limit [Gtk 2.10]
gtk_recent_chooser_set_local_only [Gtk 2.10]	gtk_recent_chooser_set_select_multiple [Gtk 2.10]
gtk_recent_chooser_set_show_icons [Gtk 2.10]	gtk_recent_chooser_set_show_not_found [Gtk 2.10]

gtk_recent_chooser_set_show_private [Gtk 2.10]	gtk_recent_chooser_set_show_tips [Gtk 2.10]
gtk_recent_chooser_set_sort_func [Gtk 2.10]	gtk_recent_chooser_set_sort_type [Gtk 2.10]
gtk_recent_chooser_unselect_all [Gtk 2.10]	gtk_recent_chooser_unselect_uri [Gtk 2.10]
gtk_recent_chooser_widget_get_type [Gtk 2.10]	gtk_recent_chooser_widget_new [Gtk 2.10]
gtk_recent_chooser_widget_new_for_manager [Gtk 2.10]	gtk_recent_filter_add_age [Gtk 2.10]
gtk_recent_filter_add_application [Gtk 2.10]	gtk_recent_filter_add_custom [Gtk 2.10]
gtk_recent_filter_add_group [Gtk 2.10]	gtk_recent_filter_add_mime_type [Gtk 2.10]
gtk_recent_filter_add_pattern [Gtk 2.10]	gtk_recent_filter_add_pixbuf_formats [Gtk 2.10]
gtk_recent_filter_filter [Gtk 2.10]	gtk_recent_filter_flags_get_type [Gtk 2.10]
gtk_recent_filter_get_name [Gtk 2.10]	gtk_recent_filter_get_needed [Gtk 2.10]
gtk_recent_filter_get_type [Gtk 2.10]	gtk_recent_filter_new [Gtk 2.10]
gtk_recent_filter_set_name [Gtk 2.10]	gtk_recent_info_exists [Gtk 2.10]
gtk_recent_info_get_added [Gtk 2.10]	gtk_recent_info_get_age [Gtk 2.10]
gtk_recent_info_get_application_info [Gtk 2.10]	gtk_recent_info_get_applications [Gtk 2.10]
gtk_recent_info_get_description [Gtk 2.10]	gtk_recent_info_get_display_name [Gtk 2.10]
gtk_recent_info_get_groups [Gtk 2.10]	gtk_recent_info_get_icon [Gtk 2.10]
gtk_recent_info_get_mime_type [Gtk 2.10]	gtk_recent_info_get_modified [Gtk 2.10]
gtk_recent_info_get_private_hint [Gtk 2.10]	gtk_recent_info_get_short_name [Gtk 2.10]
gtk_recent_info_get_type [Gtk 2.10]	gtk_recent_info_get_uri [Gtk 2.10]
gtk_recent_info_get_uri_display [Gtk 2.10]	gtk_recent_info_get_visited [Gtk 2.10]
gtk_recent_info_has_application [Gtk 2.10]	gtk_recent_info_has_group [Gtk 2.10]
gtk_recent_info_is_local [Gtk 2.10]	gtk_recent_info_last_application [Gtk 2.10]
gtk_recent_info_match [Gtk 2.10]	gtk_recent_info_ref [Gtk 2.10]

gtk_recent_info_unref [Gtk 2.10]	gtk_recent_manager_add_full [Gtk 2.10]
gtk_recent_manager_add_item [Gtk 2.10]	gtk_recent_manager_error_get_type [Gtk 2.10]
gtk_recent_manager_error_quark [Gtk 2.10]	gtk_recent_manager_get_default [Gtk 2.10]
gtk_recent_manager_get_items [Gtk 2.10]	gtk_recent_manager_get_limit [Gtk 2.10]
gtk_recent_manager_get_type [Gtk 2.10]	gtk_recent_manager_has_item [Gtk 2.10]
gtk_recent_manager_lookup_item [Gtk 2.10]	gtk_recent_manager_move_item [Gtk 2.10]
gtk_recent_manager_new [Gtk 2.10]	gtk_recent_manager_purge_items [Gtk 2.10]
gtk_recent_manager_remove_item [Gtk 2.10]	gtk_recent_manager_set_limit [Gtk 2.10]
gtk_recent_sort_type_get_type [Gtk 2.10]	gtk_relief_style_get_type [Gobject 2.32]
gtk_requisition_copy [Gtk 2.10]	gtk_requisition_free [Gtk 2.10]
gtk_requisition_get_type [Gobject 2.32]	gtk_resize_mode_get_type [Gobject 2.32]
gtk_response_type_get_type [Gobject 2.32]	gtk_ruler_draw_pos [Gtk 2.10]
gtk_ruler_draw_ticks [Gtk 2.10]	gtk_ruler_get_metric [Gtk 2.10]
gtk_ruler_get_range [Gtk 2.10]	gtk_ruler_get_type [Gobject 2.32]
gtk_ruler_set_metric [Gtk 2.10]	gtk_ruler_set_range [Gtk 2.10]
gtk_scale_get_digits [Gtk 2.10]	gtk_scale_get_draw_value [Gtk 2.10]
gtk_scale_get_layout [Gtk 2.10]	gtk_scale_get_layout_offsets [Gtk 2.10]
gtk_scale_get_type [Gobject 2.32]	gtk_scale_get_value_pos [Gtk 2.10]
gtk_scale_set_digits [Gtk 2.10]	gtk_scale_set_draw_value [Gtk 2.10]
gtk_scale_set_value_pos [Gtk 2.10]	gtk_scroll_step_get_type [Gobject 2.32]
gtk_scroll_type_get_type [Gobject 2.32]	gtk_scrollbar_get_type [Gobject 2.32]
gtk_scrolled_window_add_with_viewport [Gtk 2.10]	gtk_scrolled_window_get_hadjustment [Gtk 2.10]
gtk_scrolled_window_get_hscrollbar [Gtk 2.10]	gtk_scrolled_window_get_placement [Gtk 2.10]
gtk_scrolled_window_get_policy [Gtk 2.10]	gtk_scrolled_window_get_shadow_type [Gtk 2.10]

gtk_scrolled_window_get_type [Gobject 2.32]	gtk_scrolled_window_get_vadjustment [Gtk 2.10]
gtk_scrolled_window_get_vscrollbar [Gtk 2.10]	gtk_scrolled_window_new [Gtk 2.10]
gtk_scrolled_window_set_hadjustment [Gtk 2.10]	gtk_scrolled_window_set_placement [Gtk 2.10]
gtk_scrolled_window_set_policy [Gtk 2.10]	gtk_scrolled_window_set_shadow_type [Gtk 2.10]
gtk_scrolled_window_set_vadjustment [Gtk 2.10]	gtk_scrolled_window_unset_placement [Gtk 2.10]
gtk_selection_add_target [Gtk 2.10]	gtk_selection_add_targets [Gtk 2.10]
gtk_selection_clear_targets [Gtk 2.10]	gtk_selection_convert [Gtk 2.10]
gtk_selection_data_copy [Gtk 2.10]	gtk_selection_data_free [Gtk 2.10]
gtk_selection_data_get_pixbuf [Gtk 2.10]	gtk_selection_data_get_targets [Gtk 2.10]
gtk_selection_data_get_text [Gtk 2.10]	gtk_selection_data_get_type [Gobject 2.32]
gtk_selection_data_get_uris [Gtk 2.10]	gtk_selection_data_set [Gtk 2.10]
gtk_selection_data_set_pixbuf [Gtk 2.10]	gtk_selection_data_set_text [Gtk 2.10]
gtk_selection_data_set_uris [Gtk 2.10]	gtk_selection_data_targets_include_image [Gtk 2.10]
gtk_selection_data_targets_include_rich_text [Gtk 2.10]	gtk_selection_data_targets_include_text [Gtk 2.10]
gtk_selection_data_targets_include_uri [Gtk 2.10]	gtk_selection_mode_get_type [Gobject 2.32]
gtk_selection_owner_set [Gtk 2.10]	gtk_selection_owner_set_for_display [Gtk 2.10]
gtk_selection_remove_all [Gtk 2.10]	gtk_sensitivity_type_get_type [Gtk 2.10]
gtk_separator_get_type [Gobject 2.32]	gtk_separator_menu_item_get_type [Gobject 2.32]
gtk_separator_menu_item_new [Gtk 2.10]	gtk_separator_tool_item_get_draw [Gtk 2.10]
gtk_separator_tool_item_get_type [Gobject 2.32]	gtk_separator_tool_item_new [Gtk 2.10]
gtk_separator_tool_item_set_draw [Gtk 2.10]	gtk_set_locale [Gtk 2.10]
gtk_settings_get_default [Gtk 2.10]	gtk_settings_get_for_screen [Gtk 2.10]

gtk_settings_get_type [Gobject 2.32]	gtk_settings_install_property [Gtk 2.10]
gtk_settings_install_property_parser [Gtk 2.10]	gtk_settings_set_double_property [Gtk 2.10]
gtk_settings_set_long_property [Gtk 2.10]	gtk_settings_set_property_value [Gtk 2.10]
gtk_settings_set_string_property [Gtk 2.10]	gtk_shadow_type_get_type [Gobject 2.32]
gtk_show_about_dialog [Gtk 2.10]	gtk_side_type_get_type [Gobject 2.32]
gtk_signal_run_type_get_type [Gobject 2.32]	gtk_size_group_add_widget [Gtk 2.10]
gtk_size_group_get_ignore_hidden [Gtk 2.10]	gtk_size_group_get_mode [Gtk 2.10]
gtk_size_group_get_type [Gobject 2.32]	gtk_size_group_get_widgets [Gtk 2.10]
gtk_size_group_mode_get_type [Gobject 2.32]	gtk_size_group_new [Gtk 2.10]
gtk_size_group_remove_widget [Gtk 2.10]	gtk_size_group_set_ignore_hidden [Gtk 2.10]
gtk_size_group_set_mode [Gtk 2.10]	gtk_socket_add_id [Gtk 2.10]
gtk_socket_get_id [Gtk 2.10]	gtk_socket_get_type [Gobject 2.32]
gtk_socket_new [Gtk 2.10]	gtk_sort_type_get_type [Gobject 2.32]
gtk_spin_button_configure [Gtk 2.10]	gtk_spin_button_get_adjustment [Gtk 2.10]
gtk_spin_button_get_digits [Gtk 2.10]	gtk_spin_button_get_increments [Gtk 2.10]
gtk_spin_button_get_numeric [Gtk 2.10]	gtk_spin_button_get_range [Gtk 2.10]
gtk_spin_button_get_snap_to_ticks [Gtk 2.10]	gtk_spin_button_get_type [Gobject 2.32]
gtk_spin_button_get_update_policy [Gtk 2.10]	gtk_spin_button_get_value [Gtk 2.10]
gtk_spin_button_get_value_as_int [Gtk 2.10]	gtk_spin_button_get_wrap [Gtk 2.10]
gtk_spin_button_new [Gtk 2.10]	gtk_spin_button_new_with_range [Gtk 2.10]
gtk_spin_button_set_adjustment [Gtk 2.10]	gtk_spin_button_set_digits [Gtk 2.10]
gtk_spin_button_set_increments [Gtk 2.10]	gtk_spin_button_set_numeric [Gtk 2.10]

gtk_spin_button_set_range [Gtk 2.10]	gtk_spin_button_set_snap_to_ticks [Gtk 2.10]
gtk_spin_button_set_update_policy [Gtk 2.10]	gtk_spin_button_set_value [Gtk 2.10]
gtk_spin_button_set_wrap [Gtk 2.10]	gtk_spin_button_spin [Gtk 2.10]
gtk_spin_button_update [Gtk 2.10]	gtk_spin_button_update_policy_get_type [Gobject 2.32]
gtk_spin_type_get_type [Gobject 2.32]	gtk_state_type_get_type [Gobject 2.32]
gtk_status_icon_get_blinking [Gtk 2.10]	gtk_status_icon_get_geometry [Gtk 2.10]
gtk_status_icon_get_icon_name [Gtk 2.10]	gtk_status_icon_get_pixbuf [Gtk 2.10]
gtk_status_icon_get_size [Gtk 2.10]	gtk_status_icon_get_stock [Gtk 2.10]
gtk_status_icon_get_storage_type [Gtk 2.10]	gtk_status_icon_get_type [Gtk 2.10]
gtk_status_icon_get_visible [Gtk 2.10]	gtk_status_icon_is_embedded [Gtk 2.10]
gtk_status_icon_new [Gtk 2.10]	gtk_status_icon_new_from_file [Gtk 2.10]
gtk_status_icon_new_from_icon_name [Gtk 2.10]	gtk_status_icon_new_from_pixbuf [Gtk 2.10]
gtk_status_icon_new_from_stock [Gtk 2.10]	gtk_status_icon_position_menu [Gtk 2.10]
gtk_status_icon_set_blinking [Gtk 2.10]	gtk_status_icon_set_from_file [Gtk 2.10]
gtk_status_icon_set_from_icon_name [Gtk 2.10]	gtk_status_icon_set_from_pixbuf [Gtk 2.10]
gtk_status_icon_set_from_stock [Gtk 2.10]	gtk_status_icon_set_tooltip [Gtk 2.10]
gtk_status_icon_set_visible [Gtk 2.10]	gtk_statusbar_get_context_id [Gtk 2.10]
gtk_statusbar_get_has_resize_grip [Gtk 2.10]	gtk_statusbar_get_type [Gobject 2.32]
gtk_statusbar_new [Gtk 2.10]	gtk_statusbar_pop [Gtk 2.10]
gtk_statusbar_push [Gtk 2.10]	gtk_statusbar_remove [Gtk 2.10]
gtk_statusbar_set_has_resize_grip [Gtk 2.10]	gtk_stock_add [Gtk 2.10]
gtk_stock_add_static [Gtk 2.10]	gtk_stock_item_copy [Gtk 2.10]
gtk_stock_item_free [Gtk 2.10]	gtk_stock_list_ids [Gtk 2.10]

gtk_stock_lookup [Gtk 2.10]	gtk_stock_set_translate_func [Gtk 2.10]
gtk_style_apply_default_background [Gtk 2.10]	gtk_style_attach [Gtk 2.10]
gtk_style_copy [Gtk 2.10]	gtk_style_detach [Gtk 2.10]
gtk_style_get_type [Gobject 2.32]	gtk_style_lookup_color [Gtk 2.10]
gtk_style_lookup_icon_set [Gtk 2.10]	gtk_style_new [Gtk 2.10]
gtk_style_render_icon [Gtk 2.10]	gtk_style_set_background [Gtk 2.10]
gtk_submenu_direction_get_type [Gobject 2.32]	gtk_submenu_placement_get_type [Gobject 2.32]
gtk_table_attach [Gtk 2.10]	gtk_table_attach_defaults [Gtk 2.10]
gtk_table_get_col_spacing [Gtk 2.10]	gtk_table_get_default_col_spacing [Gtk 2.10]
gtk_table_get_default_row_spacing [Gtk 2.10]	gtk_table_get_homogeneous [Gtk 2.10]
gtk_table_get_row_spacing [Gtk 2.10]	gtk_table_get_type [Gobject 2.32]
gtk_table_new [Gtk 2.10]	gtk_table_resize [Gtk 2.10]
gtk_table_set_col_spacing [Gtk 2.10]	gtk_table_set_col_spacings [Gtk 2.10]
gtk_table_set_homogeneous [Gtk 2.10]	gtk_table_set_row_spacing [Gtk 2.10]
gtk_table_set_row_spacings [Gtk 2.10]	gtk_target_flags_get_type [Gobject 2.32]
gtk_target_list_add [Gtk 2.10]	gtk_target_list_add_image_targets [Gtk 2.10]
gtk_target_list_add_rich_text_targets [Gtk 2.10]	gtk_target_list_add_table [Gtk 2.10]
gtk_target_list_add_text_targets [Gtk 2.10]	gtk_target_list_add_uri_targets [Gtk 2.10]
gtk_target_list_find [Gtk 2.10]	gtk_target_list_get_type [Gtk 2.10]
gtk_target_list_new [Gtk 2.10]	gtk_target_list_ref [Gtk 2.10]
gtk_target_list_remove [Gtk 2.10]	gtk_target_list_unref [Gtk 2.10]
gtk_target_table_free [Gtk 2.10]	gtk_target_table_new_from_list [Gtk 2.10]
gtk_targets_include_image [Gtk 2.10]	gtk_targets_include_rich_text [Gtk 2.10]
gtk_targets_include_text [Gtk 2.10]	gtk_targets_include_uri [Gtk 2.10]
gtk_tearoff_menu_item_get_type [Gobject 2.32]	gtk_tearoff_menu_item_new [Gtk 2.10]

gtk_text_attributes_copy [Gtk 2.10]	gtk_text_attributes_copy_values [Gtk 2.10]
gtk_text_attributes_get_type [Gobject 2.32]	gtk_text_attributes_new [Gtk 2.10]
gtk_text_attributes_ref [Gtk 2.10]	gtk_text_attributes_unref [Gtk 2.10]
gtk_text_buffer_add_selection_clipboard [Gtk 2.10]	gtk_text_buffer_apply_tag [Gtk 2.10]
gtk_text_buffer_apply_tag_by_name [Gtk 2.10]	gtk_text_buffer_backspace [Gtk 2.10]
gtk_text_buffer_begin_user_action [Gtk 2.10]	gtk_text_buffer_copy_clipboard [Gtk 2.10]
gtk_text_buffer_create_child_anchor [Gtk 2.10]	gtk_text_buffer_create_mark [Gtk 2.10]
gtk_text_buffer_create_tag [Gtk 2.10]	gtk_text_buffer_cut_clipboard [Gtk 2.10]
gtk_text_buffer_delete [Gtk 2.10]	gtk_text_buffer_delete_interactive [Gtk 2.10]
gtk_text_buffer_delete_mark [Gtk 2.10]	gtk_text_buffer_delete_mark_by_name [Gtk 2.10]
gtk_text_buffer_delete_selection [Gtk 2.10]	gtk_text_buffer_deserialize [Gtk 2.10]
gtk_text_buffer_deserialize_get_can_create_tags [Gtk 2.10]	gtk_text_buffer_deserialize_set_can_create_tags [Gtk 2.10]
gtk_text_buffer_end_user_action [Gtk 2.10]	gtk_text_buffer_get_bounds [Gtk 2.10]
gtk_text_buffer_get_char_count [Gtk 2.10]	gtk_text_buffer_get_copy_target_list [Gtk 2.10]
gtk_text_buffer_get_deserialize_formats [Gtk 2.10]	gtk_text_buffer_get_end_iter [Gtk 2.10]
gtk_text_buffer_get_has_selection [Gtk 2.10]	gtk_text_buffer_get_insert [Gtk 2.10]
gtk_text_buffer_get_iter_at_child_anchor [Gtk 2.10]	gtk_text_buffer_get_iter_at_line [Gtk 2.10]
gtk_text_buffer_get_iter_at_line_index [Gtk 2.10]	gtk_text_buffer_get_iter_at_line_offset [Gtk 2.10]
gtk_text_buffer_get_iter_at_mark [Gtk 2.10]	gtk_text_buffer_get_iter_at_offset [Gtk 2.10]
gtk_text_buffer_get_line_count [Gtk 2.10]	gtk_text_buffer_get_mark [Gtk 2.10]
gtk_text_buffer_get_modified [Gtk 2.10]	gtk_text_buffer_get_paste_target_list [Gtk 2.10]

gtk_text_buffer_get_selection_bound s [Gtk 2.10]	gtk_text_buffer_get_selection_bound s [Gtk 2.10]
gtk_text_buffer_get_serialize_format s [Gtk 2.10]	gtk_text_buffer_get_slice [Gtk 2.10]
gtk_text_buffer_get_start_iter [Gtk 2.10]	gtk_text_buffer_get_tag_table [Gtk 2.10]
gtk_text_buffer_get_text [Gtk 2.10]	gtk_text_buffer_get_type [Gobject 2.32]
gtk_text_buffer_insert [Gtk 2.10]	gtk_text_buffer_insert_at_cursor [Gtk 2.10]
gtk_text_buffer_insert_child_anchor [Gtk 2.10]	gtk_text_buffer_insert_interactive [Gtk 2.10]
gtk_text_buffer_insert_interactive_at _cursor [Gtk 2.10]	gtk_text_buffer_insert_pixbuf [Gtk 2.10]
gtk_text_buffer_insert_range [Gtk 2.10]	gtk_text_buffer_insert_range_interact ive [Gtk 2.10]
gtk_text_buffer_insert_with_tags [Gtk 2.10]	gtk_text_buffer_insert_with_tags_by _name [Gtk 2.10]
gtk_text_buffer_move_mark [Gtk 2.10]	gtk_text_buffer_move_mark_by_na me [Gtk 2.10]
gtk_text_buffer_new [Gtk 2.10]	gtk_text_buffer_paste_clipboard [Gtk 2.10]
gtk_text_buffer_place_cursor [Gtk 2.10]	gtk_text_buffer_register_deserialize_ format [Gtk 2.10]
gtk_text_buffer_register_deserialize_ tagset [Gtk 2.10]	gtk_text_buffer_register_serialize_for mat [Gtk 2.10]
gtk_text_buffer_register_serialize_tag set [Gtk 2.10]	gtk_text_buffer_remove_all_tags [Gtk 2.10]
gtk_text_buffer_remove_selection_cli pboard [Gtk 2.10]	gtk_text_buffer_remove_tag [Gtk 2.10]
gtk_text_buffer_remove_tag_by_nam e [Gtk 2.10]	gtk_text_buffer_select_range [Gtk 2.10]
gtk_text_buffer_serialize [Gtk 2.10]	gtk_text_buffer_set_modified [Gtk 2.10]
gtk_text_buffer_set_text [Gtk 2.10]	gtk_text_buffer_target_info_get_type [Gtk 2.10]
gtk_text_buffer_unregister_deserializ e_format [Gtk 2.10]	gtk_text_buffer_unregister_serialize_ format [Gtk 2.10]
gtk_text_child_anchor_get_deleted [Gtk 2.10]	gtk_text_child_anchor_get_type [Gobject 2.32]
gtk_text_child_anchor_get_widgets [Gtk 2.10]	gtk_text_child_anchor_new [Gtk 2.10]

gtk_text_direction_get_type [Gobject 2.32]	gtk_text_iter_backward_char [Gtk 2.10]
gtk_text_iter_backward_chars [Gtk 2.10]	gtk_text_iter_backward_cursor_position [Gtk 2.10]
gtk_text_iter_backward_cursor_positions [Gtk 2.10]	gtk_text_iter_backward_find_char [Gtk 2.10]
gtk_text_iter_backward_line [Gtk 2.10]	gtk_text_iter_backward_lines [Gtk 2.10]
gtk_text_iter_backward_search [Gtk 2.10]	gtk_text_iter_backward_sentence_start [Gtk 2.10]
gtk_text_iter_backward_sentence_starts [Gtk 2.10]	gtk_text_iter_backward_to_tag_toggle [Gtk 2.10]
gtk_text_iter_backward_visible_cursor_position [Gtk 2.10]	gtk_text_iter_backward_visible_cursor_positions [Gtk 2.10]
gtk_text_iter_backward_visible_line [Gtk 2.10]	gtk_text_iter_backward_visible_lines [Gtk 2.10]
gtk_text_iter_backward_visible_word_start [Gtk 2.10]	gtk_text_iter_backward_visible_word_starts [Gtk 2.10]
gtk_text_iter_backward_word_start [Gtk 2.10]	gtk_text_iter_backward_word_starts [Gtk 2.10]
gtk_text_iter_begins_tag [Gtk 2.10]	gtk_text_iter_can_insert [Gtk 2.10]
gtk_text_iter_compare [Gtk 2.10]	gtk_text_iter_copy [Gtk 2.10]
gtk_text_iter_editable [Gtk 2.10]	gtk_text_iter_ends_line [Gtk 2.10]
gtk_text_iter_ends_sentence [Gtk 2.10]	gtk_text_iter_ends_tag [Gtk 2.10]
gtk_text_iter_ends_word [Gtk 2.10]	gtk_text_iter_equal [Gtk 2.10]
gtk_text_iter_forward_char [Gtk 2.10]	gtk_text_iter_forward_chars [Gtk 2.10]
gtk_text_iter_forward_cursor_position [Gtk 2.10]	gtk_text_iter_forward_cursor_positions [Gtk 2.10]
gtk_text_iter_forward_find_char [Gtk 2.10]	gtk_text_iter_forward_line [Gtk 2.10]
gtk_text_iter_forward_lines [Gtk 2.10]	gtk_text_iter_forward_search [Gtk 2.10]
gtk_text_iter_forward_sentence_end [Gtk 2.10]	gtk_text_iter_forward_sentence_ends [Gtk 2.10]
gtk_text_iter_forward_to_end [Gtk 2.10]	gtk_text_iter_forward_to_line_end [Gtk 2.10]
gtk_text_iter_forward_to_tag_toggle [Gtk 2.10]	gtk_text_iter_forward_visible_cursor_position [Gtk 2.10]

gtk_text_iter_forward_visible_cursor_positions [Gtk 2.10]	gtk_text_iter_forward_visible_line [Gtk 2.10]
gtk_text_iter_forward_visible_lines [Gtk 2.10]	gtk_text_iter_forward_visible_word_end [Gtk 2.10]
gtk_text_iter_forward_visible_word_ends [Gtk 2.10]	gtk_text_iter_forward_word_end [Gtk 2.10]
gtk_text_iter_forward_word_ends [Gtk 2.10]	gtk_text_iter_free [Gtk 2.10]
gtk_text_iter_get_attributes [Gtk 2.10]	gtk_text_iter_get_buffer [Gtk 2.10]
gtk_text_iter_get_bytes_in_line [Gtk 2.10]	gtk_text_iter_get_char [Gtk 2.10]
gtk_text_iter_get_chars_in_line [Gtk 2.10]	gtk_text_iter_get_child_anchor [Gtk 2.10]
gtk_text_iter_get_language [Gtk 2.10]	gtk_text_iter_get_line [Gtk 2.10]
gtk_text_iter_get_line_index [Gtk 2.10]	gtk_text_iter_get_line_offset [Gtk 2.10]
gtk_text_iter_get_marks [Gtk 2.10]	gtk_text_iter_get_offset [Gtk 2.10]
gtk_text_iter_get_pixbuf [Gtk 2.10]	gtk_text_iter_get_slice [Gtk 2.10]
gtk_text_iter_get_tags [Gtk 2.10]	gtk_text_iter_get_text [Gtk 2.10]
gtk_text_iter_get_toggled_tags [Gtk 2.10]	gtk_text_iter_get_type [Gobject 2.32]
gtk_text_iter_get_visible_line_index [Gtk 2.10]	gtk_text_iter_get_visible_line_offset [Gtk 2.10]
gtk_text_iter_get_visible_slice [Gtk 2.10]	gtk_text_iter_get_visible_text [Gtk 2.10]
gtk_text_iter_has_tag [Gtk 2.10]	gtk_text_iter_in_range [Gtk 2.10]
gtk_text_iter_inside_sentence [Gtk 2.10]	gtk_text_iter_inside_word [Gtk 2.10]
gtk_text_iter_is_cursor_position [Gtk 2.10]	gtk_text_iter_is_end [Gtk 2.10]
gtk_text_iter_is_start [Gtk 2.10]	gtk_text_iter_order [Gtk 2.10]
gtk_text_iter_set_line [Gtk 2.10]	gtk_text_iter_set_line_index [Gtk 2.10]
gtk_text_iter_set_line_offset [Gtk 2.10]	gtk_text_iter_set_offset [Gtk 2.10]
gtk_text_iter_set_visible_line_index [Gtk 2.10]	gtk_text_iter_set_visible_line_offset [Gtk 2.10]
gtk_text_iter_starts_line [Gtk 2.10]	gtk_text_iter_starts_sentence [Gtk 2.10]
gtk_text_iter_starts_word [Gtk 2.10]	gtk_text_iter_toggles_tag [Gtk 2.10]

gtk_text_mark_get_buffer [Gtk 2.10]	gtk_text_mark_get_deleted [Gtk 2.10]
gtk_text_mark_get_left_gravity [Gtk 2.10]	gtk_text_mark_get_name [Gtk 2.10]
gtk_text_mark_get_type [Gobject 2.32]	gtk_text_mark_get_visible [Gtk 2.10]
gtk_text_mark_set_visible [Gtk 2.10]	gtk_text_search_flags_get_type [Gobject 2.32]
gtk_text_tag_event [Gtk 2.10]	gtk_text_tag_get_priority [Gtk 2.10]
gtk_text_tag_get_type [Gobject 2.32]	gtk_text_tag_new [Gtk 2.10]
gtk_text_tag_set_priority [Gtk 2.10]	gtk_text_tag_table_add [Gtk 2.10]
gtk_text_tag_table_foreach [Gtk 2.10]	gtk_text_tag_table_get_size [Gtk 2.10]
gtk_text_tag_table_get_type [Gobject 2.32]	gtk_text_tag_table_lookup [Gtk 2.10]
gtk_text_tag_table_new [Gtk 2.10]	gtk_text_tag_table_remove [Gtk 2.10]
gtk_text_view_add_child_at_anchor [Gtk 2.10]	gtk_text_view_add_child_in_window [Gtk 2.10]
gtk_text_view_backward_display_line [Gtk 2.10]	gtk_text_view_backward_display_line_start [Gtk 2.10]
gtk_text_view_buffer_to_window_coords [Gtk 2.10]	gtk_text_view_forward_display_line [Gtk 2.10]
gtk_text_view_forward_display_line_end [Gtk 2.10]	gtk_text_view_get_accepts_tab [Gtk 2.10]
gtk_text_view_get_border_window_size [Gtk 2.10]	gtk_text_view_get_buffer [Gtk 2.10]
gtk_text_view_get_cursor_visible [Gtk 2.10]	gtk_text_view_get_default_attributes [Gtk 2.10]
gtk_text_view_get_editable [Gtk 2.10]	gtk_text_view_get_indent [Gtk 2.10]
gtk_text_view_get_iter_at_location [Gtk 2.10]	gtk_text_view_get_iter_at_position [Gtk 2.10]
gtk_text_view_get_iter_location [Gtk 2.10]	gtk_text_view_get_justification [Gtk 2.10]
gtk_text_view_get_left_margin [Gtk 2.10]	gtk_text_view_get_line_at_y [Gtk 2.10]
gtk_text_view_get_line_yrange [Gtk 2.10]	gtk_text_view_get_overwrite [Gtk 2.10]
gtk_text_view_get_pixels_above_lines [Gtk 2.10]	gtk_text_view_get_pixels_below_lines [Gtk 2.10]
gtk_text_view_get_pixels_inside_wrap [Gtk 2.10]	gtk_text_view_get_right_margin [Gtk 2.10]

gtk_text_view_get_tabs [Gtk 2.10]	gtk_text_view_get_type [Gobject 2.32]
gtk_text_view_get_visible_rect [Gtk 2.10]	gtk_text_view_get_window [Gtk 2.10]
gtk_text_view_get_window_type [Gtk 2.10]	gtk_text_view_get_wrap_mode [Gtk 2.10]
gtk_text_view_move_child [Gtk 2.10]	gtk_text_view_move_mark_onscreen [Gtk 2.10]
gtk_text_view_move_visually [Gtk 2.10]	gtk_text_view_new [Gtk 2.10]
gtk_text_view_new_with_buffer [Gtk 2.10]	gtk_text_view_place_cursor_onscreen [Gtk 2.10]
gtk_text_view_scroll_mark_onscreen [Gtk 2.10]	gtk_text_view_scroll_to_iter [Gtk 2.10]
gtk_text_view_scroll_to_mark [Gtk 2.10]	gtk_text_view_set_accepts_tab [Gtk 2.10]
gtk_text_view_set_border_window_size [Gtk 2.10]	gtk_text_view_set_buffer [Gtk 2.10]
gtk_text_view_set_cursor_visible [Gtk 2.10]	gtk_text_view_set_editable [Gtk 2.10]
gtk_text_view_set_indent [Gtk 2.10]	gtk_text_view_set_justification [Gtk 2.10]
gtk_text_view_set_left_margin [Gtk 2.10]	gtk_text_view_set_overwrite [Gtk 2.10]
gtk_text_view_set_pixels_above_lines [Gtk 2.10]	gtk_text_view_set_pixels_below_lines [Gtk 2.10]
gtk_text_view_set_pixels_inside_wrap [Gtk 2.10]	gtk_text_view_set_right_margin [Gtk 2.10]
gtk_text_view_set_tabs [Gtk 2.10]	gtk_text_view_set_wrap_mode [Gtk 2.10]
gtk_text_view_starts_display_line [Gtk 2.10]	gtk_text_view_window_to_buffer_coords [Gtk 2.10]
gtk_text_window_type_get_type [Gobject 2.32]	gtk_toggle_action_get_active [Gtk 2.10]
gtk_toggle_action_get_draw_as_radio [Gtk 2.10]	gtk_toggle_action_get_type [Gobject 2.32]
gtk_toggle_action_new [Gtk 2.10]	gtk_toggle_action_set_active [Gtk 2.10]
gtk_toggle_action_set_draw_as_radio [Gtk 2.10]	gtk_toggle_action_toggled [Gtk 2.10]
gtk_toggle_button_get_active [Gtk 2.10]	gtk_toggle_button_get_inconsistent [Gtk 2.10]

gtk_toggle_button_get_mode [Gtk 2.10]	gtk_toggle_button_get_type [Gobject 2.32]
gtk_toggle_button_new [Gtk 2.10]	gtk_toggle_button_new_with_label [Gtk 2.10]
gtk_toggle_button_new_with_mnemonic [Gtk 2.10]	gtk_toggle_button_set_active [Gtk 2.10]
gtk_toggle_button_set_inconsistent [Gtk 2.10]	gtk_toggle_button_set_mode [Gtk 2.10]
gtk_toggle_button_toggled [Gtk 2.10]	gtk_toggle_tool_button_get_active [Gtk 2.10]
gtk_toggle_tool_button_get_type [Gobject 2.32]	gtk_toggle_tool_button_new [Gtk 2.10]
gtk_toggle_tool_button_new_from_stock [Gtk 2.10]	gtk_toggle_tool_button_set_active [Gtk 2.10]
gtk_tool_button_get_icon_name [Gtk 2.10]	gtk_tool_button_get_icon_widget [Gtk 2.10]
gtk_tool_button_get_label [Gtk 2.10]	gtk_tool_button_get_label_widget [Gtk 2.10]
gtk_tool_button_get_stock_id [Gtk 2.10]	gtk_tool_button_get_type [Gobject 2.32]
gtk_tool_button_get_use_underline [Gtk 2.10]	gtk_tool_button_new [Gtk 2.10]
gtk_tool_button_new_from_stock [Gtk 2.10]	gtk_tool_button_set_icon_name [Gtk 2.10]
gtk_tool_button_set_icon_widget [Gtk 2.10]	gtk_tool_button_set_label [Gtk 2.10]
gtk_tool_button_set_label_widget [Gtk 2.10]	gtk_tool_button_set_stock_id [Gtk 2.10]
gtk_tool_button_set_use_underline [Gtk 2.10]	gtk_tool_item_get_expand [Gtk 2.10]
gtk_tool_item_get_homogeneous [Gtk 2.10]	gtk_tool_item_get_icon_size [Gtk 2.10]
gtk_tool_item_get_is_important [Gtk 2.10]	gtk_tool_item_get_orientation [Gtk 2.10]
gtk_tool_item_get_proxy_menu_item [Gtk 2.10]	gtk_tool_item_get_relief_style [Gtk 2.10]
gtk_tool_item_get_toolbar_style [Gtk 2.10]	gtk_tool_item_get_type [Gobject 2.32]
gtk_tool_item_get_use_drag_window [Gtk 2.10]	gtk_tool_item_get_visible_horizontal [Gtk 2.10]
gtk_tool_item_get_visible_vertical [Gtk 2.10]	gtk_tool_item_new [Gtk 2.10]

gtk_tool_item_rebuild_menu [Gtk 2.10]	gtk_tool_item_retrieve_proxy_menu_item [Gtk 2.10]
gtk_tool_item_set_expand [Gtk 2.10]	gtk_tool_item_set_homogeneous [Gtk 2.10]
gtk_tool_item_set_is_important [Gtk 2.10]	gtk_tool_item_set_proxy_menu_item [Gtk 2.10]
gtk_tool_item_set_tooltip [Gtk 2.10]	gtk_tool_item_set_use_drag_window [Gtk 2.10]
gtk_tool_item_set_visible_horizontal [Gtk 2.10]	gtk_tool_item_set_visible_vertical [Gtk 2.10]
gtk_toolbar_child_type_get_type [Gobject 2.32]	gtk_toolbar_get_drop_index [Gtk 2.10]
gtk_toolbar_get_icon_size [Gtk 2.10]	gtk_toolbar_get_item_index [Gtk 2.10]
gtk_toolbar_get_n_items [Gtk 2.10]	gtk_toolbar_get_nth_item [Gtk 2.10]
gtk_toolbar_get_orientation [Gtk 2.10]	gtk_toolbar_get_relief_style [Gtk 2.10]
gtk_toolbar_get_show_arrow [Gtk 2.10]	gtk_toolbar_get_style [Gtk 2.10]
gtk_toolbar_get_tooltips [LSB]	gtk_toolbar_get_type [Gobject 2.32]
gtk_toolbar_insert [Gtk 2.10]	gtk_toolbar_new [Gtk 2.10]
gtk_toolbar_set_drop_highlight_item [Gtk 2.10]	gtk_toolbar_set_icon_size [Gtk 2.10]
gtk_toolbar_set_orientation [Gtk 2.10]	gtk_toolbar_set_show_arrow [Gtk 2.10]
gtk_toolbar_set_style [Gtk 2.10]	gtk_toolbar_set_tooltips [LSB]
gtk_toolbar_space_style_get_type [Gobject 2.32]	gtk_toolbar_style_get_type [Gobject 2.32]
gtk_toolbar_unset_style [Gtk 2.10]	gtk_tooltips_data_get [Gtk 2.10]
gtk_tooltips_disable [Gtk 2.10]	gtk_tooltips_enable [Gtk 2.10]
gtk_tooltips_force_window [Gtk 2.10]	gtk_tooltips_get_info_from_tip_window [Gtk 2.10]
gtk_tooltips_get_type [Gobject 2.32]	gtk_tooltips_new [Gtk 2.10]
gtk_tooltips_set_tip [Gtk 2.10]	gtk_tree_drag_dest_drag_data_received [Gtk 2.10]
gtk_tree_drag_dest_get_type [Gobject 2.32]	gtk_tree_drag_dest_row_drop_possible [Gtk 2.10]
gtk_tree_drag_source_drag_data_delete [Gtk 2.10]	gtk_tree_drag_source_drag_data_get [Gtk 2.10]
gtk_tree_drag_source_get_type [Gobject 2.32]	gtk_tree_drag_source_row_draggable [Gtk 2.10]

gtk_tree_get_row_drag_data [Gtk 2.10]	gtk_tree_iter_copy [Gtk 2.10]
gtk_tree_iter_free [Gtk 2.10]	gtk_tree_iter_get_type [Gobject 2.32]
gtk_tree_model_filter_clear_cache [Gtk 2.10]	gtk_tree_model_filter_convert_child_iter_to_iter [Gtk 2.10]
gtk_tree_model_filter_convert_child_path_to_path [Gtk 2.10]	gtk_tree_model_filter_convert_iter_to_child_iter [Gtk 2.10]
gtk_tree_model_filter_convert_path_to_child_path [Gtk 2.10]	gtk_tree_model_filter_get_model [Gtk 2.10]
gtk_tree_model_filter_get_type [Gobject 2.32]	gtk_tree_model_filter_new [Gtk 2.10]
gtk_tree_model_filter_refilter [Gtk 2.10]	gtk_tree_model_filter_set_modify_func [Gtk 2.10]
gtk_tree_model_filter_set_visible_column [Gtk 2.10]	gtk_tree_model_filter_set_visible_func [Gtk 2.10]
gtk_tree_model_flags_get_type [Gobject 2.32]	gtk_tree_model_foreach [Gtk 2.10]
gtk_tree_model_get [Gtk 2.10]	gtk_tree_model_get_column_type [Gtk 2.10]
gtk_tree_model_get_flags [Gtk 2.10]	gtk_tree_model_get_iter [Gtk 2.10]
gtk_tree_model_get_iter_first [Gtk 2.10]	gtk_tree_model_get_iter_from_string [Gtk 2.10]
gtk_tree_model_get_n_columns [Gtk 2.10]	gtk_tree_model_get_path [Gtk 2.10]
gtk_tree_model_get_string_from_iter [Gtk 2.10]	gtk_tree_model_get_type [Gobject 2.32]
gtk_tree_model_get_valist [Gtk 2.10]	gtk_tree_model_get_value [Gtk 2.10]
gtk_tree_model_iter_children [Gtk 2.10]	gtk_tree_model_iter_has_child [Gtk 2.10]
gtk_tree_model_iter_n_children [Gtk 2.10]	gtk_tree_model_iter_next [Gtk 2.10]
gtk_tree_model_iter_nth_child [Gtk 2.10]	gtk_tree_model_iter_parent [Gtk 2.10]
gtk_tree_model_ref_node [Gtk 2.10]	gtk_tree_model_row_changed [Gtk 2.10]
gtk_tree_model_row_deleted [Gtk 2.10]	gtk_tree_model_row_has_child_toggled [Gtk 2.10]
gtk_tree_model_row_inserted [Gtk 2.10]	gtk_tree_model_rows_reordered [Gtk 2.10]
gtk_tree_model_sort_clear_cache [Gtk 2.10]	gtk_tree_model_sort_convert_child_iter_to_iter [Gtk 2.10]

gtk_tree_model_sort_convert_child_path_to_path [Gtk 2.10]	gtk_tree_model_sort_convert_iter_to_child_iter [Gtk 2.10]
gtk_tree_model_sort_convert_path_to_child_path [Gtk 2.10]	gtk_tree_model_sort_get_model [Gtk 2.10]
gtk_tree_model_sort_get_type [Gobject 2.32]	gtk_tree_model_sort_iter_is_valid [Gtk 2.10]
gtk_tree_model_sort_new_with_model [Gtk 2.10]	gtk_tree_model_sort_reset_default_sort_func [Gtk 2.10]
gtk_tree_model_unref_node [Gtk 2.10]	gtk_tree_path_append_index [Gtk 2.10]
gtk_tree_path_compare [Gtk 2.10]	gtk_tree_path_copy [Gtk 2.10]
gtk_tree_path_down [Gtk 2.10]	gtk_tree_path_free [Gtk 2.10]
gtk_tree_path_get_depth [Gtk 2.10]	gtk_tree_path_get_indices [Gtk 2.10]
gtk_tree_path_get_type [Gobject 2.32]	gtk_tree_path_is_ancestor [Gtk 2.10]
gtk_tree_path_is_descendant [Gtk 2.10]	gtk_tree_path_new [Gtk 2.10]
gtk_tree_path_new_first [Gtk 2.10]	gtk_tree_path_new_from_indices [Gtk 2.10]
gtk_tree_path_new_from_string [Gtk 2.10]	gtk_tree_path_next [Gtk 2.10]
gtk_tree_path_prepend_index [Gtk 2.10]	gtk_tree_path_prev [Gtk 2.10]
gtk_tree_path_to_string [Gtk 2.10]	gtk_tree_path_up [Gtk 2.10]
gtk_tree_row_reference_copy [Gtk 2.10]	gtk_tree_row_reference_deleted [Gtk 2.10]
gtk_tree_row_reference_free [Gtk 2.10]	gtk_tree_row_reference_get_model [Gtk 2.10]
gtk_tree_row_reference_get_path [Gtk 2.10]	gtk_tree_row_reference_get_type [Gobject 2.32]
gtk_tree_row_reference_inserted [Gtk 2.10]	gtk_tree_row_reference_new [Gtk 2.10]
gtk_tree_row_reference_new_proxy [Gtk 2.10]	gtk_tree_row_reference_reordered [Gtk 2.10]
gtk_tree_row_reference_valid [Gtk 2.10]	gtk_tree_selection_count_selected_rows [Gtk 2.10]
gtk_tree_selection_get_mode [Gtk 2.10]	gtk_tree_selection_get_selected [Gtk 2.10]
gtk_tree_selection_get_selected_rows [Gtk 2.10]	gtk_tree_selection_get_tree_view [Gtk 2.10]

gtk_tree_selection_get_type [Gobject 2.32]	gtk_tree_selection_get_user_data [Gtk 2.10]
gtk_tree_selection_iter_is_selected [Gtk 2.10]	gtk_tree_selection_path_is_selected [Gtk 2.10]
gtk_tree_selection_select_all [Gtk 2.10]	gtk_tree_selection_select_iter [Gtk 2.10]
gtk_tree_selection_select_path [Gtk 2.10]	gtk_tree_selection_select_range [Gtk 2.10]
gtk_tree_selection_selected_foreach [Gtk 2.10]	gtk_tree_selection_set_mode [Gtk 2.10]
gtk_tree_selection_set_select_function [Gtk 2.10]	gtk_tree_selection_unselect_all [Gtk 2.10]
gtk_tree_selection_unselect_iter [Gtk 2.10]	gtk_tree_selection_unselect_path [Gtk 2.10]
gtk_tree_selection_unselect_range [Gtk 2.10]	gtk_tree_set_row_drag_data [Gtk 2.10]
gtk_tree_sortable_get_sort_column_id [Gtk 2.10]	gtk_tree_sortable_get_type [Gobject 2.32]
gtk_tree_sortable_has_default_sort_func [Gtk 2.10]	gtk_tree_sortable_set_default_sort_func [Gtk 2.10]
gtk_tree_sortable_set_sort_column_id [Gtk 2.10]	gtk_tree_sortable_set_sort_func [Gtk 2.10]
gtk_tree_sortable_sort_column_changed [Gtk 2.10]	gtk_tree_store_append [Gtk 2.10]
gtk_tree_store_clear [Gtk 2.10]	gtk_tree_store_get_type [Gobject 2.32]
gtk_tree_store_insert [Gtk 2.10]	gtk_tree_store_insert_after [Gtk 2.10]
gtk_tree_store_insert_before [Gtk 2.10]	gtk_tree_store_insert_with_values [Gtk 2.10]
gtk_tree_store_insert_with_valuesv [Gtk 2.10]	gtk_tree_store_is_ancestor [Gtk 2.10]
gtk_tree_store_iter_depth [Gtk 2.10]	gtk_tree_store_iter_is_valid [Gtk 2.10]
gtk_tree_store_move_after [Gtk 2.10]	gtk_tree_store_move_before [Gtk 2.10]
gtk_tree_store_new [Gtk 2.10]	gtk_tree_store_newv [Gtk 2.10]
gtk_tree_store_prepend [Gtk 2.10]	gtk_tree_store_remove [Gtk 2.10]
gtk_tree_store_reorder [Gtk 2.10]	gtk_tree_store_set [Gtk 2.10]
gtk_tree_store_set_column_types [Gtk 2.10]	gtk_tree_store_set_valist [Gtk 2.10]
gtk_tree_store_set_value [Gtk 2.10]	gtk_tree_store_swap [Gtk 2.10]

gtk_tree_view_append_column [Gtk 2.10]	gtk_tree_view_collapse_all [Gtk 2.10]
gtk_tree_view_collapse_row [Gtk 2.10]	gtk_tree_view_column_add_attribute [Gtk 2.10]
gtk_tree_view_column_cell_get_position [Gtk 2.10]	gtk_tree_view_column_cell_get_size [Gtk 2.10]
gtk_tree_view_column_cell_is_visible [Gtk 2.10]	gtk_tree_view_column_cell_set_cell_data [Gtk 2.10]
gtk_tree_view_column_clear [Gtk 2.10]	gtk_tree_view_column_clear_attributes [Gtk 2.10]
gtk_tree_view_column_clicked [Gtk 2.10]	gtk_tree_view_column_focus_cell [Gtk 2.10]
gtk_tree_view_column_get_alignment [Gtk 2.10]	gtk_tree_view_column_get_cell_renderers [Gtk 2.10]
gtk_tree_view_column_get_clickable [Gtk 2.10]	gtk_tree_view_column_get_expand [Gtk 2.10]
gtk_tree_view_column_get_fixed_width [Gtk 2.10]	gtk_tree_view_column_get_max_width [Gtk 2.10]
gtk_tree_view_column_get_min_width [Gtk 2.10]	gtk_tree_view_column_get_reorderable [Gtk 2.10]
gtk_tree_view_column_get_resizable [Gtk 2.10]	gtk_tree_view_column_get_sizing [Gtk 2.10]
gtk_tree_view_column_get_sort_column_id [Gtk 2.10]	gtk_tree_view_column_get_sort_indicator [Gtk 2.10]
gtk_tree_view_column_get_sort_order [Gtk 2.10]	gtk_tree_view_column_get_spacing [Gtk 2.10]
gtk_tree_view_column_get_title [Gtk 2.10]	gtk_tree_view_column_get_type [Gobject 2.32]
gtk_tree_view_column_get_visible [Gtk 2.10]	gtk_tree_view_column_get_widget [Gtk 2.10]
gtk_tree_view_column_get_width [Gtk 2.10]	gtk_tree_view_column_new [Gtk 2.10]
gtk_tree_view_column_new_with_attributes [Gtk 2.10]	gtk_tree_view_column_pack_end [Gtk 2.10]
gtk_tree_view_column_pack_start [Gtk 2.10]	gtk_tree_view_column_queue_resize [Gtk 2.10]
gtk_tree_view_column_set_alignment [Gtk 2.10]	gtk_tree_view_column_set_attributes [Gtk 2.10]
gtk_tree_view_column_set_cell_data_func [Gtk 2.10]	gtk_tree_view_column_set_clickable [Gtk 2.10]
gtk_tree_view_column_set_expand [Gtk 2.10]	gtk_tree_view_column_set_fixed_width [Gtk 2.10]

gtk_tree_view_column_set_max_width [Gtk 2.10]	gtk_tree_view_column_set_min_width [Gtk 2.10]
gtk_tree_view_column_set_reorderable [Gtk 2.10]	gtk_tree_view_column_set_resizable [Gtk 2.10]
gtk_tree_view_column_set_sizing [Gtk 2.10]	gtk_tree_view_column_set_sort_column_id [Gtk 2.10]
gtk_tree_view_column_set_sort_indicator [Gtk 2.10]	gtk_tree_view_column_set_sort_order [Gtk 2.10]
gtk_tree_view_column_set_spacing [Gtk 2.10]	gtk_tree_view_column_set_title [Gtk 2.10]
gtk_tree_view_column_set_visible [Gtk 2.10]	gtk_tree_view_column_set_widget [Gtk 2.10]
gtk_tree_view_column_sizing_get_type [Gobject 2.32]	gtk_tree_view_columns_autosize [Gtk 2.10]
gtk_tree_view_create_row_drag_icon [Gtk 2.10]	gtk_tree_view_drop_position_get_type [Gobject 2.32]
gtk_tree_view_enable_model_drag_dest [Gtk 2.10]	gtk_tree_view_enable_model_drag_source [Gtk 2.10]
gtk_tree_view_expand_all [Gtk 2.10]	gtk_tree_view_expand_row [Gtk 2.10]
gtk_tree_view_expand_to_path [Gtk 2.10]	gtk_tree_view_get_background_area [Gtk 2.10]
gtk_tree_view_get_bin_window [Gtk 2.10]	gtk_tree_view_get_cell_area [Gtk 2.10]
gtk_tree_view_get_column [Gtk 2.10]	gtk_tree_view_get_columns [Gtk 2.10]
gtk_tree_view_get_cursor [Gtk 2.10]	gtk_tree_view_get_dest_row_at_pos [Gtk 2.10]
gtk_tree_view_get_drag_dest_row [Gtk 2.10]	gtk_tree_view_get_enable_search [Gtk 2.10]
gtk_tree_view_get_enable_tree_lines [Gtk 2.10]	gtk_tree_view_get_expander_column [Gtk 2.10]
gtk_tree_view_get_fixed_height_mode [Gtk 2.10]	gtk_tree_view_get_grid_lines [Gtk 2.10]
gtk_tree_view_get_hadjustment [Gtk 2.10]	gtk_tree_view_get_headers_clickable [Gtk 2.10]
gtk_tree_view_get_headers_visible [Gtk 2.10]	gtk_tree_view_get_hover_expand [Gtk 2.10]
gtk_tree_view_get_hover_selection [Gtk 2.10]	gtk_tree_view_get_model [Gtk 2.10]
gtk_tree_view_get_path_at_pos [Gtk 2.10]	gtk_tree_view_get_reorderable [Gtk 2.10]

gtk_tree_view_get_row_separator_func [Gtk 2.10]	gtk_tree_view_get_rubber_banding [Gtk 2.10]
gtk_tree_view_get_rules_hint [Gtk 2.10]	gtk_tree_view_get_search_column [Gtk 2.10]
gtk_tree_view_get_search_entry [Gtk 2.10]	gtk_tree_view_get_search_equal_func [Gtk 2.10]
gtk_tree_view_get_search_position_func [Gtk 2.10]	gtk_tree_view_get_selection [Gtk 2.10]
gtk_tree_view_get_type [Gobject 2.32]	gtk_tree_view_get_vadjustment [Gtk 2.10]
gtk_tree_view_get_visible_range [Gtk 2.10]	gtk_tree_view_get_visible_rect [Gtk 2.10]
gtk_tree_view_grid_lines_get_type [Gtk 2.10]	gtk_tree_view_insert_column [Gtk 2.10]
gtk_tree_view_insert_column_with_attributes [Gtk 2.10]	gtk_tree_view_insert_column_with_data_func [Gtk 2.10]
gtk_tree_view_map_expanded_rows [Gtk 2.10]	gtk_tree_view_mode_get_type [Gobject 2.32]
gtk_tree_view_move_column_after [Gtk 2.10]	gtk_tree_view_new [Gtk 2.10]
gtk_tree_view_new_with_model [Gtk 2.10]	gtk_tree_view_remove_column [Gtk 2.10]
gtk_tree_view_row_activated [Gtk 2.10]	gtk_tree_view_row_expanded [Gtk 2.10]
gtk_tree_view_scroll_to_cell [Gtk 2.10]	gtk_tree_view_scroll_to_point [Gtk 2.10]
gtk_tree_view_set_column_drag_function [Gtk 2.10]	gtk_tree_view_set_cursor [Gtk 2.10]
gtk_tree_view_set_cursor_on_cell [Gtk 2.10]	gtk_tree_view_set_destroy_count_func [Gtk 2.10]
gtk_tree_view_set_drag_dest_row [Gtk 2.10]	gtk_tree_view_set_enable_search [Gtk 2.10]
gtk_tree_view_set_enable_tree_lines [Gtk 2.10]	gtk_tree_view_set_expander_column [Gtk 2.10]
gtk_tree_view_set_fixed_height_mode [Gtk 2.10]	gtk_tree_view_set_grid_lines [Gtk 2.10]
gtk_tree_view_set_hadjustment [Gtk 2.10]	gtk_tree_view_set_headers_clickable [Gtk 2.10]
gtk_tree_view_set_headers_visible [Gtk 2.10]	gtk_tree_view_set_hover_expand [Gtk 2.10]
gtk_tree_view_set_hover_selection [Gtk 2.10]	gtk_tree_view_set_model [Gtk 2.10]

gtk_tree_view_set_reorderable [Gtk 2.10]	gtk_tree_view_set_row_separator_func [Gtk 2.10]
gtk_tree_view_set_rubber_banding [Gtk 2.10]	gtk_tree_view_set_rules_hint [Gtk 2.10]
gtk_tree_view_set_search_column [Gtk 2.10]	gtk_tree_view_set_search_entry [Gtk 2.10]
gtk_tree_view_set_search_equal_func [Gtk 2.10]	gtk_tree_view_set_search_position_func [Gtk 2.10]
gtk_tree_view_set_vadjustment [Gtk 2.10]	gtk_tree_view_tree_to_widget_coords [Gtk 2.10]
gtk_tree_view_unset_rows_drag_dest [Gtk 2.10]	gtk_tree_view_unset_rows_drag_source [Gtk 2.10]
gtk_tree_view_widget_to_tree_coords [Gtk 2.10]	gtk_true [Gtk 2.10]
gtk_type_class [Gtk 2.10]	gtk_ui_manager_add_ui [Gtk 2.10]
gtk_ui_manager_add_ui_from_file [Gtk 2.10]	gtk_ui_manager_add_ui_from_string [Gtk 2.10]
gtk_ui_manager_ensure_update [Gtk 2.10]	gtk_ui_manager_get_accel_group [Gtk 2.10]
gtk_ui_manager_get_action [Gtk 2.10]	gtk_ui_manager_get_action_groups [Gtk 2.10]
gtk_ui_manager_get_add_tearoffs [Gtk 2.10]	gtk_ui_manager_get_toplevels [Gtk 2.10]
gtk_ui_manager_get_type [Gobject 2.32]	gtk_ui_manager_get_ui [Gtk 2.10]
gtk_ui_manager_get_widget [Gtk 2.10]	gtk_ui_manager_insert_action_group [Gtk 2.10]
gtk_ui_manager_item_type_get_type [Gobject 2.32]	gtk_ui_manager_new [Gtk 2.10]
gtk_ui_manager_new_merge_id [Gtk 2.10]	gtk_ui_manager_remove_action_group [Gtk 2.10]
gtk_ui_manager_remove_ui [Gtk 2.10]	gtk_ui_manager_set_add_tearoffs [Gtk 2.10]
gtk_unit_get_type [Gtk 2.10]	gtk_update_type_get_type [Gobject 2.32]
gtk_vbox_get_type [Gobject 2.32]	gtk_vbox_new [Gtk 2.10]
gtk_vbutton_box_get_type [Gobject 2.32]	gtk_vbutton_box_new [Gtk 2.10]
gtk_viewport_get_hadjustment [Gtk 2.10]	gtk_viewport_get_shadow_type [Gtk 2.10]
gtk_viewport_get_type [Gobject 2.32]	gtk_viewport_get_vadjustment [Gtk 2.10]

gtk_viewport_new [Gtk 2.10]	gtk_viewport_set_hadjustment [Gtk 2.10]
gtk_viewport_set_shadow_type [Gtk 2.10]	gtk_viewport_set_vadjustment [Gtk 2.10]
gtk_visibility_get_type [Gobject 2.32]	gtk_vpaned_get_type [Gobject 2.32]
gtk_vpaned_new [Gtk 2.10]	gtk_vruler_get_type [Gobject 2.32]
gtk_vruler_new [Gtk 2.10]	gtk_vscale_get_type [Gobject 2.32]
gtk_vscale_new [Gtk 2.10]	gtk_vscale_new_with_range [Gtk 2.10]
gtk_vscrollbar_get_type [Gobject 2.32]	gtk_vscrollbar_new [Gtk 2.10]
gtk_vseparator_get_type [Gobject 2.32]	gtk_vseparator_new [Gtk 2.10]
gtk_widget_activate [Gtk 2.10]	gtk_widget_add_accelerator [Gtk 2.10]
gtk_widget_add_events [Gtk 2.10]	gtk_widget_add_mnemonic_label [Gtk 2.10]
gtk_widget_can_activate_accel [Gtk 2.10]	gtk_widget_child_focus [Gtk 2.10]
gtk_widget_child_notify [Gtk 2.10]	gtk_widget_class_find_style_property [Gtk 2.10]
gtk_widget_class_install_style_property [Gtk 2.10]	gtk_widget_class_install_style_property_parser [Gtk 2.10]
gtk_widget_class_list_style_properties [Gtk 2.10]	gtk_widget_class_path [Gtk 2.10]
gtk_widget_create_pango_context [Gtk 2.10]	gtk_widget_create_pango_layout [Gtk 2.10]
gtk_widget_destroy [Gtk 2.10]	gtk_widget_destroyed [Gtk 2.10]
gtk_widget_ensure_style [Gtk 2.10]	gtk_widget_event [Gtk 2.10]
gtk_widget_flags_get_type [Gobject 2.32]	gtk_widget_freeze_child_notify [Gtk 2.10]
gtk_widget_get_accessible [Gtk 2.10]	gtk_widget_get_action [Gtk 2.10]
gtk_widget_get_ancestor [Gtk 2.10]	gtk_widget_get_child_requisition [Gtk 2.10]
gtk_widget_get_child_visible [Gtk 2.10]	gtk_widget_get_clipboard [Gtk 2.10]
gtk_widget_get_colormap [Gtk 2.10]	gtk_widget_get_composite_name [Gtk 2.10]
gtk_widget_get_default_colormap [Gtk 2.10]	gtk_widget_get_default_direction [Gtk 2.10]

gtk_widget_get_default_style [Gtk 2.10]	gtk_widget_get_default_visual [Gtk 2.10]
gtk_widget_get_direction [Gtk 2.10]	gtk_widget_get_display [Gtk 2.10]
gtk_widget_get_events [Gtk 2.10]	gtk_widget_get_extension_events [Gtk 2.10]
gtk_widget_get_modifier_style [Gtk 2.10]	gtk_widget_get_name [Gtk 2.10]
gtk_widget_get_no_show_all [Gtk 2.10]	gtk_widget_get_pango_context [Gtk 2.10]
gtk_widget_get_parent [Gtk 2.10]	gtk_widget_get_parent_window [Gtk 2.10]
gtk_widget_get_pointer [Gtk 2.10]	gtk_widget_get_root_window [Gtk 2.10]
gtk_widget_get_screen [Gtk 2.10]	gtk_widget_get_settings [Gtk 2.10]
gtk_widget_get_size_request [Gtk 2.10]	gtk_widget_get_style [Gtk 2.10]
gtk_widget_get_toplevel [Gtk 2.10]	gtk_widget_get_type [GObject 2.32]
gtk_widget_get_visual [Gtk 2.10]	gtk_widget_grab_default [Gtk 2.10]
gtk_widget_grab_focus [Gtk 2.10]	gtk_widget_has_screen [Gtk 2.10]
gtk_widget_help_type_get_type [GObject 2.32]	gtk_widget_hide [Gtk 2.10]
gtk_widget_hide_all [Gtk 2.10]	gtk_widget_hide_on_delete [Gtk 2.10]
gtk_widget_input_shape_combine_mask [Gtk 2.10]	gtk_widget_intersect [Gtk 2.10]
gtk_widget_is_ancestor [Gtk 2.10]	gtk_widget_is_composited [Gtk 2.10]
gtk_widget_is_focus [Gtk 2.10]	gtk_widget_list_accel_closures [Gtk 2.10]
gtk_widget_list_mnemonic_labels [Gtk 2.10]	gtk_widget_map [Gtk 2.10]
gtk_widget_mnemonic_activate [Gtk 2.10]	gtk_widget_modify_base [Gtk 2.10]
gtk_widget_modify_bg [Gtk 2.10]	gtk_widget_modify_fg [Gtk 2.10]
gtk_widget_modify_font [Gtk 2.10]	gtk_widget_modify_style [Gtk 2.10]
gtk_widget_modify_text [Gtk 2.10]	gtk_widget_new [Gtk 2.10]
gtk_widget_path [Gtk 2.10]	gtk_widget_pop_colormap [Gtk 2.10]
gtk_widget_pop_composite_child [Gtk 2.10]	gtk_widget_push_colormap [Gtk 2.10]
gtk_widget_push_composite_child [Gtk 2.10]	gtk_widget_queue_draw [Gtk 2.10]

gtk_widget_queue_draw_area [Gtk 2.10]	gtk_widget_queue_resize [Gtk 2.10]
gtk_widget_queue_resize_no_redraw [Gtk 2.10]	gtk_widget_realize [Gtk 2.10]
gtk_widget_ref [Gtk 2.10]	gtk_widget_region_intersect [Gtk 2.10]
gtk_widget_remove_accelerator [Gtk 2.10]	gtk_widget_remove_mnemonic_label [Gtk 2.10]
gtk_widget_render_icon [Gtk 2.10]	gtk_widget_reparent [Gtk 2.10]
gtk_widget_reset_rc_styles [Gtk 2.10]	gtk_widget_reset_shapes [Gtk 2.10]
gtk_widget_send_expose [Gtk 2.10]	gtk_widget_set_accel_path [Gtk 2.10]
gtk_widget_set_app_paintable [Gtk 2.10]	gtk_widget_set_child_visible [Gtk 2.10]
gtk_widget_set_colormap [Gtk 2.10]	gtk_widget_set_composite_name [Gtk 2.10]
gtk_widget_set_default_colormap [Gtk 2.10]	gtk_widget_set_default_direction [Gtk 2.10]
gtk_widget_set_direction [Gtk 2.10]	gtk_widget_set_double_buffered [Gtk 2.10]
gtk_widget_set_events [Gtk 2.10]	gtk_widget_set_extension_events [Gtk 2.10]
gtk_widget_set_name [Gtk 2.10]	gtk_widget_set_no_show_all [Gtk 2.10]
gtk_widget_set_parent [Gtk 2.10]	gtk_widget_set_parent_window [Gtk 2.10]
gtk_widget_set_redraw_on_allocate [Gtk 2.10]	gtk_widget_set_scroll_adjustments [Gtk 2.10]
gtk_widget_set_sensitive [Gtk 2.10]	gtk_widget_set_size_request [Gtk 2.10]
gtk_widget_set_state [Gtk 2.10]	gtk_widget_set_style [Gtk 2.10]
gtk_widget_shape_combine_mask [Gtk 2.10]	gtk_widget_show [Gtk 2.10]
gtk_widget_show_all [Gtk 2.10]	gtk_widget_show_now [Gtk 2.10]
gtk_widget_size_allocate [Gtk 2.10]	gtk_widget_size_request [Gtk 2.10]
gtk_widget_style_get [Gtk 2.10]	gtk_widget_style_get_property [Gtk 2.10]
gtk_widget_style_get_valist [Gtk 2.10]	gtk_widget_thaw_child_notify [Gtk 2.10]
gtk_widget_translate_coordinates [Gtk 2.10]	gtk_widget_unmap [Gtk 2.10]
gtk_widget_unparent [Gtk 2.10]	gtk_widget_unrealize [Gtk 2.10]

gtk_widget_unref [Gtk 2.10]	gtk_window_activate_default [Gtk 2.10]
gtk_window_activate_focus [Gtk 2.10]	gtk_window_activate_key [Gtk 2.10]
gtk_window_add_accel_group [Gtk 2.10]	gtk_window_add_mnemonic [Gtk 2.10]
gtk_window_begin_move_drag [Gtk 2.10]	gtk_window_begin_resize_drag [Gtk 2.10]
gtk_window_deiconify [Gtk 2.10]	gtk_window_fullscreen [Gtk 2.10]
gtk_window_get_accept_focus [Gtk 2.10]	gtk_window_get_decorated [Gtk 2.10]
gtk_window_get_default_icon_list [Gtk 2.10]	gtk_window_get_default_size [Gtk 2.10]
gtk_window_get_deletable [Gtk 2.10]	gtk_window_get_destroy_with_parent [Gtk 2.10]
gtk_window_get_focus [Gtk 2.10]	gtk_window_get_focus_on_map [Gtk 2.10]
gtk_window_get_frame_dimensions [Gtk 2.10]	gtk_window_get_gravity [Gtk 2.10]
gtk_window_get_group [Gtk 2.10]	gtk_window_get_has_frame [Gtk 2.10]
gtk_window_get_icon [Gtk 2.10]	gtk_window_get_icon_list [Gtk 2.10]
gtk_window_get_icon_name [Gtk 2.10]	gtk_window_get_mnemonic_modifier [Gtk 2.10]
gtk_window_get_modal [Gtk 2.10]	gtk_window_get_position [Gtk 2.10]
gtk_window_get_resizable [Gtk 2.10]	gtk_window_get_role [Gtk 2.10]
gtk_window_get_screen [Gtk 2.10]	gtk_window_get_size [Gtk 2.10]
gtk_window_get_skip_pager_hint [Gtk 2.10]	gtk_window_get_skip_taskbar_hint [Gtk 2.10]
gtk_window_get_title [Gtk 2.10]	gtk_window_get_transient_for [Gtk 2.10]
gtk_window_get_type [GObject 2.32]	gtk_window_get_type_hint [Gtk 2.10]
gtk_window_get_urgency_hint [Gtk 2.10]	gtk_window_group_add_window [Gtk 2.10]
gtk_window_group_get_type [GObject 2.32]	gtk_window_group_new [Gtk 2.10]
gtk_window_group_remove_window [Gtk 2.10]	gtk_window_has_toplevel_focus [Gtk 2.10]
gtk_window_iconify [Gtk 2.10]	gtk_window_is_active [Gtk 2.10]
gtk_window_list_toplevels [Gtk 2.10]	gtk_window_maximize [Gtk 2.10]

gtk_window_mnemonic_activate [Gtk 2.10]	gtk_window_move [Gtk 2.10]
gtk_window_new [Gtk 2.10]	gtk_window_parse_geometry [Gtk 2.10]
gtk_window_position_get_type [Gobject 2.32]	gtk_window_present [Gtk 2.10]
gtk_window_present_with_time [Gtk 2.10]	gtk_window_propagate_key_event [Gtk 2.10]
gtk_window_remove_accel_group [Gtk 2.10]	gtk_window_remove_mnemonic [Gtk 2.10]
gtk_window_reshow_with_initial_size [Gtk 2.10]	gtk_window_resize [Gtk 2.10]
gtk_window_set_accept_focus [Gtk 2.10]	gtk_window_set_auto_startup_notification [Gtk 2.10]
gtk_window_set_decorated [Gtk 2.10]	gtk_window_set_default [Gtk 2.10]
gtk_window_set_default_icon [Gtk 2.10]	gtk_window_set_default_icon_from_file [Gtk 2.10]
gtk_window_set_default_icon_list [Gtk 2.10]	gtk_window_set_default_icon_name [Gtk 2.10]
gtk_window_set_default_size [Gtk 2.10]	gtk_window_set_deletable [Gtk 2.10]
gtk_window_set_destroy_with_parent [Gtk 2.10]	gtk_window_set_focus [Gtk 2.10]
gtk_window_set_focus_on_map [Gtk 2.10]	gtk_window_set_frame_dimensions [Gtk 2.10]
gtk_window_set_geometry_hints [Gtk 2.10]	gtk_window_set_gravity [Gtk 2.10]
gtk_window_set_has_frame [Gtk 2.10]	gtk_window_set_icon [Gtk 2.10]
gtk_window_set_icon_from_file [Gtk 2.10]	gtk_window_set_icon_list [Gtk 2.10]
gtk_window_set_icon_name [Gtk 2.10]	gtk_window_set_keep_above [Gtk 2.10]
gtk_window_set_keep_below [Gtk 2.10]	gtk_window_set_mnemonic_modifier [Gtk 2.10]
gtk_window_set_modal [Gtk 2.10]	gtk_window_set_position [Gtk 2.10]
gtk_window_set_resizable [Gtk 2.10]	gtk_window_set_role [Gtk 2.10]
gtk_window_set_screen [Gtk 2.10]	gtk_window_set_skip_pager_hint [Gtk 2.10]
gtk_window_set_skip_taskbar_hint [Gtk 2.10]	gtk_window_set_title [Gtk 2.10]

gtk_window_set_transient_for [Gtk 2.10]	gtk_window_set_type_hint [Gtk 2.10]
gtk_window_set_urgency_hint [Gtk 2.10]	gtk_window_set_wmclass [Gtk 2.10]
gtk_window_stick [Gtk 2.10]	gtk_window_type_get_type [Gobject 2.32]
gtk_window_unfullscreen [Gtk 2.10]	gtk_window_unmaximize [Gtk 2.10]
gtk_window_unstick [Gtk 2.10]	gtk_wrap_mode_get_type [Gobject 2.32]

An LSB conforming implementation shall provide the generic deprecated functions for GTK main Widgets library specified in Table 17-124, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 17-124 libgtk-x11-2.0 - GTK main Widgets library Deprecated Function Interfaces

gtk_binding_parse_binding [Gtk 2.10]	gtk_file_selection_complete [Gtk 2.10]
gtk_file_selection_get_filename [Gtk 2.10]	gtk_file_selection_get_select_multiple [Gtk 2.10]
gtk_file_selection_get_selections [Gtk 2.10]	gtk_file_selection_hide_fileop_buttons [Gtk 2.10]
gtk_file_selection_new [Gtk 2.10]	gtk_file_selection_set_filename [Gtk 2.10]
gtk_file_selection_set_select_multiple [Gtk 2.10]	gtk_file_selection_show_fileop_buttons [Gtk 2.10]
gtk_menu_item_remove_submenu [Gtk 2.10]	gtk_rc_style_ref [Gtk 2.10]
gtk_rc_style_unref [Gtk 2.10]	gtk_toolbar_get_tooltips [LSB]
gtk_toolbar_set_tooltips [LSB]	gtk_tree_view_tree_to_widget_coords [Gtk 2.10]
gtk_tree_view_widget_to_tree_coords [Gtk 2.10]	gtk_widget_ref [Gtk 2.10]
gtk_widget_unref [Gtk 2.10]	

An LSB conforming implementation shall provide the generic data interfaces for GTK main Widgets library specified in Table 17-125, with the full mandatory functionality as described in the referenced underlying specification.

Table 17-125 libgtk-x11-2.0 - GTK main Widgets library Data Interfaces

gtk_binary_age [Gtk 2.10]	gtk_debug_flags [Gtk 2.10]
gtk_interface_age [Gtk 2.10]	gtk_major_version [Gtk 2.10]

gtk_micro_version [Gtk 2.10]

gtk_minor_version [Gtk 2.10]

17.32 Data Definitions for libgtk-x11-2.0

This section defines global identifiers and their values that are associated with interfaces contained in libgtk-x11-2.0. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

17.32.1 gtk-2.0/gtk/gtk.h

```
#define GTK_NOTE(type,action)
#define _GTK_ASSISTANT_H_
#define _GTK_PAGE_SETUP_H_
#define _GTK_PAPER_SIZE_H_
#define _GTK_PRINT_CONTEXT_H_
#define _GTK_PRINT_OPERATION_H_
#define _GTK_PRINT_SETTINGS_H_
#define _GTK_RECENT_CHOOSER_DIALOG_H_
#define _GTK_RECENT_CHOOSER_H_
#define _GTK_RECENT_CHOOSER_MENU_H_
#define _GTK_RECENT_CHOOSER_WIDGET_H_
#define GTK_OBJECT_FLOATING(obj) \
    ((GTK_OBJECT_FLAGS (obj) & GTK_FLOATING) != 0)
#define GTK_WIDGET_APP_PAINTABLE(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_APP_PAINTABLE) != 0)
#define GTK_WIDGET_CAN_DEFAULT(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_CAN_DEFAULT) != 0)
#define GTK_WIDGET_CAN_FOCUS(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_CAN_FOCUS) != 0)
#define GTK_WIDGET_COMPOSITE_CHILD(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_COMPOSITE_CHILD) != 0)
#define GTK_WIDGET_DOUBLE_BUFFERED(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_DOUBLE_BUFFERED) != 0)
#define GTK_WIDGET_HAS_DEFAULT(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_HAS_DEFAULT) != 0)
#define GTK_WIDGET_HAS_FOCUS(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_HAS_FOCUS) != 0)
#define GTK_WIDGET_HAS_GRAB(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_HAS_GRAB) != 0)
#define GTK_WIDGET_MAPPED(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_MAPPED) != 0)
#define GTK_WIDGET_NO_WINDOW(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_NO_WINDOW) != 0)
#define GTK_WIDGET_PARENT_SENSITIVE(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_PARENT_SENSITIVE) != 0)
#define GTK_WIDGET_RC_STYLE(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_RC_STYLE) != 0)
#define GTK_WIDGET_REALIZED(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_REALIZED) != 0)
```

```

#define GTK_WIDGET_RECEIVES_DEFAULT(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_RECEIVES_DEFAULT) != 0)
#define GTK_WIDGET_SENSITIVE(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_SENSITIVE) != 0)
#define GTK_WIDGET_TOPLEVEL(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_TOPLEVEL) != 0)
#define GTK_WIDGET_VISIBLE(wid) \
    ((GTK_WIDGET_FLAGS (wid) & GTK_VISIBLE) != 0)
#define GTK_TYPE_CALENDAR_DISPLAY_OPTIONS \
    (gtk_calendar_display_options_get_type())
#define GTK_TYPE_CELL_RENDERER_PIXBUF \
    (gtk_cell_renderer_pixbuf_get_type ())
#define GTK_TYPE_CELL_RENDERER_PROGRESS \
    (gtk_cell_renderer_progress_get_type ())
#define GTK_TYPE_CELL_RENDERER_TOGGLE \
    (gtk_cell_renderer_toggle_get_type ())
#define GTK_ICON_VIEW(obj) \
    (GTK_CHECK_CAST ((obj), GTK_TYPE_ICON_VIEW, GtkIconView))
#define GTK_OBJECT(object) \
    (GTK_CHECK_CAST ((object), GTK_TYPE_OBJECT, GObject))
#define GTK_ICON_VIEW_CLASS(klass) \
    (GTK_CHECK_CLASS_CAST ((klass), GTK_TYPE_ICON_VIEW, \
        GtkIconViewClass))
#define GTK_LIST_CLASS(klass) \
    (GTK_CHECK_CLASS_CAST ((klass), GTK_TYPE_LIST, \
        GtkListClass))
#define GTK_OBJECT_CLASS(klass) \
    (GTK_CHECK_CLASS_CAST ((klass), GTK_TYPE_OBJECT, \
        GObjectClass))
#define GTK_IS_ICON_VIEW_CLASS(klass) \
    (GTK_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ICON_VIEW))
#define GTK_IS_LIST_CLASS(klass) \
    (GTK_CHECK_CLASS_TYPE ((klass), GTK_TYPE_LIST))
#define GTK_IS_OBJECT_CLASS(klass) \
    (GTK_CHECK_CLASS_TYPE ((klass), GTK_TYPE_OBJECT))
#define GTK_ICON_VIEW_GET_CLASS(obj) \
    (GTK_CHECK_GET_CLASS ((obj), GTK_TYPE_ICON_VIEW, \
        GtkIconViewClass))
#define GTK_LIST_GET_CLASS(obj) \
    (GTK_CHECK_GET_CLASS ((obj), GTK_TYPE_LIST, GtkListClass))
#define GTK_OBJECT_GET_CLASS(object) \
    (GTK_CHECK_GET_CLASS ((object), GTK_TYPE_OBJECT, \
        GObjectClass))
#define GTK_TYPE_COLOR_SELECTION_DIALOG \
    (gtk_color_selection_dialog_get_type ())
#define GTK_TYPE_FONT_SELECTION_DIALOG \
    (gtk_font_selection_dialog_get_type ())
#define GTK_IS_RESIZE_CONTAINER(widget) \
    (GTK_IS_CONTAINER (widget) && ((GtkContainer*) (widget))-> \
        _resize_mode \
        != GTK_RESIZE_PARENT)
#define GTK_CHECK_VERSION(major,minor,micro) \
    (GTK_MAJOR_VERSION > (major) || (GTK_MAJOR_VERSION == (major) \
    && \
        GTK_MINOR_VERSION > (minor)) || (GTK_MAJOR_VERSION == (major) \
    && \
        GTK_MINOR_VERSION == (minor) && GTK_MICRO_VERSION >= \
        (micro)))
#define GTK_TYPE_PROGRESS_BAR_ORIENTATION \
    (gtk_progress_bar_orientation_get_type())
#define GTK_TYPE_SPIN_BUTTON_UPDATE_POLICY \
    (gtk_spin_button_update_policy_get_type())
#define GTK_TYPE_TREE_VIEW_COLUMN_SIZING \
    (gtk_tree_view_column_sizing_get_type())
#define GTK_TYPE_TREE_VIEW_DROP_POSITION \
    (gtk_tree_view_drop_position_get_type())

```

```

#define GTK_TYPE_UI_MANAGER_ITEM_TYPE \
    (gtk_ui_manager_item_type_get_type())
#define GTK_WIDGET_IS_SENSITIVE(wid) \
    (GTK_WIDGET_SENSITIVE (wid) && GTK_WIDGET_PARENT_SENSITIVE
(wid))
#define GTK_WIDGET_DRAWABLE(wid) \
    (GTK_WIDGET_VISIBLE (wid) && GTK_WIDGET_MAPPED (wid))
#define GTK_ABOUT_DIALOG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ABOUT_DIALOG, \
    GtkAboutDialogClass))
#define GTK_ACCEL_GROUP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ACCEL_GROUP, \
    GtkAccelGroupClass))
#define GTK_ACCEL_LABEL_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ACCEL_LABEL, \
    GtkAccelLabelClass))
#define GTK_ACCEL_MAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ACCEL_MAP, \
    GtkAccelMapClass))
#define GTK_ACCESSIBLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ACCESSIBLE, \
    GtkAccessibleClass))
#define GTK_ACTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ACTION, \
    GtkActionClass))
#define GTK_ADJUSTMENT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ADJUSTMENT, \
    GtkAdjustmentClass))
#define GTK_ALIGNMENT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ALIGNMENT, \
    GtkAlignmentClass))
#define GTK_ARROW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ARROW, \
    GtkArrowClass))
#define GTK_ASPECT_FRAME_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ASPECT_FRAME, \
    GtkAspectFrameClass))
#define GTK_BIN_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_BIN, \
    GtkBinClass))
#define GTK_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_BOX, \
    GtkBoxClass))
#define GTK_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_BUTTON, \
    GtkButtonClass))
#define GTK_BUTTON_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_BUTTON_BOX, \
    GtkButtonBoxClass))
#define GTK_CALENDAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CALENDAR, \
    GtkCalendarClass))
#define GTK_CELL_RENDERER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CELL_RENDERER, \
    \
    GtkCellRendererClass))
#define GTK_CELL_RENDERER_COMBO_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_CELL_RENDERER_COMBO, \
    GtkCellRendererComboClass))
#define GTK_CELL_RENDERER_PIXBUF_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_CELL_RENDERER_PIXBUF, \
    GtkCellRendererPixbufClass))
#define GTK_CELL_RENDERER_PROGRESS_CLASS(klass) \

```

```

        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_CELL_RENDERER_PROGRESS, \
        GtkCellRendererProgressClass))
#define GTK_CELL_RENDERER_TEXT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_CELL_RENDERER_TEXT, \
        GtkCellRendererTextClass))
#define GTK_CELL_RENDERER_TOGGLE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_CELL_RENDERER_TOGGLE, \
        GtkCellRendererToggleClass))
#define GTK_CHECK_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CHECK_BUTTON, \
        GtkCheckButtonClass))
#define GTK_CHECK_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CHECK_MENU_ITEM,
\
        GtkCheckMenuItemClass))
#define GTK_COLOR_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_COLOR_BUTTON, \
        GtkColorButtonClass))
#define GTK_COLOR_SELECTION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_COLOR_SELECTION,
\
        GtkColorSelectionClass))
#define GTK_COLOR_SELECTION_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_COLOR_SELECTION_DIALOG, \
        GtkColorSelectionDialogClass))
#define GTK_CONTAINER_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CONTAINER, \
        GtkContainerClass))
#define GTK_CURVE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CURVE,
        GtkCurveClass))
#define GTK_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_DIALOG,
        GtkDialogClass))
#define GTK_DRAWING_AREA_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_DRAWING_AREA, \
        GtkDrawingAreaClass))
#define GTK_ENTRY_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ENTRY,
        GtkEntryClass))
#define GTK_ENTRY_COMPLETION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_ENTRY_COMPLETION, \
        GtkEntryCompletionClass))
#define GTK_EVENT_BOX_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_EVENT_BOX, \
        GtkEventBoxClass))
#define GTK_EXPANDER_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_EXPANDER, \
        GtkExpanderClass))
#define GTK_FILE_CHOOSER_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_FILE_CHOOSER_BUTTON, \
        GtkFileChooserButtonClass))
#define GTK_FILE_CHOOSER_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_FILE_CHOOSER_DIALOG, \
        GtkFileChooserDialogClass))
#define GTK_FILE_CHOOSER_WIDGET_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_FILE_CHOOSER_WIDGET, \
        GtkFileChooserWidgetClass))

```



```

#define GTK_FILE_SELECTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_FILE_SELECTION, \
    \
    GtkFileSelectionClass))
#define GTK_FIXED_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_FIXED, \
    GtkFixedClass))
#define GTK_FONT_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_FONT_BUTTON, \
    GtkFontButtonClass))
#define GTK_FONT_SELECTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_FONT_SELECTION, \
    \
    GtkFontSelectionClass))
#define GTK_FONT_SELECTION_DIALOG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_FONT_SELECTION_DIALOG, \
    GtkFontSelectionDialogClass))
#define GTK_FRAME_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_FRAME, \
    GtkFrameClass))
#define GTK_GAMMA_CURVE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_GAMMA_CURVE, \
    GtkGammaCurveClass))
#define GTK_HANDLE_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HANDLE_BOX, \
    GtkHandleBoxClass))
#define GTK_HBOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HBOX, \
    GtkHBoxClass))
#define GTK_HBUTTON_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HBUTTON_BOX, \
    GtkHButtonBoxClass))
#define GTK_HPANED_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HPANED, \
    GtkHPanedClass))
#define GTK_HRULER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HRULER, \
    GtkHRulerClass))
#define GTK_HSCALE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HSCALE, \
    GtkHScaleClass))
#define GTK_HSCROLLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HSCROLLBAR, \
    GtkHScrollbarClass))
#define GTK_HSEPARATOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_HSEPARATOR, \
    GtkHSeparatorClass))
#define GTK_ICON_FACTORY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ICON_FACTORY, \
    GtkIconFactoryClass))
#define GTK_ICON_THEME_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ICON_THEME, \
    GtkIconThemeClass))
#define GTK_IMAGE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_IMAGE, \
    GtkImageClass))
#define GTK_IMAGE_MENU_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_IMAGE_MENU_ITEM, \
    \
    GtkImageMenuItemClass))
#define GTK_IM_CONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_IM_CONTEXT, \
    GtkIMContextClass))
#define GTK_IM_CONTEXT_SIMPLE_CLASS(klass) \
    \

```



```

        (G_TYPE_CHECK_CLASS_CAST ((klass),
GTK_TYPE_IM_CONTEXT_SIMPLE, \
        GtkIMContextSimpleClass))
#define GTK_IM_MULTICONTEXT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_IM_MULTICONTEXT, \
        \
        GtkIMMulticontextClass))
#define GTK_INPUT_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_INPUT_DIALOG, \
        GtkInputDialogClass))
#define GTK_INVISIBLE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_INVISIBLE, \
        GtkInvisibleClass))
#define GTK_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_ITEM, \
        GtkItemClass))
#define GTK_LABEL_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_LABEL, \
        GtkLabelClass))
#define GTK_LAYOUT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_LAYOUT, \
        GtkLayoutClass))
#define GTK_LIST_STORE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_LIST_STORE, \
        GtkListStoreClass))
#define GTK_MENU_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_MENU, \
        GtkMenuClass))
#define GTK_MENU_BAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_MENU_BAR, \
        GtkMenuBarClass))
#define GTK_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_MENU_ITEM, \
        GtkMenuItemClass))
#define GTK_MENU_SHELL_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_MENU_SHELL, \
        GtkMenuShellClass))
#define GTK_MESSAGE_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_MESSAGE_DIALOG, \
        \
        GtkMessageDialogClass))
#define GTK_MISC_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_MISC, \
        GtkMiscClass))
#define GTK_NOTEBOOK_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_NOTEBOOK, \
        GtkNotebookClass))
#define GTK_PANED_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_PANED, \
        GtkPanedClass))
#define GTK_PLUG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_PLUG, \
        GtkPlugClass))
#define GTK_PROGRESS_BAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_PROGRESS_BAR, \
        GtkProgressBarClass))
#define GTK_RADIO_ACTION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_RADIO_ACTION, \
        GtkRadioActionClass))
#define GTK_RADIO_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_RADIO_BUTTON, \
        GtkRadioButtonClass))
#define GTK_RADIO_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_RADIO_MENU_ITEM, \
        \
        GtkRadioMenuItemClass))

```

```

#define GTK_RADIO_TOOL_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_RADIO_TOOL_BUTTON, \
    GtkRadioToolButtonClass))
#define GTK_RANGE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GtkRangeClass), \
    GTK_TYPE_RANGE, \
    GtkRangeClass)
#define GTK_RC_STYLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_RC_STYLE, \
    GtkRcStyleClass))
#define GTK_RULER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_RULER, \
    GtkRulerClass))
#define GTK_SCALE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SCALE, \
    GtkScaleClass))
#define GTK_SCROLLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SCROLLBAR, \
    GtkScrollbarClass))
#define GTK_SCROLLED_WINDOW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SCROLLED_WINDOW, \
    GtkScrolledWindowClass))
#define GTK_SEPARATOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SEPARATOR, \
    GtkSeparatorClass))
#define GTK_SEPARATOR_MENU_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SEPARATOR_MENU_ITEM, \
    GtkSeparatorMenuItemClass))
#define GTK_SEPARATOR_TOOL_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SEPARATOR_TOOL_ITEM, \
    GtkSeparatorToolItemClass))
#define GTK_SETTINGS_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SETTINGS, \
    GtkSettingsClass))
#define GTK_SIZE_GROUP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SIZE_GROUP, \
    GtkSizeGroupClass))
#define GTK_SOCKET_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SOCKET, \
    GtkSocketClass))
#define GTK_SPIN_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_SPIN_BUTTON, \
    GtkSpinButtonClass))
#define GTK_STATUSBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_STATUSBAR, \
    GtkStatusbarClass))
#define GTK_STYLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_STYLE, \
    GtkStyleClass))
#define GTK_TABLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_TABLE, \
    GtkTableClass))
#define GTK_TEAROFF_MENU_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_TEAROFF_MENU_ITEM, \
    GtkTearoffMenuItemClass))
#define GTK_TEXT_BUFFER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_TEXT_BUFFER, \
    GtkTextBufferClass))
#define GTK_TEXT_CHILD_ANCHOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_TEXT_CHILD_ANCHOR, \
    GtkTextChildAnchorClass))

```

```

#define GTK_TEXT_MARK_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TEXT_MARK, \
    GtkTextMarkClass))
#define GTK_TEXT_TAG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TEXT_TAG, \
    GtkTextTagClass))
#define GTK_TEXT_TAG_TABLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TEXT_TAG_TABLE, \
    \
    GtkTextTagTableClass))
#define GTK_TEXT_VIEW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TEXT_VIEW, \
    GtkTextViewClass))
#define GTK_TOGGLE_ACTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TOGGLE_ACTION, \
    \
    GtkToggleActionClass))
#define GTK_TOGGLE_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TOGGLE_BUTTON, \
    \
    GtkToggleButtonClass))
#define GTK_TOGGLE_TOOL_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_TOGGLE_TOOL_BUTTON, \
    GtkToggleToolButtonClass))
#define GTK_TOOLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TOOLBAR, \
    GtkToolbarClass))
#define GTK_TOOLTIPS_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TOOLTIPS, \
    GtkToolTipsClass))
#define GTK_TOOL_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TOOL_BUTTON, \
    GtkToolButtonClass))
#define GTK_TOOL_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TOOL_ITEM, \
    GtkToolItemClass))
#define GTK_TREE_MODEL_SORT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TREE_MODEL_SORT, \
    \
    GtkTreeModelSortClass))
#define GTK_TREE_SELECTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TREE_SELECTION, \
    \
    GtkTreeSelectionClass))
#define GTK_TREE_STORE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TREE_STORE, \
    GtkTreeStoreClass))
#define GTK_TREE_VIEW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_TREE_VIEW, \
    GtkTreeViewClass))
#define GTK_TREE_VIEW_COLUMN_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), \
    GTK_TYPE_TREE_VIEW_COLUMN, \
    GtkTreeViewColumnClass))
#define GTK_UI_MANAGER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_UI_MANAGER, \
    GtkUIManagerClass))
#define GTK_VBOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VBOX, \
    GtkVBoxClass))
#define GTK_VBUTTON_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VBUTTON_BOX, \
    GtkVButtonBoxClass))
#define GTK_VIEWPORT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VIEWPORT, \

```

```

        GtkViewportClass))
#define GTK_VPANED_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VPANED,
    GtkVPanedClass))
#define GTK_VRULER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VRULER,
    GtkVRulerClass))
#define GTK_VSCALE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VSCALE,
    GtkVScaleClass))
#define GTK_VSCROLLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VSCROLLBAR, \
    GtkVScrollbarClass))
#define GTK_VSEPARATOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_VSEPARATOR, \
    GtkVSeparatorClass))
#define GTK_WIDGET_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_WIDGET,
    GtkWidgetClass))
#define GTK_WINDOW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_WINDOW,
    GtkWindowClass))
#define GTK_WINDOW_GROUP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_WINDOW_GROUP, \
    GtkWindowGroupClass))
#define GTK_CELL_EDITABLE_CLASS(obj) \
    (G_TYPE_CHECK_CLASS_CAST ((obj), GTK_TYPE_CELL_EDITABLE, \
    GtkCellEditableIface))
#define GTK_TREE_SORTABLE_CLASS(obj) \
    (G_TYPE_CHECK_CLASS_CAST ((obj), GTK_TYPE_TREE_SORTABLE, \
    GtkTreeSortableIface))
#define GTK_ACTION_GROUP_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), GTK_TYPE_ACTION_GROUP,
    \
    GtkActionGroupClass))
#define GTK_CELL_VIEW_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), GTK_TYPE_CELL_VIEW, \
    GtkCellViewClass))
#define GTK_COMBO_BOX_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), GTK_TYPE_COMBO_BOX, \
    GtkComboBoxClass))
#define GTK_COMBO_BOX_ENTRY_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), GTK_TYPE_COMBO_BOX_ENTRY, \
    GtkComboBoxEntryClass))
#define GTK_EDITABLE_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), GTK_TYPE_EDITABLE, \
    GtkEditableClass))
#define GTK_TREE_MODEL_FILTER_CLASS(vtable) \
    (G_TYPE_CHECK_CLASS_CAST ((vtable), GTK_TYPE_TREE_MODEL_FILTER, \
    GtkTreeModelFilterClass))
#define GTK_MENU_TOOL_BUTTON_CLASS(k) \
    (G_TYPE_CHECK_CLASS_CAST ((k), GTK_TYPE_MENU_TOOL_BUTTON, \
    GtkMenuToolButtonClass))
#define GTK_IS_MENU_TOOL_BUTTON_CLASS(k) \
    (G_TYPE_CHECK_CLASS_TYPE ((k), GTK_TYPE_MENU_TOOL_BUTTON))
#define GTK_IS_ABOUT_DIALOG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ABOUT_DIALOG))
#define GTK_IS_ACCEL_GROUP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ACCEL_GROUP))
#define GTK_IS_ACCEL_LABEL_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ACCEL_LABEL))
#define GTK_IS_ACCEL_MAP_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ACCEL_MAP))
#define GTK_IS_ACCESSIBLE_CLASS(klass) \

```

```

        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ACCESSIBLE))
#define GTK_IS_ACTION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ACTION))
#define GTK_IS_ADJUSTMENT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ADJUSTMENT))
#define GTK_IS_ALIGNMENT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ALIGNMENT))
#define GTK_IS_ARROW_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ARROW))
#define GTK_IS_ASPECT_FRAME_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ASPECT_FRAME))
#define GTK_IS_BIN_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_BIN))
#define GTK_IS_BOX_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_BOX))
#define GTK_IS_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_BUTTON))
#define GTK_IS_BUTTON_BOX_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_BUTTON_BOX))
#define GTK_IS_CALENDAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CALENDAR))
#define GTK_IS_CELL_RENDERER_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CELL_RENDERER))
#define GTK_IS_CELL_RENDERER_COMBO_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_CELL_RENDERER_COMBO))
#define GTK_IS_CELL_RENDERER_PIXBUF_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_CELL_RENDERER_PIXBUF))
#define GTK_IS_CELL_RENDERER_PROGRESS_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_CELL_RENDERER_PROGRESS))
#define GTK_IS_CELL_RENDERER_TEXT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_CELL_RENDERER_TEXT))
#define GTK_IS_CELL_RENDERER_TOGGLE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_CELL_RENDERER_TOGGLE))
#define GTK_IS_CHECK_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CHECK_BUTTON))
#define GTK_IS_CHECK_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_CHECK_MENU_ITEM))
#define GTK_IS_COLOR_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_COLOR_BUTTON))
#define GTK_IS_COLOR_SELECTION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_COLOR_SELECTION))
#define GTK_IS_COLOR_SELECTION_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_COLOR_SELECTION_DIALOG))
#define GTK_IS_CONTAINER_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CONTAINER))
#define GTK_IS_CURVE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CURVE))
#define GTK_IS_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_DIALOG))
#define GTK_IS_DRAWING_AREA_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_DRAWING_AREA))
#define GTK_IS_ENTRY_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ENTRY))
#define GTK_IS_ENTRY_COMPLETION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_ENTRY_COMPLETION))
#define GTK_IS_EVENT_BOX_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_EVENT_BOX))

```

```

#define GTK_IS_EXPANDER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_EXPANDER))
#define GTK_IS_FILE_CHOOSER_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_FILE_CHOOSER_BUTTON))
#define GTK_IS_FILE_CHOOSER_DIALOG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_FILE_CHOOSER_DIALOG))
#define GTK_IS_FILE_CHOOSER_WIDGET_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_FILE_CHOOSER_WIDGET))
#define GTK_IS_FILE_SELECTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_FILE_SELECTION))
#define GTK_IS_FIXED_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_FIXED))
#define GTK_IS_FONT_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_FONT_BUTTON))
#define GTK_IS_FONT_SELECTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_FONT_SELECTION))
#define GTK_IS_FONT_SELECTION_DIALOG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_FONT_SELECTION_DIALOG))
#define GTK_IS_FRAME_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_FRAME))
#define GTK_IS_GAMMA_CURVE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_GAMMA_CURVE))
#define GTK_IS_HANDLE_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HANDLE_BOX))
#define GTK_IS_HBOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HBOX))
#define GTK_IS_HBUTTON_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HBUTTON_BOX))
#define GTK_IS_HPANED_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HPANED))
#define GTK_IS_HRULER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HRULER))
#define GTK_IS_HSCALE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HSCALE))
#define GTK_IS_HSCROLLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HSCROLLBAR))
#define GTK_IS_HSEPARATOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_HSEPARATOR))
#define GTK_IS_ICON_FACTORY_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ICON_FACTORY))
#define GTK_IS_ICON_THEME_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ICON_THEME))
#define GTK_IS_IMAGE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_IMAGE))
#define GTK_IS_IMAGE_MENU_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_IMAGE_MENU_ITEM))
#define GTK_IS_IM_CONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_IM_CONTEXT))
#define GTK_IS_IM_CONTEXT_SIMPLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_IM_CONTEXT_SIMPLE))
#define GTK_IS_IM_MULTICONTEXT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), \
    GTK_TYPE_IM_MULTICONTEXT))
#define GTK_IS_INPUT_DIALOG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_INPUT_DIALOG))
#define GTK_IS_INVISIBLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_INVISIBLE))
#define GTK_IS_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_ITEM))
#define GTK_IS_LABEL_CLASS(klass) \

```



```

        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_LABEL))
#define GTK_IS_LAYOUT_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_LAYOUT))
#define GTK_IS_LIST_STORE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_LIST_STORE))
#define GTK_IS_MENU_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_MENU))
#define GTK_IS_MENU_BAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_MENU_BAR))
#define GTK_IS_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_MENU_ITEM))
#define GTK_IS_MENU_SHELL_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_MENU_SHELL))
#define GTK_IS_MESSAGE_DIALOG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_MESSAGE_DIALOG))
#define GTK_IS_MISC_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_MISC))
#define GTK_IS_NOTEBOOK_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_NOTEBOOK))
#define GTK_IS_PANED_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_PANED))
#define GTK_IS_PLUG_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_PLUG))
#define GTK_IS_PROGRESS_BAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_PROGRESS_BAR))
#define GTK_IS_RADIO_ACTION_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RADIO_ACTION))
#define GTK_IS_RADIO_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RADIO_BUTTON))
#define GTK_IS_RADIO_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_RADIO_MENU_ITEM))
#define GTK_IS_RADIO_TOOL_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_RADIO_TOOL_BUTTON))
#define GTK_IS_RANGE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RANGE))
#define GTK_IS_RC_STYLE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RC_STYLE))
#define GTK_IS_RULER_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RULER))
#define GTK_IS_SCALE_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SCALE))
#define GTK_IS_SCROLLBAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SCROLLBAR))
#define GTK_IS_SCROLLED_WINDOW_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_SCROLLED_WINDOW))
#define GTK_IS_SEPARATOR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SEPARATOR))
#define GTK_IS_SEPARATOR_MENU_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_SEPARATOR_MENU_ITEM))
#define GTK_IS_SEPARATOR_TOOL_ITEM_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_SEPARATOR_TOOL_ITEM))
#define GTK_IS_SETTINGS_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SETTINGS))
#define GTK_IS_SIZE_GROUP_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SIZE_GROUP))
#define GTK_IS_SOCKET_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SOCKET))
#define GTK_IS_SPIN_BUTTON_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_SPIN_BUTTON))
#define GTK_IS_STATUSBAR_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_STATUSBAR))

```

```

#define GTK_IS_STYLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_STYLE))
#define GTK_IS_TABLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TABLE))
#define GTK_IS_TEAROFF_MENU_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_TEAROFF_MENU_ITEM))
#define GTK_IS_TEXT_BUFFER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TEXT_BUFFER))
#define GTK_IS_TEXT_CHILD_ANCHOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_TEXT_CHILD_ANCHOR))
#define GTK_IS_TEXT_MARK_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TEXT_MARK))
#define GTK_IS_TEXT_TAG_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TEXT_TAG))
#define GTK_IS_TEXT_TAG_TABLE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TEXT_TAG_TABLE))
#define GTK_IS_TEXT_VIEW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TEXT_VIEW))
#define GTK_IS_TOGGLE_ACTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TOGGLE_ACTION))
#define GTK_IS_TOGGLE_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TOGGLE_BUTTON))
#define GTK_IS_TOGGLE_TOOL_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_TOGGLE_TOOL_BUTTON))
#define GTK_IS_TOOLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TOOLBAR))
#define GTK_IS_TOOLTIPS_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TOOLTIPS))
#define GTK_IS_TOOL_BUTTON_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TOOL_BUTTON))
#define GTK_IS_TOOL_ITEM_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TOOL_ITEM))
#define GTK_IS_TREE_MODEL_SORT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_TREE_MODEL_SORT))
#define GTK_IS_TREE_SELECTION_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TREE_SELECTION))
#define GTK_IS_TREE_STORE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TREE_STORE))
#define GTK_IS_TREE_VIEW_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_TREE_VIEW))
#define GTK_IS_TREE_VIEW_COLUMN_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass),
GTK_TYPE_TREE_VIEW_COLUMN))
#define GTK_IS_UI_MANAGER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_UI_MANAGER))
#define GTK_IS_VBOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VBOX))
#define GTK_IS_VBUTTON_BOX_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VBUTTON_BOX))
#define GTK_IS_VIEWPORT_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VIEWPORT))
#define GTK_IS_VPANED_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VPANED))
#define GTK_IS_VRULER_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VRULER))
#define GTK_IS_VSCALE_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VSCALE))
#define GTK_IS_VSCROLLBAR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VSCROLLBAR))
#define GTK_IS_VSEPARATOR_CLASS(klass) \
    (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_VSEPARATOR))
#define GTK_IS_WIDGET_CLASS(klass) \

```



```

        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_WIDGET))
#define GTK_IS_WINDOW_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_WINDOW))
#define GTK_IS_WINDOW_GROUP_CLASS(klass) \
        (G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_WINDOW_GROUP))
#define GTK_IS_ACTION_GROUP_CLASS(vtable) \
        (G_TYPE_CHECK_CLASS_TYPE ((vtable), GTK_TYPE_ACTION_GROUP))
#define GTK_IS_CELL_VIEW_CLASS(vtable) \
        (G_TYPE_CHECK_CLASS_TYPE ((vtable), GTK_TYPE_CELL_VIEW))
#define GTK_IS_COMBO_BOX_CLASS(vtable) \
        (G_TYPE_CHECK_CLASS_TYPE ((vtable), GTK_TYPE_COMBO_BOX))
#define GTK_IS_COMBO_BOX_ENTRY_CLASS(vtable) \
        (G_TYPE_CHECK_CLASS_TYPE ((vtable), \
GTK_TYPE_COMBO_BOX_ENTRY))
#define GTK_IS_EDITABLE_CLASS(vtable) \
        (G_TYPE_CHECK_CLASS_TYPE ((vtable), GTK_TYPE_EDITABLE))
#define GTK_IS_TREE_MODEL_FILTER_CLASS(vtable) \
        (G_TYPE_CHECK_CLASS_TYPE ((vtable), \
GTK_TYPE_TREE_MODEL_FILTER))
#define GTK_ACCEL_MAP(accel_map) \
        (G_TYPE_CHECK_INSTANCE_CAST ((accel_map), \
GTK_TYPE_ACCEL_MAP, \
GtkAccelMap))
#define GTK_MENU_TOOL_BUTTON(o) \
        (G_TYPE_CHECK_INSTANCE_CAST ((o), GTK_TYPE_MENU_TOOL_BUTTON, \
\
        GtkMenuToolButton))
#define GTK_TOOL_ITEM(o) \
        (G_TYPE_CHECK_INSTANCE_CAST ((o), GTK_TYPE_TOOL_ITEM, \
GtkToolItem))
#define GTK_ACCEL_LABEL(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ACCEL_LABEL, \
GtkAccelLabel))
#define GTK_ACCESSIBLE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ACCESSIBLE, \
GtkAccessible))
#define GTK_ACTION(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ACTION, \
GtkAction))
#define GTK_ACTION_GROUP(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ACTION_GROUP, \
\
        GtkActionGroup))
#define GTK_ADJUSTMENT(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ADJUSTMENT, \
GtkAdjustment))
#define GTK_ALIGNMENT(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ALIGNMENT, \
GtkAlignment))
#define GTK_ARROW(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ARROW, \
GtkArrow))
#define GTK_ASPECT_FRAME(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ASPECT_FRAME, \
\
        GtkAspectFrame))
#define GTK_BIN(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_BIN, GtkBin))
#define GTK_BOX(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_BOX, GtkBox))
#define GTK_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_BUTTON, \
GtkButton))
#define GTK_BUTTON_BOX(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_BUTTON_BOX, \
GtkButtonBox))

```

```

#define GTK_CALENDAR(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CALENDAR, \
    GtkCalendar))
#define GTK_CELL_EDITABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CELL_EDITABLE, \
    \
    GtkCellEditable))
#define GTK_CELL_LAYOUT(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CELL_LAYOUT, \
    GtkCellLayout))
#define GTK_CELL_RENDERER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CELL_RENDERER, \
    \
    GtkCellRenderer))
#define GTK_CELL_RENDERER_COMBO(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_CELL_RENDERER_COMBO, \
    GtkCellRendererCombo))
#define GTK_CELL_RENDERER_PIXBUF(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_CELL_RENDERER_PIXBUF, \
    GtkCellRendererPixbuf))
#define GTK_CELL_RENDERER_PROGRESS(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_CELL_RENDERER_PROGRESS, \
    GtkCellRendererProgress))
#define GTK_CELL_RENDERER_TEXT(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_CELL_RENDERER_TEXT, \
    GtkCellRendererText))
#define GTK_CELL_RENDERER_TOGGLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_CELL_RENDERER_TOGGLE, \
    GtkCellRendererToggle))
#define GTK_CELL_VIEW(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CELL_VIEW, \
    GtkCellView))
#define GTK_CHECK_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CHECK_BUTTON, \
    \
    GtkCheckButton))
#define GTK_CHECK_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_CHECK_MENU_ITEM, \
    GtkCheckMenuItem))
#define GTK_CLIPBOARD(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CLIPBOARD, \
    GtkClipboard))
#define GTK_COLOR_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_COLOR_BUTTON, \
    \
    GtkColorButton))
#define GTK_COLOR_SELECTION(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_COLOR_SELECTION, \
    GtkColorSelection))
#define GTK_COLOR_SELECTION_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_COLOR_SELECTION_DIALOG, \
    GtkColorSelectionDialog))
#define GTK_COMBO_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_COMBO_BOX, \
    GtkComboBox))
#define GTK_COMBO_BOX_ENTRY(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_COMBO_BOX_ENTRY, \

```

```

        GtkComboBoxEntry))
#define GTK_CONTAINER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CONTAINER, \
    GtkContainer))
#define GTK_CURVE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CURVE, \
    GtkCurve))
#define GTK_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_DIALOG, \
    GtkDialog))
#define GTK_DRAWING_AREA(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_DRAWING_AREA, \
    \
    GtkDrawingArea))
#define GTK_EDITABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_EDITABLE, \
    GtkEditable))
#define GTK_ENTRY(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ENTRY, \
    GtkEntry))
#define GTK_ENTRY_COMPLETION(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_ENTRY_COMPLETION, \
    GtkEntryCompletion))
#define GTK_EVENT_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_EVENT_BOX, \
    GtkEventBox))
#define GTK_EXPANDER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_EXPANDER, \
    GtkExpander))
#define GTK_FILE_CHOOSER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FILE_CHOOSER, \
    \
    GtkFileChooser))
#define GTK_FILE_CHOOSER_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_FILE_CHOOSER_DIALOG, \
    GtkFileChooserDialog))
#define GTK_FILE_CHOOSER_WIDGET(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_FILE_CHOOSER_WIDGET, \
    GtkFileChooserWidget))
#define GTK_FILE_FILTER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FILE_FILTER, \
    GtkFileFilter))
#define GTK_FILE_SELECTION(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FILE_SELECTION, \
    \
    GtkFileSelection))
#define GTK_FIXED(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FIXED, \
    GtkFixed))
#define GTK_FONT_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FONT_BUTTON, \
    GtkFontButton))
#define GTK_FONT_SELECTION(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FONT_SELECTION, \
    \
    GtkFontSelection))
#define GTK_FONT_SELECTION_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_FONT_SELECTION_DIALOG, \
    GtkFontSelectionDialog))
#define GTK_FRAME(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_FRAME, \
    GtkFrame))

```

```

#define GTK_GAMMA_CURVE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_GAMMA_CURVE, \
    GtkGammaCurve))
#define GTK_HANDLE_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HANDLE_BOX, \
    GtkHandleBox))
#define GTK_HBOX(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HBOX, GtkHBox))
#define GTK_HBUTTON_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HBUTTON_BOX, \
    GtkHButtonBox))
#define GTK_HPANED(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HPANED, \
    GtkHPaned))
#define GTK_HRULER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HRULER, \
    GtkHRuler))
#define GTK_HSCALE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HSCALE, \
    GtkHScale))
#define GTK_HSCROLLBAR(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HSCROLLBAR, \
    GtkHScrollbar))
#define GTK_HSEPARATOR(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_HSEPARATOR, \
    GtkHSeparator))
#define GTK_ICON_THEME(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ICON_THEME, \
    GtkIconTheme))
#define GTK_IMAGE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_IMAGE, \
    GtkImage))
#define GTK_IMAGE_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_IMAGE_MENU_ITEM, \
    GtkImageMenuItem))
#define GTK_IM_CONTEXT(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_IM_CONTEXT, \
    GtkIMContext))
#define GTK_IM_CONTEXT_SIMPLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_IM_CONTEXT_SIMPLE, \
    GtkIMContextSimple))
#define GTK_IM_MULTICONTEXT(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_IM_MULTICONTEXT, \
    GtkIMMulticontext))
#define GTK_INPUT_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_INPUT_DIALOG, \
    GtkInputDialog))
#define GTK_INVISIBLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_INVISIBLE, \
    GtkInvisible))
#define GTK_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_ITEM, GtkItem))
#define GTK_LABEL(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_LABEL, \
    GtkLabel))
#define GTK_LAYOUT(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_LAYOUT, \
    GtkLayout))
#define GTK_LIST_STORE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_LIST_STORE, \
    GtkListStore))
#define GTK_MENU(obj) \

```

```

        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_MENU, GtkMenu))
#define GTK_MENU_BAR(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_MENU_BAR,
GtkMenuBar))
#define GTK_MENU_ITEM(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_MENU_ITEM,
GtkMenuItem))
#define GTK_MENU_SHELL(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_MENU_SHELL, \
GtkMenuShell))
#define GTK_MESSAGE_DIALOG(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_MESSAGE_DIALOG,
\
        GtkMessageDialog))
#define GTK_MISC(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_MISC, GtkMisc))
#define GTK_NOTEBOOK(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_NOTEBOOK,
GtkNotebook))
#define GTK_PANED(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_PANED,
GtkPaned))
#define GTK_PLUG(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_PLUG, GtkPlug))
#define GTK_PROGRESS_BAR(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_PROGRESS_BAR,
\
        GtkProgressBar))
#define GTK_RADIO_ACTION(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RADIO_ACTION,
\
        GtkRadioAction))
#define GTK_RADIO_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RADIO_BUTTON,
\
        GtkRadioButton))
#define GTK_RADIO_MENU_ITEM(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RADIO_MENU_ITEM, \
        GtkRadioMenuItem))
#define GTK_RADIO_TOOL_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RADIO_TOOL_BUTTON, \
        GtkRadioToolButton))
#define GTK_RANGE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RANGE,
GtkRange))
#define GTK_RULER(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RULER,
GtkRuler))
#define GTK_SCALE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_SCALE,
GtkScale))
#define GTK_SCROLLBAR(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_SCROLLBAR, \
        GtkScrollbar))
#define GTK_SCROLLED_WINDOW(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_SCROLLED_WINDOW, \
        GtkScrolledWindow))
#define GTK_SEPARATOR(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_SEPARATOR, \
        GtkSeparator))
#define GTK_SEPARATOR_MENU_ITEM(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_SEPARATOR_MENU_ITEM, \
        GtkSeparatorMenuItem))

```

```

        GtkSeparatorMenuItem))
#define GTK_SEPARATOR_TOOL_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_SEPARATOR_TOOL_ITEM, \
    GtkSeparatorToolItem))
#define GTK_SETTINGS(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkSettings))
#define GTK_SIZE_GROUP(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkSizeGroup))
#define GTK_SOCKET(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkSocket))
#define GTK_SPIN_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkSpinButton))
#define GTK_STATUSBAR(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkStatusbar))
#define GTK_TABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkTable))
#define GTK_TEAROFF_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_TEAROFF_MENU_ITEM, \
    GtkTearoffMenuItem))
#define GTK_TEXT_BUFFER(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkTextBuffer))
#define GTK_TEXT_TAG(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkTextTag))
#define GTK_TEXT_TAG_TABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    \
    GtkTextTagTable))
#define GTK_TEXT_VIEW(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkTextView))
#define GTK_TOGGLE_ACTION(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    \
    GtkToggleAction))
#define GTK_TOGGLE_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    \
    GtkToggleButton))
#define GTK_TOGGLE_TOOL_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GTK_TYPE_TOGGLE_TOOL_BUTTON, \
    GtkToggleToolButton))
#define GTK_TOOLBAR(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkToolbar))
#define GTK_TOOLTIPS(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkTooltips))
#define GTK_TOOL_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    GtkToolButton))
#define GTK_TREE_DRAG_DEST(obj) \
    (G_TYPE_CHECK_INSTANCE_CAST ((obj), \
    \
    GtkTreeDragDest))
#define GTK_TREE_DRAG_SOURCE(obj) \

```

```

        (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_TREE_DRAG_SOURCE, \
        GtkTreeDragSource))
#define GTK_TREE_MODEL(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_TREE_MODEL, \
        GtkTreeModel))
#define GTK_TREE_MODEL_FILTER(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_TREE_MODEL_FILTER, \
        GtkTreeModelFilter))
#define GTK_TREE_MODEL_SORT(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_TREE_MODEL_SORT, \
        GtkTreeModelSort))
#define GTK_TREE_SELECTION(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_TREE_SELECTION, \
        GtkTreeSelection))
#define GTK_TREE_SORTABLE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_TREE_SORTABLE, \
        GtkTreeSortable))
#define GTK_TREE_STORE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_TREE_STORE, \
        GtkTreeStore))
#define GTK_TREE_VIEW(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_TREE_VIEW,
GtkTreeView))
#define GTK_TREE_VIEW_COLUMN(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_TREE_VIEW_COLUMN, \
        GtkTreeViewColumn))
#define GTK_UI_MANAGER(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_UI_MANAGER, \
        GtkUIManager))
#define GTK_VBOX(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VBOX, GtkVBox))
#define GTK_VBUTTON_BOX(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VBUTTON_BOX, \
        GtkVButtonBox))
#define GTK_VIEWPORT(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VIEWPORT,
GtkViewport))
#define GTK_VPANED(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VPANED,
GtkVPaned))
#define GTK_VRULER(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VRULER,
GtkVRuler))
#define GTK_VSCALE(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VSCALE,
GtkVScale))
#define GTK_VSCROLLBAR(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VSCROLLBAR, \
        GtkVScrollbar))
#define GTK_VSEPARATOR(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_VSEPARATOR, \
        GtkVSeparator))
#define GTK_WINDOW(obj) \
        (G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_WINDOW,
GtkWindow))
#define GTK_ABOUT_DIALOG(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object),
GTK_TYPE_ABOUT_DIALOG, \
        GtkAboutDialog))
#define GTK_ACCEL_GROUP(object) \

```



```

        (G_TYPE_CHECK_INSTANCE_CAST ((object), GTK_TYPE_ACCEL_GROUP,
\
        GtkAccelGroup))
#define GTK_FILE_CHOOSER_BUTTON(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object),
GTK_TYPE_FILE_CHOOSER_BUTTON, \
        GtkFileChooserButton))
#define GTK_ICON_FACTORY(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object),
GTK_TYPE_ICON_FACTORY, \
        GtkIconFactory))
#define GTK_RC_STYLE(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GTK_TYPE_RC_STYLE, \
        GtkRcStyle))
#define GTK_STYLE(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GTK_TYPE_STYLE,
GtkStyle))
#define GTK_TEXT_CHILD_ANCHOR(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object),
GTK_TYPE_TEXT_CHILD_ANCHOR, \
        GtkTextChildAnchor))
#define GTK_TEXT_MARK(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object), GTK_TYPE_TEXT_MARK,
\
        GtkTextMark))
#define GTK_WINDOW_GROUP(object) \
        (G_TYPE_CHECK_INSTANCE_CAST ((object),
GTK_TYPE_WINDOW_GROUP, \
        GtkWindowGroup))
#define GTK_WIDGET(widget) \
        (G_TYPE_CHECK_INSTANCE_CAST ((widget), GTK_TYPE_WIDGET,
GtkWidget))
#define GTK_IS_ACCEL_MAP(accel_map) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((accel_map),
GTK_TYPE_ACCEL_MAP))
#define GTK_IS_MENU_TOOL_BUTTON(o) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((o),
GTK_TYPE_MENU_TOOL_BUTTON))
#define GTK_IS_TOOL_ITEM(o) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((o), GTK_TYPE_TOOL_ITEM))
#define GTK_IS_ACCEL_LABEL(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ACCEL_LABEL))
#define GTK_IS_ACCESSIBLE(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ACCESSIBLE))
#define GTK_IS_ACTION(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ACTION))
#define GTK_IS_ACTION_GROUP(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ACTION_GROUP))
#define GTK_IS_ADJUSTMENT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ADJUSTMENT))
#define GTK_IS_ALIGNMENT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ALIGNMENT))
#define GTK_IS_ARROW(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ARROW))
#define GTK_IS_ASPECT_FRAME(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ASPECT_FRAME))
#define GTK_IS_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_BUTTON))
#define GTK_IS_BUTTON_BOX(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_BUTTON_BOX))
#define GTK_IS_CALENDAR(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CALENDAR))
#define GTK_IS_CELL_EDITABLE(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CELL_EDITABLE))
#define GTK_IS_CELL_LAYOUT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CELL_LAYOUT))

```



```

#define GTK_IS_CELL_RENDERER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CELL_RENDERER))
#define GTK_IS_CELL_RENDERER_COMBO(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_CELL_RENDERER_COMBO))
#define GTK_IS_CELL_RENDERER_PIXBUF(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_CELL_RENDERER_PIXBUF))
#define GTK_IS_CELL_RENDERER_PROGRESS(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_CELL_RENDERER_PROGRESS))
#define GTK_IS_CELL_RENDERER_TEXT(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_CELL_RENDERER_TEXT))
#define GTK_IS_CELL_RENDERER_TOGGLE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_CELL_RENDERER_TOGGLE))
#define GTK_IS_CELL_VIEW(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CELL_VIEW))
#define GTK_IS_CHECK_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CHECK_BUTTON))
#define GTK_IS_CHECK_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_CHECK_MENU_ITEM))
#define GTK_IS_CLIPBOARD(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CLIPBOARD))
#define GTK_IS_COLOR_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_COLOR_BUTTON))
#define GTK_IS_COLOR_SELECTION(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_COLOR_SELECTION))
#define GTK_IS_COLOR_SELECTION_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_COLOR_SELECTION_DIALOG))
#define GTK_IS_COMBO_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_COMBO_BOX))
#define GTK_IS_COMBO_BOX_ENTRY(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_COMBO_BOX_ENTRY))
#define GTK_IS_CONTAINER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CONTAINER))
#define GTK_IS_CURVE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CURVE))
#define GTK_IS_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_DIALOG))
#define GTK_IS_DRAWING_AREA(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_DRAWING_AREA))
#define GTK_IS_EDITABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_EDITABLE))
#define GTK_IS_ENTRY(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ENTRY))
#define GTK_IS_ENTRY_COMPLETION(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_ENTRY_COMPLETION))
#define GTK_IS_EVENT_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_EVENT_BOX))
#define GTK_IS_EXPANDER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_EXPANDER))
#define GTK_IS_FILE_CHOOSER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_FILE_CHOOSER))
#define GTK_IS_FILE_CHOOSER_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_FILE_CHOOSER_DIALOG))
#define GTK_IS_FILE_CHOOSER_WIDGET(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), \
    GTK_TYPE_FILE_CHOOSER_WIDGET))

```

```

#define GTK_IS_FILE_FILTER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_FILE_FILTER))
#define GTK_IS_FILE_SELECTION(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_FILE_SELECTION))
#define GTK_IS_FIXED(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_FIXED))
#define GTK_IS_FONT_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_FONT_BUTTON))
#define GTK_IS_FONT_SELECTION(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_FONT_SELECTION))
#define GTK_IS_FONT_SELECTION_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_FONT_SELECTION_DIALOG))
#define GTK_IS_FRAME(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_FRAME))
#define GTK_IS_GAMMA_CURVE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_GAMMA_CURVE))
#define GTK_IS_HANDLE_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HANDLE_BOX))
#define GTK_IS_HBOX(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HBOX))
#define GTK_IS_HBUTTON_BOX(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HBUTTON_BOX))
#define GTK_IS_HPANED(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HPANED))
#define GTK_IS_HRULER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HRULER))
#define GTK_IS_HSCALE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HSCALE))
#define GTK_IS_HSCROLLBAR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HSCROLLBAR))
#define GTK_IS_HSEPARATOR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_HSEPARATOR))
#define GTK_IS_ICON_THEME(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ICON_THEME))
#define GTK_IS_IMAGE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_IMAGE))
#define GTK_IS_IMAGE_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_IMAGE_MENU_ITEM))
#define GTK_IS_IM_CONTEXT(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_IM_CONTEXT))
#define GTK_IS_IM_CONTEXT_SIMPLE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_IM_CONTEXT_SIMPLE))
#define GTK_IS_IM_MULTICONTEXT(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_IM_MULTICONTEXT))
#define GTK_IS_INPUT_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_INPUT_DIALOG))
#define GTK_IS_INVISIBLE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_INVISIBLE))
#define GTK_IS_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_ITEM))
#define GTK_IS_LABEL(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_LABEL))
#define GTK_IS_LAYOUT(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_LAYOUT))
#define GTK_IS_LIST_STORE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_LIST_STORE))
#define GTK_IS_MENU(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_MENU))
#define GTK_IS_MENU_BAR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_MENU_BAR))

```

```

#define GTK_IS_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_MENU_ITEM))
#define GTK_IS_MENU_SHELL(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_MENU_SHELL))
#define GTK_IS_MESSAGE_DIALOG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_MESSAGE_DIALOG))
#define GTK_IS_MISC(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_MISC))
#define GTK_IS_NOTEBOOK(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_NOTEBOOK))
#define GTK_IS_PANED(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_PANED))
#define GTK_IS_PLUG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_PLUG))
#define GTK_IS_PROGRESS_BAR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_PROGRESS_BAR))
#define GTK_IS_RADIO_ACTION(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_RADIO_ACTION))
#define GTK_IS_RADIO_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_RADIO_BUTTON))
#define GTK_IS_RADIO_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_RADIO_MENU_ITEM))
#define GTK_IS_RADIO_TOOL_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_RADIO_TOOL_BUTTON))
#define GTK_IS_RANGE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_RANGE))
#define GTK_IS_RULER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_RULER))
#define GTK_IS_SCALE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SCALE))
#define GTK_IS_SCROLLBAR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SCROLLBAR))
#define GTK_IS_SCROLLED_WINDOW(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_SCROLLED_WINDOW))
#define GTK_IS_SEPARATOR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SEPARATOR))
#define GTK_IS_SEPARATOR_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_SEPARATOR_MENU_ITEM))
#define GTK_IS_SEPARATOR_TOOL_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_SEPARATOR_TOOL_ITEM))
#define GTK_IS_SETTINGS(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SETTINGS))
#define GTK_IS_SIZE_GROUP(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SIZE_GROUP))
#define GTK_IS_SOCKET(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SOCKET))
#define GTK_IS_SPIN_BUTTON(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_SPIN_BUTTON))
#define GTK_IS_STATUSBAR(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_STATUSBAR))
#define GTK_IS_TABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TABLE))
#define GTK_IS_TEAROFF_MENU_ITEM(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TEAROFF_MENU_ITEM))
#define GTK_IS_TEXT_BUFFER(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TEXT_BUFFER))
#define GTK_IS_TEXT_TAG(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TEXT_TAG))
#define GTK_IS_TEXT_TAG_TABLE(obj) \
    (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TEXT_TAG_TABLE))

```

```

        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TEXT_TAG_TABLE))
#define GTK_IS_TEXT_VIEW(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TEXT_VIEW))
#define GTK_IS_TOGGLE_ACTION(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TOGGLE_ACTION))
#define GTK_IS_TOGGLE_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TOGGLE_BUTTON))
#define GTK_IS_TOGGLE_TOOL_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TOGGLE_TOOL_BUTTON))
#define GTK_IS_TOOLBAR(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TOOLBAR))
#define GTK_IS_TOOLTIPS(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TOOLTIPS))
#define GTK_IS_TOOL_BUTTON(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TOOL_BUTTON))
#define GTK_IS_TREE_DRAG_DEST(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TREE_DRAG_DEST))
#define GTK_IS_TREE_DRAG_SOURCE(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TREE_DRAG_SOURCE))
#define GTK_IS_TREE_MODEL(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TREE_MODEL))
#define GTK_IS_TREE_MODEL_FILTER(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TREE_MODEL_FILTER))
#define GTK_IS_TREE_MODEL_SORT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TREE_MODEL_SORT))
#define GTK_IS_TREE_SELECTION(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TREE_SELECTION))
#define GTK_IS_TREE_SORTABLE(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TREE_SORTABLE))
#define GTK_IS_TREE_STORE(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TREE_STORE))
#define GTK_IS_TREE_VIEW(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_TREE_VIEW))
#define GTK_IS_TREE_VIEW_COLUMN(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_TREE_VIEW_COLUMN))
#define GTK_IS_UI_MANAGER(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_UI_MANAGER))
#define GTK_IS_VBOX(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VBOX))
#define GTK_IS_VBUTTON_BOX(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VBUTTON_BOX))
#define GTK_IS_VIEWPORT(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VIEWPORT))
#define GTK_IS_VPANED(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VPANED))
#define GTK_IS_VRULER(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VRULER))
#define GTK_IS_VSCALE(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VSCALE))
#define GTK_IS_VSCROLLBAR(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VSCROLLBAR))
#define GTK_IS_VSEPARATOR(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_VSEPARATOR))
#define GTK_IS_WINDOW(obj) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_WINDOW))
#define GTK_IS_ABOUT_DIALOG(object) \
        (G_TYPE_CHECK_INSTANCE_TYPE ((object),
GTK_TYPE_ABOUT_DIALOG))

```

```

#define GTK_IS_ACCEL_GROUP(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object),
GTK_TYPE_ACCEL_GROUP))
#define GTK_IS_FILE_CHOOSER_BUTTON(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object),
GTK_TYPE_FILE_CHOOSER_BUTTON))
#define GTK_IS_ICON_FACTORY(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object),
GTK_TYPE_ICON_FACTORY))
#define GTK_IS_RC_STYLE(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object), GTK_TYPE_RC_STYLE))
#define GTK_IS_STYLE(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object), GTK_TYPE_STYLE))
#define GTK_IS_TEXT_CHILD_ANCHOR(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object),
GTK_TYPE_TEXT_CHILD_ANCHOR))
#define GTK_IS_TEXT_MARK(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object), GTK_TYPE_TEXT_MARK))
#define GTK_IS_WINDOW_GROUP(object) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((object),
GTK_TYPE_WINDOW_GROUP))
#define GTK_IS_WIDGET(widget) \
    (G_TYPE_CHECK_INSTANCE_TYPE      ((widget), GTK_TYPE_WIDGET))
#define GTK_ACTION_GROUP_GET_CLASS(inst) \
    (G_TYPE_INSTANCE_GET_CLASS      ((inst), GTK_TYPE_ACTION_GROUP,
\
    GtkActionGroupClass))
#define GTK_CELL_VIEW_GET_CLASS(inst) \
    (G_TYPE_INSTANCE_GET_CLASS      ((inst), GTK_TYPE_CELL_VIEW, \
    GtkCellViewClass))
#define GTK_COMBO_BOX_GET_CLASS(inst) \
    (G_TYPE_INSTANCE_GET_CLASS      ((inst), GTK_TYPE_COMBO_BOX, \
    GtkComboBoxClass))
#define GTK_COMBO_BOX_ENTRY_GET_CLASS(inst) \
    (G_TYPE_INSTANCE_GET_CLASS      ((inst),
GTK_TYPE_COMBO_BOX_ENTRY, \
    GtkComboBoxEntryClass))
#define GTK_MENU_TOOL_BUTTON_GET_CLASS(o) \
    (G_TYPE_INSTANCE_GET_CLASS      ((o), GTK_TYPE_MENU_TOOL_BUTTON,
\
    GtkMenuToolButtonClass))
#define GTK_ABOUT_DIALOG_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ABOUT_DIALOG, \
    GtkAboutDialogClass))
#define GTK_ACCEL_GROUP_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ACCEL_GROUP, \
    GtkAccelGroupClass))
#define GTK_ACCEL_LABEL_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ACCEL_LABEL, \
    GtkAccelLabelClass))
#define GTK_ACCEL_MAP_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ACCEL_MAP, \
    GtkAccelMapClass))
#define GTK_ACCESSIBLE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ACCESSIBLE, \
    GtkAccessibleClass))
#define GTK_ADJUSTMENT_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ADJUSTMENT, \
    GtkAdjustmentClass))
#define GTK_ALIGNMENT_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ALIGNMENT, \
    GtkAlignmentClass))
#define GTK_ARROW_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS      ((obj), GTK_TYPE_ARROW,
    GtkArrowClass))
#define GTK_ASPECT_FRAME_GET_CLASS(obj) \

```

```

        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_ASPECT_FRAME, \
        GtkAspectFrameClass))
#define GTK_BIN_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_BIN, \
        GtkBinClass))
#define GTK_BOX_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_BOX, \
        GtkBoxClass))
#define GTK_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_BUTTON, \
        GtkButtonClass))
#define GTK_BUTTON_BOX_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_BUTTON_BOX, \
        GtkButtonBoxClass))
#define GTK_CALENDAR_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CALENDAR, \
        GtkCalendarClass))
#define GTK_CELL_RENDERER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CELL_RENDERER, \
        \
        GtkCellRendererClass))
#define GTK_CELL_RENDERER_COMBO_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_CELL_RENDERER_COMBO, \
        GtkCellRendererTextClass))
#define GTK_CELL_RENDERER_PIXBUF_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_CELL_RENDERER_PIXBUF, \
        GtkCellRendererPixbufClass))
#define GTK_CELL_RENDERER_PROGRESS_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_CELL_RENDERER_PROGRESS, \
        GtkCellRendererProgressClass))
#define GTK_CELL_RENDERER_TEXT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_CELL_RENDERER_TEXT, \
        GtkCellRendererTextClass))
#define GTK_CELL_RENDERER_TOGGLE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_CELL_RENDERER_TOGGLE, \
        GtkCellRendererToggleClass))
#define GTK_CHECK_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CHECK_BUTTON, \
        GtkCheckButtonClass))
#define GTK_CHECK_MENU_ITEM_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CHECK_MENU_ITEM, \
        \
        GtkCheckMenuItemClass))
#define GTK_COLOR_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_COLOR_BUTTON, \
        GtkColorButtonClass))
#define GTK_COLOR_SELECTION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_COLOR_SELECTION, \
        \
        GtkColorSelectionClass))
#define GTK_COLOR_SELECTION_DIALOG_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_COLOR_SELECTION_DIALOG, \
        GtkColorSelectionDialogClass))
#define GTK_CONTAINER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CONTAINER, \
        GtkContainerClass))
#define GTK_CURVE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CURVE, \
        GtkCurveClass))
#define GTK_DIALOG_GET_CLASS(obj) \

```

```

        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_DIALOG,
GtkDialogClass))
#define GTK_DRAWING_AREA_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_DRAWING_AREA, \
GtkDrawingAreaClass))
#define GTK_ENTRY_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_ENTRY,
GtkEntryClass))
#define GTK_ENTRY_COMPLETION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj),
GTK_TYPE_ENTRY_COMPLETION, \
GtkEntryCompletionClass))
#define GTK_EVENT_BOX_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_EVENT_BOX, \
GtkEventBoxClass))
#define GTK_EXPANDER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_EXPANDER, \
GtkExpanderClass))
#define GTK_FILE_CHOOSER_DIALOG_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj),
GTK_TYPE_FILE_CHOOSER_DIALOG, \
GtkFileChooserDialogClass))
#define GTK_FILE_CHOOSER_WIDGET_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj),
GTK_TYPE_FILE_CHOOSER_WIDGET, \
GtkFileChooserWidgetClass))
#define GTK_FILE_SELECTION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_FILE_SELECTION,
\
GtkFileSelectionClass))
#define GTK_FIXED_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_FIXED,
GtkFixedClass))
#define GTK_FONT_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_FONT_BUTTON, \
GtkFontButtonClass))
#define GTK_FONT_SELECTION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_FONT_SELECTION,
\
GtkFontSelectionClass))
#define GTK_FONT_SELECTION_DIALOG_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj),
GTK_TYPE_FONT_SELECTION_DIALOG, \
GtkFontSelectionDialogClass))
#define GTK_FRAME_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_FRAME,
GtkFrameClass))
#define GTK_GAMMA_CURVE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_GAMMA_CURVE, \
GtkGammaCurveClass))
#define GTK_HANDLE_BOX_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HANDLE_BOX, \
GtkHandleBoxClass))
#define GTK_HBOX_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HBOX,
GtkHBoxClass))
#define GTK_HBUTTON_BOX_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HBUTTON_BOX, \
GtkHButtonBoxClass))
#define GTK_HPANED_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HPANED,
GtkHPanedClass))
#define GTK_HRULER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HRULER,
GtkHRulerClass))
#define GTK_HSCALE_GET_CLASS(obj) \

```



```

        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HSCALE,
GtkHScaleClass))
#define GTK_HSCROLLBAR_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HSCROLLBAR, \
GtkHScrollbarClass))
#define GTK_HSEPARATOR_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_HSEPARATOR, \
GtkHSeparatorClass))
#define GTK_ICON_FACTORY_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_ICON_FACTORY, \
GtkIconFactoryClass))
#define GTK_ICON_THEME_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_ICON_THEME, \
GtkIconThemeClass))
#define GTK_IMAGE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_IMAGE, \
GtkImageClass))
#define GTK_IMAGE_MENU_ITEM_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_IMAGE_MENU_ITEM, \
\
        GtkImageMenuItemClass))
#define GTK_IM_CONTEXT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_IM_CONTEXT, \
GtkIMContextClass))
#define GTK_IM_CONTEXT_SIMPLE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_IM_CONTEXT_SIMPLE, \
\
        GtkIMContextSimpleClass))
#define GTK_IM_MULTICONTEXT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_IM_MULTICONTEXT, \
\
        GtkIMMulticontextClass))
#define GTK_INPUT_DIALOG_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_INPUT_DIALOG, \
GtkInputDialogClass))
#define GTK_INVISIBLE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_INVISIBLE, \
GtkInvisibleClass))
#define GTK_ITEM_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_ITEM, \
GtkItemClass))
#define GTK_LABEL_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_LABEL, \
GtkLabelClass))
#define GTK_LAYOUT_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_LAYOUT, \
GtkLayoutClass))
#define GTK_LIST_STORE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_LIST_STORE, \
GtkListStoreClass))
#define GTK_MENU_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_MENU, \
GtkMenuClass))
#define GTK_MENU_BAR_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_MENU_BAR, \
GtkMenuBarClass))
#define GTK_MENU_ITEM_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_MENU_ITEM, \
GtkMenuItemClass))
#define GTK_MENU_SHELL_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_MENU_SHELL, \
GtkMenuShellClass))
#define GTK_MESSAGE_DIALOG_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_MESSAGE_DIALOG, \
\
        GtkMessageDialogClass))

```



```

#define GTK_MISC_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_MISC, \
    GtkMiscClass))
#define GTK_NOTEBOOK_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_NOTEBOOK, \
    GtkNotebookClass))
#define GTK_PANED_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_PANED, \
    GtkPanedClass))
#define GTK_PLUG_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_PLUG, \
    GtkPlugClass))
#define GTK_PROGRESS_BAR_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_PROGRESS_BAR, \
    GtkProgressBarClass))
#define GTK_RADIO_BUTTON_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RADIO_BUTTON, \
    GtkRadioButtonClass))
#define GTK_RADIO_MENU_ITEM_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RADIO_MENU_ITEM, \
    \
    GtkRadioMenuItemClass))
#define GTK_RANGE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RANGE, \
    GtkRangeClass))
#define GTK_RC_STYLE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RC_STYLE, \
    GtkRcStyleClass))
#define GTK_RULER_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RULER, \
    GtkRulerClass))
#define GTK_SCALE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SCALE, \
    GtkScaleClass))
#define GTK_SCROLLBAR_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SCROLLBAR, \
    GtkScrollbarClass))
#define GTK_SCROLLED_WINDOW_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SCROLLED_WINDOW, \
    \
    GtkScrolledWindowClass))
#define GTK_SEPARATOR_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SEPARATOR, \
    GtkSeparatorClass))
#define GTK_SEPARATOR_MENU_ITEM_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), \
    GTK_TYPE_SEPARATOR_MENU_ITEM, \
    GtkSeparatorMenuItemClass))
#define GTK_SETTINGS_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SETTINGS, \
    GtkSettingsClass))
#define GTK_SIZE_GROUP_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SIZE_GROUP, \
    GtkSizeGroupClass))
#define GTK_SOCKET_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SOCKET, \
    GtkSocketClass))
#define GTK_SPIN_BUTTON_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_SPIN_BUTTON, \
    GtkSpinButtonClass))
#define GTK_STATUSBAR_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_STATUSBAR, \
    GtkStatusbarClass))
#define GTK_STYLE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_STYLE, \
    GtkStyleClass))

```

```

#define GTK_TABLE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TABLE, \
    GtkTableClass))
#define GTK_TEAROFF_MENU_ITEM_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), \
    GTK_TYPE_TEAROFF_MENU_ITEM, \
    GtkTearoffMenuItemClass))
#define GTK_TEXT_BUFFER_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TEXT_BUFFER, \
    GtkTextBufferClass))
#define GTK_TEXT_CHILD_ANCHOR_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), \
    GTK_TYPE_TEXT_CHILD_ANCHOR, \
    GtkTextChildAnchorClass))
#define GTK_TEXT_MARK_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TEXT_MARK, \
    GtkTextMarkClass))
#define GTK_TEXT_TAG_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TEXT_TAG, \
    GtkTextTagClass))
#define GTK_TEXT_TAG_TABLE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TEXT_TAG_TABLE, \
    \
    GtkTextTagTableClass))
#define GTK_TEXT_VIEW_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TEXT_VIEW, \
    GtkTextViewClass))
#define GTK_TOGGLE_BUTTON_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TOGGLE_BUTTON, \
    \
    GtkToggleButtonClass))
#define GTK_TOOLBAR_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TOOLBAR, \
    GtkToolbarClass))
#define GTK_TOOLTIPS_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TOOLTIPS, \
    GtkTooltipsClass))
#define GTK_TREE_MODEL_FILTER_GET_CLASS(inst) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), \
    GTK_TYPE_TREE_MODEL_FILTER, \
    GtkTreeModelFilterClass))
#define GTK_TREE_MODEL_SORT_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TREE_MODEL_SORT, \
    \
    GtkTreeModelSortClass))
#define GTK_TREE_SELECTION_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TREE_SELECTION, \
    \
    GtkTreeSelectionClass))
#define GTK_TREE_STORE_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TREE_STORE, \
    GtkTreeStoreClass))
#define GTK_TREE_VIEW_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TREE_VIEW, \
    GtkTreeViewClass))
#define GTK_TREE_VIEW_COLUMN_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), \
    GTK_TYPE_TREE_VIEW_COLUMN, \
    GtkTreeViewColumnClass))
#define GTK_VBOX_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VBOX, \
    GtkVBoxClass))
#define GTK_VBUTTON_BOX_GET_CLASS(obj) \
    (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VBUTTON_BOX, \
    GtkVButtonBoxClass))
#define GTK_VIEWPORT_GET_CLASS(obj) \

```

```

        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VIEWPORT, \
        GtkViewportClass))
#define GTK_VPANED_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VPANED, \
        GtkVPanedClass))
#define GTK_VRULER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VRULER, \
        GtkVRulerClass))
#define GTK_VSCALE_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VSCALE, \
        GtkVScaleClass))
#define GTK_VSCROLLBAR_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VSCROLLBAR, \
        GtkVScrollbarClass))
#define GTK_VSEPARATOR_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_VSEPARATOR, \
        GtkVSeparatorClass))
#define GTK_WIDGET_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_WIDGET, \
        GtkWidgetClass))
#define GTK_WINDOW_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_WINDOW, \
        GtkWindowClass))
#define GTK_WINDOW_GROUP_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_WINDOW_GROUP, \
        GtkWindowGroupClass))
#define GTK_FILE_CHOOSER_BUTTON_GET_CLASS(object) \
        (G_TYPE_INSTANCE_GET_CLASS ((object), \
        GTK_TYPE_FILE_CHOOSER_BUTTON, \
        GtkFileChooserButtonClass))
#define GTK_TOOL_ITEM_GET_CLASS(o) \
        (G_TYPE_INSTANCE_GET_CLASS ((o), GTK_TYPE_TOOL_ITEM, \
        GtkToolItemClass))
#define GTK_ACTION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_ACTION, \
        GtkActionClass))
#define GTK_RADIO_ACTION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RADIO_ACTION, \
        GtkRadioActionClass))
#define GTK_RADIO_TOOL_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_RADIO_TOOL_BUTTON, \
        GtkRadioToolButtonClass))
#define GTK_SEPARATOR_TOOL_ITEM_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_SEPARATOR_TOOL_ITEM, \
        GtkSeparatorToolItemClass))
#define GTK_TOGGLE_ACTION_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TOGGLE_ACTION, \
        GtkToggleActionClass))
#define GTK_TOGGLE_TOOL_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), \
        GTK_TYPE_TOGGLE_TOOL_BUTTON, \
        GtkToggleToolButtonClass))
#define GTK_TOOL_BUTTON_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_TOOL_BUTTON, \
        GtkToolButtonClass))
#define GTK_UI_MANAGER_GET_CLASS(obj) \
        (G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_UI_MANAGER, \
        GtkUIManagerClass))
#define GTK_EDITABLE_GET_CLASS(inst) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((inst), GTK_TYPE_EDITABLE, \
        \
        GtkEditableClass))
#define GTK_CELL_EDITABLE_GET_IFACE(obj) \

```

```

        (G_TYPE_INSTANCE_GET_INTERFACE ((obj),
GTK_TYPE_CELL_EDITABLE, \
        GtkCellEditableIface))
#define GTK_CELL_LAYOUT_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), GTK_TYPE_CELL_LAYOUT,
\
        GtkCellLayoutIface))
#define GTK_TREE_DRAG_DEST_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj),
GTK_TYPE_TREE_DRAG_DEST, \
        GtkTreeDragDestIface))
#define GTK_TREE_DRAG_SOURCE_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj),
GTK_TYPE_TREE_DRAG_SOURCE, \
        GtkTreeDragSourceIface))
#define GTK_TREE_MODEL_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj), GTK_TYPE_TREE_MODEL,
\
        GtkTreeModelIface))
#define GTK_TREE_SORTABLE_GET_IFACE(obj) \
        (G_TYPE_INSTANCE_GET_INTERFACE ((obj),
GTK_TYPE_TREE_SORTABLE, \
        GtkTreeSortableIface))
#define GTK_OBJECT_TYPE_NAME(object) \
        (g_type_name (GTK_OBJECT_TYPE (object)))
#define
GTK_CONTAINER_WARN_INVALID_CHILD_PROPERTY_ID(object,property_id,p
spec) \
        G_OBJECT_WARN_INVALID_PSPEC ((object), "child property id",
\
        (property_id), (pspec))
#define GTK_OBJECT_UNSET_FLAGS(obj,flag) \
        G_STMT_START{ (GTK_OBJECT_FLAGS (obj)      &=
~(flag)); }G_STMT_END
#define GTK_OBJECT_SET_FLAGS(obj,flag) \
        G_STMT_START{ (GTK_OBJECT_FLAGS (obj)      |=
(flag)); }G_STMT_END
#define GTK_WIDGET_UNSET_FLAGS(wid,flag) \
        G_STMT_START{ (GTK_WIDGET_FLAGS (wid)      &=
~(flag)); }G_STMT_END
#define GTK_WIDGET_SET_FLAGS(wid,flag) \
        G_STMT_START{ (GTK_WIDGET_FLAGS (wid)      |=
(flag)); }G_STMT_END
#define GTK_MAJOR_VERSION      (2)
#define GTK_TEXT_VIEW_PRIORITY_VALIDATE (GDK_PRIORITY_REDRAW + 5)
#define GTK_TYPE_ABOUT_DIALOG  (gtk_about_dialog_get_type ())
#define GTK_TYPE_ACCEL_FLAGS    (gtk_accel_flags_get_type ())
#define GTK_TYPE_ACCEL_GROUP    (gtk_accel_group_get_type ())
#define GTK_TYPE_ACCEL_LABEL    (gtk_accel_label_get_type ())
#define GTK_TYPE_ACCEL_MAP      (gtk_accel_map_get_type ())
#define GTK_TYPE_ACCESSIBLE     (gtk_accessible_get_type ())
#define GTK_TYPE_ACTION         (gtk_action_get_type ())
#define GTK_TYPE_ACTION_GROUP   (gtk_action_group_get_type ())
#define GTK_TYPE_ADJUSTMENT     (gtk_adjustment_get_type ())
#define GTK_TYPE_ALIGNMENT      (gtk_alignment_get_type ())
#define GTK_TYPE_ANCHOR_TYPE    (gtk_anchor_type_get_type ())
#define GTK_TYPE_ARG_FLAGS      (gtk_arg_flags_get_type ())
#define GTK_TYPE_ARROW          (gtk_arrow_get_type ())
#define GTK_TYPE_ARROW_TYPE     (gtk_arrow_type_get_type ())
#define GTK_TYPE_ASPECT_FRAME   (gtk_aspect_frame_get_type ())
#define GTK_TYPE_ASSISTANT      (gtk_assistant_get_type ())
#define GTK_TYPE_ATTACH_OPTIONS (gtk_attach_options_get_type ())
#define GTK_TYPE_BIN            (gtk_bin_get_type ())
#define GTK_TYPE_BORDER         (gtk_border_get_type ())
#define GTK_TYPE_BOX            (gtk_box_get_type ())
#define GTK_TYPE_BUTTONS_TYPE   (gtk_buttons_type_get_type ())

```

```

#define GTK_TYPE_BUTTON_BOX      (gtk_button_box_get_type ())
#define                           GTK_TYPE_BUTTON_BOX_STYLE
(gtk_button_box_style_get_type())
#define GTK_TYPE_BUTTON          (gtk_button_get_type ())
#define GTK_TYPE_CALENDAR        (gtk_calendar_get_type ())
#define GTK_TYPE_CELL_EDITABLE   (gtk_cell_editable_get_type ())
#define GTK_TYPE_CELL_LAYOUT     (gtk_cell_layout_get_type ())
#define                           GTK_TYPE_CELL_RENDERER_ACCEL
(gtk_cell_renderer_accel_get_type ())
#define                           GTK_TYPE_CELL_RENDERER_ACCEL_MODE
(gtk_cell_renderer_accel_mode_get_type ())
#define                           GTK_TYPE_CELL_RENDERER_COMBO
(gtk_cell_renderer_combo_get_type ())
#define GTK_TYPE_CELL_RENDERER   (gtk_cell_renderer_get_type ())
#define                           GTK_TYPE_CELL_RENDERER_MODE
(gtk_cell_renderer_mode_get_type())
#define                           GTK_TYPE_CELL_RENDERER_SPIN
(gtk_cell_renderer_spin_get_type ())
#define                           GTK_TYPE_CELL_RENDERER_STATE
(gtk_cell_renderer_state_get_type())
#define                           GTK_TYPE_CELL_RENDERER_TEXT
(gtk_cell_renderer_text_get_type ())
#define GTK_TYPE_CELL_VIEW        (gtk_cell_view_get_type ())
#define GTK_TYPE_CHECK_BUTTON     (gtk_check_button_get_type ())
#define GTK_LIST(obj)             (GTK_CHECK_CAST ((obj), GTK_TYPE_LIST,
GtkList))
#define                           GTK_TYPE_CHECK_MENU_ITEM
(gtk_check_menu_item_get_type ())
#define   GTK_IS_ICON_VIEW(obj)   (GTK_CHECK_TYPE ((obj),
GTK_TYPE_ICON_VIEW))
#define   GTK_IS_LIST(obj)        (GTK_CHECK_TYPE ((obj),
GTK_TYPE_LIST))
#define   GTK_IS_OBJECT(object)   (GTK_CHECK_TYPE ((object),
GTK_TYPE_OBJECT))
#define GTK_TYPE_CLIPBOARD        (gtk_clipboard_get_type ())
#define GTK_TYPE_COLOR_BUTTON     (gtk_color_button_get_type ())
#define                           GTK_TYPE_COLOR_SELECTION
(gtk_color_selection_get_type ())
#define                           GTK_TYPE_COMBO_BOX_ENTRY
(gtk_combo_box_entry_get_type ())
#define GTK_TYPE_COMBO_BOX       (gtk_combo_box_get_type ())
#define GTK_TYPE_CONTAINER        (gtk_container_get_type ())
#define GTK_TYPE_CORNER_TYPE      (gtk_corner_type_get_type())
#define GTK_TYPE_CURVE            (gtk_curve_get_type ())
#define GTK_TYPE_CURVE_TYPE       (gtk_curve_type_get_type())
#define GTK_TYPE_DEBUG_FLAG       (gtk_debug_flag_get_type())
#define GTK_TYPE_DELETE_TYPE      (gtk_delete_type_get_type())
#define GTK_TYPE_DEST_DEFAULTS    (gtk_dest_defaults_get_type())
#define GTK_TYPE_DIALOG_FLAGS     (gtk_dialog_flags_get_type())
#define GTK_TYPE_DIALOG           (gtk_dialog_get_type ())
#define GTK_TYPE_DIRECTION_TYPE   (gtk_direction_type_get_type())
#define GTK_TYPE_DRAWING_AREA     (gtk_drawing_area_get_type ())
#define GTK_TYPE_EDITABLE         (gtk_editable_get_type ())
#define                           GTK_TYPE_ENTRY_COMPLETION
(gtk_entry_completion_get_type ())
#define GTK_TYPE_ENTRY            (gtk_entry_get_type ())
#define GTK_TYPE_EVENT_BOX        (gtk_event_box_get_type ())
#define GTK_TYPE_EXPANDER         (gtk_expander_get_type ())
#define GTK_TYPE_EXPANDER_STYLE   (gtk_expander_style_get_type())
#define                           GTK_TYPE_FILE_CHOOSER_ACTION
(gtk_file_chooser_action_get_type())
#define                           GTK_TYPE_FILE_CHOOSER_BUTTON
(gtk_file_chooser_button_get_type ())
#define                           GTK_TYPE_FILE_CHOOSER_DIALOG
(gtk_file_chooser_dialog_get_type ())

```

```

#define GTK_TYPE_FILE_CHOOSER_ERROR
(gtk_file_chooser_error_get_type())
#define GTK_FILE_CHOOSER_ERROR (gtk_file_chooser_error_quark ())
#define GTK_TYPE_FILE_CHOOSER (gtk_file_chooser_get_type ())
#define GTK_TYPE_FILE_CHOOSER_WIDGET
(gtk_file_chooser_widget_get_type ())
#define GTK_TYPE_FILE_FILTER_FLAGS
(gtk_file_filter_flags_get_type())
#define GTK_TYPE_FILE_FILTER (gtk_file_filter_get_type ())
#define GTK_TYPE_FILE_SELECTION (gtk_file_selection_get_type ())
#define GTK_TYPE_FIXED (gtk_fixed_get_type ())
#define GTK_TYPE_FONT_BUTTON (gtk_font_button_get_type ())
#define GTK_TYPE_FONT_SELECTION (gtk_font_selection_get_type ())
#define GTK_TYPE_FRAME (gtk_frame_get_type ())
#define GTK_TYPE_GAMMA_CURVE (gtk_gamma_curve_get_type ())
#define GTK_TYPE_HANDLE_BOX (gtk_handle_box_get_type ())
#define GTK_TYPE_HBOX (gtk_hbox_get_type ())
#define GTK_TYPE_HBUTTON_BOX (gtk_hbutton_box_get_type ())
#define GTK_TYPE_HPANED (gtk_hpaned_get_type ())
#define GTK_TYPE_HRULER (gtk_hruler_get_type ())
#define GTK_TYPE_HSCALE (gtk_hscale_get_type ())
#define GTK_TYPE_HSCROLLBAR (gtk_hscrollbar_get_type ())
#define GTK_TYPE_HSEPARATOR (gtk_hseparator_get_type ())
#define GTK_TYPE_ICON_FACTORY (gtk_icon_factory_get_type ())
#define GTK_TYPE_ICON_INFO (gtk_icon_info_get_type ())
#define GTK_TYPE_ICON_LOOKUP_FLAGS
(gtk_icon_lookup_flags_get_type())
#define GTK_TYPE_ICON_SET (gtk_icon_set_get_type ())
#define GTK_TYPE_ICON_SIZE (gtk_icon_size_get_type())
#define GTK_TYPE_ICON_SOURCE (gtk_icon_source_get_type ())
#define GTK_TYPE_ICON_THEME_ERROR
(gtk_icon_theme_error_get_type())
#define GTK_TYPE_ICON_THEME (gtk_icon_theme_get_type ())
#define GTK_TYPE_ICON_VIEW (gtk_icon_view_get_type ())
#define GTK_TYPE_IDENTIFIER (gtk_identifier_get_type ())
#define GTK_TYPE_IMAGE (gtk_image_get_type ())
#define GTK_TYPE_IMAGE_MENU_ITEM
(gtk_image_menu_item_get_type ())
#define GTK_TYPE_IMAGE_TYPE (gtk_image_type_get_type())
#define GTK_TYPE_IM_CONTEXT (gtk_im_context_get_type ())
#define GTK_TYPE_IM_CONTEXT_SIMPLE
(gtk_im_context_simple_get_type ())
#define GTK_TYPE_IM_MULTICONTEXT
(gtk_im_multicontext_get_type ())
#define GTK_TYPE_IM_PREEDIT_STYLE
(gtk_im_preedit_style_get_type())
#define GTK_TYPE_IM_STATUS_STYLE
(gtk_im_status_style_get_type())
#define GTK_TYPE_INPUT_DIALOG (gtk_input_dialog_get_type ())
#define GTK_TYPE_INVISIBLE (gtk_invisible_get_type ())
#define GTK_TYPE_ITEM (gtk_item_get_type ())
#define GTK_TYPE_JUSTIFICATION (gtk_justification_get_type())
#define GTK_TYPE_LABEL (gtk_label_get_type ())
#define GTK_TYPE_LAYOUT (gtk_layout_get_type ())
#define GTK_TYPE_LINK_BUTTON (gtk_link_button_get_type ())
#define GTK_LINK_BUTTON_H (gtk_link_button_get_type ())
#define GTK_TYPE_LIST (gtk_list_get_type ())
#define GTK_TYPE_LIST_STORE (gtk_list_store_get_type ())
#define GTK_TYPE_MATCH_TYPE (gtk_match_type_get_type())
#define GTK_TYPE_MENU_BAR (gtk_menu_bar_get_type ())
#define GTK_TYPE_MENU_DIRECTION_TYPE
(gtk_menu_direction_type_get_type())
#define GTK_TYPE_MENU (gtk_menu_get_type ())
#define GTK_TYPE_MENU_ITEM (gtk_menu_item_get_type ())
#define GTK_TYPE_MENU_SHELL (gtk_menu_shell_get_type ())

```



```

#define                                     GTK_TYPE_MENU_TOOL_BUTTON
(gtk_menu_tool_button_get_type ())
#define GTK_TYPE_MESSAGE_DIALOG (gtk_message_dialog_get_type ())
#define GTK_TYPE_MESSAGE_TYPE (gtk_message_type_get_type ())
#define GTK_TYPE_METRIC_TYPE (gtk_metric_type_get_type ())
#define GTK_TYPE_MISC (gtk_misc_get_type ())
#define GTK_TYPE_MOVEMENT_STEP (gtk_movement_step_get_type ())
#define GTK_TYPE_NOTEBOOK (gtk_notebook_get_type ())
#define GTK_TYPE_NOTEBOOK_TAB (gtk_notebook_tab_get_type ())
#define GTK_OBJECT_FLAGS(obj) (GTK_OBJECT (obj)->flags)
#define GTK_WIDGET_FLAGS(wid) (GTK_OBJECT_FLAGS (wid))
#define GTK_TYPE_OBJECT_FLAGS (gtk_object_flags_get_type ())
#define GTK_TYPE_OBJECT (gtk_object_get_type ())
#define GTK_WIDGET_TYPE(wid) (GTK_OBJECT_TYPE (wid))
#define GTK_TYPE_ORIENTATION (gtk_orientation_get_type ())
#define GTK_TYPE_PACK_TYPE (gtk_pack_type_get_type ())
#define                                     GTK_TYPE_PAGE_ORIENTATION
(gtk_page_orientation_get_type ())
#define GTK_TYPE_PAGE_SETUP (gtk_page_setup_get_type ())
#define GTK_TYPE_PAGE_SET (gtk_page_set_get_type ())
#define GTK_TYPE_PANED (gtk_paned_get_type ())
#define GTK_TYPE_PAPER_SIZE (gtk_paper_size_get_type ())
#define                                     GTK_TYPE_PATH_PRIORITY_TYPE
(gtk_path_priority_type_get_type ())
#define GTK_TYPE_PATH_TYPE (gtk_path_type_get_type ())
#define GTK_TYPE_PLUG (gtk_plug_get_type ())
#define GTK_TYPE_POLICY_TYPE (gtk_policy_type_get_type ())
#define GTK_TYPE_POSITION_TYPE (gtk_position_type_get_type ())
#define GTK_TYPE_PRINT_CONTEXT (gtk_print_context_get_type ())
#define GTK_TYPE_PRINT_DUPLEX (gtk_print_duplex_get_type ())
#define GTK_TYPE_PRINT_ERROR (gtk_print_error_get_type ())
#define                                     GTK_TYPE_PRINT_OPERATION_ACTION
(gtk_print_operation_action_get_type ())
#define                                     GTK_TYPE_PRINT_OPERATION
(gtk_print_operation_get_type ())
#define                                     GTK_TYPE_PRINT_OPERATION_RESULT
(gtk_print_operation_result_get_type ())
#define GTK_TYPE_PRINT_PAGES (gtk_print_pages_get_type ())
#define GTK_TYPE_PRINT_QUALITY (gtk_print_quality_get_type ())
#define GTK_TYPE_PRINT_SETTINGS (gtk_print_settings_get_type ())
#define GTK_TYPE_PRINT_STATUS (gtk_print_status_get_type ())
#define GTK_TYPE_PROGRESS_BAR (gtk_progress_bar_get_type ())
#define                                     GTK_TYPE_PROGRESS_BAR_STYLE
(gtk_progress_bar_style_get_type ())
#define GTK_TYPE_RADIO_ACTION (gtk_radio_action_get_type ())
#define GTK_TYPE_RADIO_BUTTON (gtk_radio_button_get_type ())
#define                                     GTK_TYPE_RADIO_MENU_ITEM
(gtk_radio_menu_item_get_type ())
#define                                     GTK_TYPE_RADIO_TOOL_BUTTON
(gtk_radio_tool_button_get_type ())
#define GTK_TYPE_RANGE (gtk_range_get_type ())
#define GTK_TYPE_RC_FLAGS (gtk_rc_flags_get_type ())
#define GTK_TYPE_RC_STYLE (gtk_rc_style_get_type ())
#define GTK_TYPE_RC_TOKEN_TYPE (gtk_rc_token_type_get_type ())
#define                                     GTK_TYPE_RECENT_CHOOSER_DIALOG
(gtk_recent_chooser_dialog_get_type ())
#define                                     GTK_TYPE_RECENT_CHOOSER_ERROR
(gtk_recent_chooser_error_get_type ())
#define                                     GTK_RECENT_CHOOSER_ERROR
(gtk_recent_chooser_error_quark ())
#define GTK_TYPE_RECENT_CHOOSER (gtk_recent_chooser_get_type ())
#define                                     GTK_TYPE_RECENT_CHOOSER_MENU
(gtk_recent_chooser_menu_get_type ())
#define                                     GTK_TYPE_RECENT_CHOOSER_WIDGET
(gtk_recent_chooser_widget_get_type ())

```

```

#define GTK_TYPE_RECENT_FILTER_FLAGS
(gtk_recent_filter_flags_get_type ())
#define GTK_TYPE_RECENT_MANAGER_ERROR
(gtk_recent_manager_error_get_type ())
#define GTK_TYPE_RECENT_SORT_TYPE
(gtk_recent_sort_type_get_type ())
#define GTK_TYPE_RELIEF_STYLE (gtk_relief_style_get_type())
#define GTK_TYPE_REQUISITION (gtk_requisition_get_type ())
#define GTK_TYPE_RESIZE_MODE (gtk_resize_mode_get_type())
#define GTK_TYPE_RESPONSE_TYPE (gtk_response_type_get_type())
#define GTK_TYPE_RULER (gtk_ruler_get_type ())
#define GTK_TYPE_SCALE (gtk_scale_get_type ())
#define GTK_TYPE_SCROLLBAR (gtk_scrollbar_get_type ())
#define GTK_TYPE_SCROLLED_WINDOW
(gtk_scrolled_window_get_type ())
#define GTK_TYPE_SCROLL_STEP (gtk_scroll_step_get_type())
#define GTK_TYPE_SCROLL_TYPE (gtk_scroll_type_get_type())
#define GTK_TYPE_SELECTION_DATA (gtk_selection_data_get_type())
#define GTK_TYPE_SELECTION_MODE (gtk_selection_mode_get_type())
#define GTK_TYPE_SENSITIVITY_TYPE
(gtk_sensitivity_type_get_type ())
#define GTK_TYPE_SEPARATOR (gtk_separator_get_type ())
#define GTK_TYPE_SEPARATOR_MENU_ITEM
(gtk_separator_menu_item_get_type ())
#define GTK_TYPE_SEPARATOR_TOOL_ITEM
(gtk_separator_tool_item_get_type ())
#define GTK_TYPE_SETTINGS (gtk_settings_get_type ())
#define GTK_TYPE_SHADOW_TYPE (gtk_shadow_type_get_type())
#define GTK_TYPE_SIDE_TYPE (gtk_side_type_get_type())
#define GTK_TYPE_SIGNAL_RUN_TYPE
(gtk_signal_run_type_get_type())
#define GTK_TYPE_SIZE_GROUP (gtk_size_group_get_type ())
#define GTK_TYPE_SIZE_GROUP_MODE
(gtk_size_group_mode_get_type())
#define GTK_TYPE_SOCKET (gtk_socket_get_type ())
#define GTK_TYPE_SORT_TYPE (gtk_sort_type_get_type())
#define GTK_TYPE_SPIN_BUTTON (gtk_spin_button_get_type ())
#define GTK_TYPE_SPIN_TYPE (gtk_spin_type_get_type())
#define GTK_TYPE_STATE_TYPE (gtk_state_type_get_type())
#define GTK_TYPE_STATUSBAR (gtk_statusbar_get_type ())
#define GTK_STYLE_ATTACHED(style) (GTK_STYLE (style)-
>attach_count > 0)
#define GTK_TYPE_STYLE (gtk_style_get_type ())
#define GTK_TYPE_SUBMENU_DIRECTION
(gtk_submenu_direction_get_type())
#define GTK_TYPE_SUBMENU_PLACEMENT
(gtk_submenu_placement_get_type())
#define GTK_TYPE_TABLE (gtk_table_get_type ())
#define GTK_TYPE_TARGET_FLAGS (gtk_target_flags_get_type())
#define GTK_TYPE_TARGET_LIST (gtk_target_list_get_type ())
#define GTK_TYPE_TEAROFF_MENU_ITEM
(gtk_tearoff_menu_item_get_type ())
#define GTK_TYPE_TEXT_ATTRIBUTES
(gtk_text_attributes_get_type ())
#define GTK_TYPE_TEXT_BUFFER (gtk_text_buffer_get_type ())
#define GTK_TYPE_TEXT_BUFFER_TARGET_INFO
(gtk_text_buffer_target_info_get_type ())
#define GTK_TYPE_TEXT_CHILD_ANCHOR
(gtk_text_child_anchor_get_type ())
#define GTK_TYPE_TEXT_DIRECTION (gtk_text_direction_get_type())
#define GTK_TYPE_TEXT_ITER (gtk_text_iter_get_type ())
#define GTK_TYPE_TEXT_MARK (gtk_text_mark_get_type ())
#define GTK_TYPE_TEXT_SEARCH_FLAGS
(gtk_text_search_flags_get_type())
#define GTK_TYPE_TEXT_TAG (gtk_text_tag_get_type ())
#define GTK_TYPE_TEXT_TAG_TABLE (gtk_text_tag_table_get_type ())

```



```

#define GTK_TYPE_TEXT_VIEW      (gtk_text_view_get_type ())
#define                          GTK_TYPE_TEXT_WINDOW_TYPE
(gtk_text_window_type_get_type())
#define GTK_TYPE_TOGGLE_ACTION  (gtk_toggle_action_get_type ())
#define GTK_TYPE_TOGGLE_BUTTON  (gtk_toggle_button_get_type ())
#define                          GTK_TYPE_TOGGLE_TOOL_BUTTON
(gtk_toggle_tool_button_get_type ())
#define                          GTK_TYPE_TOOLBAR_CHILD_TYPE
(gtk_toolbar_child_type_get_type())
#define GTK_TYPE_TOOLBAR        (gtk_toolbar_get_type ())
#define                          GTK_TYPE_TOOLBAR_SPACE_STYLE
(gtk_toolbar_space_style_get_type())
#define GTK_TYPE_TOOLBAR_STYLE  (gtk_toolbar_style_get_type())
#define GTK_TYPE_TOOLTIPS      (gtk_tooltips_get_type ())
#define GTK_TYPE_TOOL_BUTTON    (gtk_tool_button_get_type ())
#define GTK_TYPE_TOOL_ITEM      (gtk_tool_item_get_type ())
#define GTK_TYPE_TREE_DRAG_DEST (gtk_tree_drag_dest_get_type ())
#define                          GTK_TYPE_TREE_DRAG_SOURCE
(gtk_tree_drag_source_get_type ())
#define GTK_TYPE_TREE_ITER      (gtk_tree_iter_get_type ())
#define                          GTK_TYPE_TREE_MODEL_FILTER
(gtk_tree_model_filter_get_type ())
#define                          GTK_TYPE_TREE_MODEL_FLAGS
(gtk_tree_model_flags_get_type())
#define GTK_TYPE_TREE_MODEL     (gtk_tree_model_get_type ())
#define                          GTK_TYPE_TREE_MODEL_SORT
(gtk_tree_model_sort_get_type ())
#define GTK_TYPE_TREE_PATH      (gtk_tree_path_get_type ())
#define                          GTK_TYPE_TREE_ROW_REFERENCE
(gtk_tree_row_reference_get_type ())
#define GTK_TYPE_TREE_SELECTION (gtk_tree_selection_get_type ())
#define GTK_TYPE_TREE_SORTABLE  (gtk_tree_sortable_get_type ())
#define GTK_TYPE_TREE_STORE     (gtk_tree_store_get_type ())
#define                          GTK_TYPE_TREE_VIEW_COLUMN
(gtk_tree_view_column_get_type ())
#define GTK_TYPE_TREE_VIEW      (gtk_tree_view_get_type ())
#define                          GTK_TYPE_TREE_VIEW_GRID_LINES
(gtk_tree_view_grid_lines_get_type ())
#define GTK_TYPE_TREE_VIEW_MODE (gtk_tree_view_mode_get_type())
#define GTK_TYPE_UI_MANAGER     (gtk_ui_manager_get_type ())
#define GTK_TYPE_UNIT           (gtk_unit_get_type ())
#define GTK_TYPE_UPDATE_TYPE    (gtk_update_type_get_type())
#define GTK_TYPE_VBOX           (gtk_vbox_get_type ())
#define GTK_TYPE_VBUTTON_BOX    (gtk_vbutton_box_get_type ())
#define GTK_TYPE_VIEWPORT       (gtk_viewport_get_type ())
#define GTK_TYPE_VISIBILITY     (gtk_visibility_get_type())
#define GTK_TYPE_VPANED         (gtk_vpaned_get_type ())
#define GTK_TYPE_VRULER         (gtk_vruler_get_type ())
#define GTK_TYPE_VSCALE         (gtk_vscale_get_type ())
#define GTK_TYPE_VSCROLLBAR     (gtk_vscrollbar_get_type ())
#define GTK_TYPE_VSEPARATOR     (gtk_vseparator_get_type ())
#define GTK_WIDGET_SAVED_STATE(wid)      (GTK_WIDGET (wid)-
>saved_state)
#define GTK_WIDGET_STATE(wid)      (GTK_WIDGET (wid)->state)
#define GTK_TYPE_WIDGET_FLAGS      (gtk_widget_flags_get_type())
#define GTK_TYPE_WIDGET            (gtk_widget_get_type ())
#define                          GTK_TYPE_WIDGET_HELP_TYPE
(gtk_widget_help_type_get_type())
#define GTK_TYPE_WINDOW           (gtk_window_get_type ())
#define GTK_TYPE_WINDOW_GROUP     (gtk_window_group_get_type ())
#define                          GTK_TYPE_WINDOW_POSITION
(gtk_window_position_get_type())
#define GTK_TYPE_WINDOW_TYPE      (gtk_window_type_get_type())
#define GTK_TYPE_WRAP_MODE        (gtk_wrap_mode_get_type())
#define GTK_PRIORITY_RESIZE        (G_PRIORITY_HIGH_IDLE + 10)

```

```

#define GTK_ASSISTANT_CLASS(c) (G_TYPE_CHECK_CLASS_CAST ((c),
GTK_TYPE_ASSISTANT, GtkAssistantClass))
#define GTK_CELL_RENDERER_ACCEL_CLASS(klass)
(G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CELL_RENDERER_ACCEL,
GtkCellRendererAccelClass))
#define GTK_CELL_RENDERER_SPIN_CLASS(klass)
(G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_CELL_RENDERER_SPIN,
GtkCellRendererSpinClass))
#define GTK_LINK_BUTTON_CLASS(klass) (G_TYPE_CHECK_CLASS_CAST
((klass), GTK_TYPE_LINK_BUTTON, GtkLinkButtonClass))
#define GTK_PRINT_OPERATION_CLASS(klass)
(G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_PRINT_OPERATION,
GtkPrintOperationClass))
#define GTK_RECENT_CHOOSER_DIALOG_CLASS(klass)
(G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_RECENT_CHOOSER_DIALOG,
GtkRecentChooserDialogClass))
#define GTK_RECENT_CHOOSER_MENU_CLASS(klass)
(G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_RECENT_CHOOSER_MENU,
GtkRecentChooserMenuClass))
#define GTK_RECENT_CHOOSER_WIDGET_CLASS(klass)
(G_TYPE_CHECK_CLASS_CAST ((klass), GTK_TYPE_RECENT_CHOOSER_WIDGET,
GtkRecentChooserWidgetClass))
#define GTK_IS_ASSISTANT_CLASS(c) (G_TYPE_CHECK_CLASS_TYPE
((c), GTK_TYPE_ASSISTANT))
#define GTK_IS_CELL_RENDERER_ACCEL_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CELL_RENDERER_ACCEL))
#define GTK_IS_CELL_RENDERER_SPIN_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_CELL_RENDERER_SPIN))
#define GTK_IS_LINK_BUTTON_CLASS(klass) (G_TYPE_CHECK_CLASS_TYPE
((klass), GTK_TYPE_LINK_BUTTON))
#define GTK_IS_PRINT_OPERATION_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_PRINT_OPERATION))
#define GTK_IS_RECENT_CHOOSER_DIALOG_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RECENT_CHOOSER_DIALOG))
#define GTK_IS_RECENT_CHOOSER_MENU_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RECENT_CHOOSER_MENU))
#define GTK_IS_RECENT_CHOOSER_WIDGET_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RECENT_CHOOSER_WIDGET))
#define GTK_ASSISTANT(o) (G_TYPE_CHECK_INSTANCE_CAST ((o),
GTK_TYPE_ASSISTANT, GtkAssistant))
#define GTK_CELL_RENDERER_ACCEL(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CELL_RENDERER_ACCEL,
GtkCellRendererAccel))
#define GTK_CELL_RENDERER_SPIN(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_CELL_RENDERER_SPIN,
GtkCellRendererSpin))
#define GTK_LINK_BUTTON(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_LINK_BUTTON, GtkLinkButton))
#define GTK_PAGE_SETUP(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_PAGE_SETUP, GtkPageSetup))
#define GTK_PRINT_CONTEXT(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_PRINT_CONTEXT, GtkPrintContext))
#define GTK_PRINT_OPERATION(obj) (G_TYPE_CHECK_INSTANCE_CAST
((obj), GTK_TYPE_PRINT_OPERATION, GtkPrintOperation))
#define GTK_PRINT_SETTINGS(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_PRINT_SETTINGS, GtkPrintSettings))
#define GTK_RECENT_CHOOSER(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_RECENT_CHOOSER, GtkRecentChooser))
#define GTK_RECENT_CHOOSER_DIALOG(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RECENT_CHOOSER_DIALOG,
GtkRecentChooserDialog))
#define GTK_RECENT_CHOOSER_MENU(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RECENT_CHOOSER_MENU,
GtkRecentChooserMenu))

```

```

#define                                GTK_RECENT_CHOOSER_WIDGET(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj), GTK_TYPE_RECENT_CHOOSER_WIDGET,
GtkRecentChooserWidget))
#define GTK_IS_ASSISTANT(o)           (G_TYPE_CHECK_INSTANCE_TYPE ((o),
GTK_TYPE_ASSISTANT))
#define GTK_IS_BIN(obj)              (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_BIN))
#define GTK_IS_BOX(obj)              (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_BOX))
#define                                GTK_IS_CELL_RENDERER_ACCEL(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CELL_RENDERER_ACCEL))
#define                                GTK_IS_CELL_RENDERER_SPIN(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_CELL_RENDERER_SPIN))
#define GTK_IS_LINK_BUTTON(obj)      (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_LINK_BUTTON))
#define GTK_IS_PAGE_SETUP(obj)      (G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_PAGE_SETUP))
#define GTK_IS_PRINT_CONTEXT(obj)    (G_TYPE_CHECK_INSTANCE_TYPE
((obj), GTK_TYPE_PRINT_CONTEXT))
#define                                GTK_IS_PRINT_OPERATION(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_PRINT_OPERATION))
#define GTK_IS_PRINT_SETTINGS(obj)   (G_TYPE_CHECK_INSTANCE_TYPE
((obj), GTK_TYPE_PRINT_SETTINGS))
#define GTK_IS_RECENT_CHOOSER(obj)   (G_TYPE_CHECK_INSTANCE_TYPE
((obj), GTK_TYPE_RECENT_CHOOSER))
#define                                GTK_IS_RECENT_CHOOSER_DIALOG(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_RECENT_CHOOSER_DIALOG))
#define                                GTK_IS_RECENT_CHOOSER_MENU(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj), GTK_TYPE_RECENT_CHOOSER_MENU))
#define                                GTK_IS_RECENT_CHOOSER_WIDGET(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_RECENT_CHOOSER_WIDGET))
#define GTK_OBJECT_TYPE(object)      (G_TYPE_FROM_INSTANCE (object))
#define GTK_ASSISTANT_GET_CLASS(o)   (G_TYPE_INSTANCE_GET_CLASS
((o), GTK_TYPE_ASSISTANT, GtkAssistantClass))
#define                                GTK_CELL_RENDERER_ACCEL_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CELL_RENDERER_ACCEL,
GtkCellRendererAccelClass))
#define                                GTK_CELL_RENDERER_SPIN_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_CELL_RENDERER_SPIN,
GtkCellRendererSpinClass))
#define GTK_LINK_BUTTON_GET_CLASS(obj) (G_TYPE_INSTANCE_GET_CLASS
((obj), GTK_TYPE_LINK_BUTTON, GtkLinkButtonClass))
#define                                GTK_PRINT_OPERATION_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_PRINT_OPERATION,
GtkPrintOperationClass))
#define                                GTK_RECENT_CHOOSER_DIALOG_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RECENT_CHOOSER_DIALOG,
GtkRecentChooserDialogClass))
#define                                GTK_RECENT_CHOOSER_MENU_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RECENT_CHOOSER_MENU,
GtkRecentChooserMenuClass))
#define                                GTK_RECENT_CHOOSER_WIDGET_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RECENT_CHOOSER_WIDGET,
GtkRecentChooserWidgetClass))
#define                                GTK_RECENT_CHOOSER_GET_IFACE(inst)
(G_TYPE_INSTANCE_GET_INTERFACE ((inst), GTK_TYPE_RECENT_CHOOSER,
GtkRecentChooserIface))
#define GTK_BUTTONBOX_DEFAULT        -1
#define GTK_INPUT_ERROR              -1
#define GTK_PATH_PRIO_MASK            0x0f
#define GTK_MINOR_VERSION             10
#define GTK_BINARY_AGE                1004
#define GTK_INTERFACE_AGE             4
#define GTK_MICRO_VERSION              4

```

```

#define GTK_MAX_COMPOSE_LEN      7
#define GTK_PRINT_SETTINGS_COLLATE      "collate"
#define GTK_PRINT_SETTINGS_DEFAULT_SOURCE      "default-source"
#define GTK_PRINT_SETTINGS_DITHER      "dither"
#define GTK_PRINT_SETTINGS_DUPLEX      "duplex"
#define GTKMAIN_C_VAR      extern
#define GTKVAR      extern
#define GTK_PRINT_SETTINGS_FINISHINGS      "finishings"
#define GTK_STOCK_ABOUT      "gtk-about"
#define GTK_STOCK_ADD      "gtk-add"
#define GTK_STOCK_APPLY      "gtk-apply"
#define GTK_STOCK_BOLD      "gtk-bold"
#define GTK_STOCK_CANCEL      "gtk-cancel"
#define GTK_STOCK_CDROM      "gtk-cdrom"
#define GTK_STOCK_CLEAR      "gtk-clear"
#define GTK_STOCK_CLOSE      "gtk-close"
#define GTK_STOCK_COLOR_PICKER      "gtk-color-picker"
#define GTK_STOCK_CONNECT      "gtk-connect"
#define GTK_STOCK_CONVERT      "gtk-convert"
#define GTK_STOCK_COPY      "gtk-copy"
#define GTK_STOCK_CUT      "gtk-cut"
#define GTK_STOCK_DELETE      "gtk-delete"
#define GTK_STOCK_DIALOG_AUTHENTICATION      "gtk-dialog-authentication"
#define GTK_STOCK_DIALOG_ERROR      "gtk-dialog-error"
#define GTK_STOCK_DIALOG_INFO      "gtk-dialog-info"
#define GTK_STOCK_DIALOG_QUESTION      "gtk-dialog-question"
#define GTK_STOCK_DIALOG_WARNING      "gtk-dialog-warning"
#define GTK_STOCK_DIRECTORY      "gtk-directory"
#define GTK_STOCK_DISCONNECT      "gtk-disconnect"
#define GTK_STOCK_DND      "gtk-dnd"
#define GTK_STOCK_DND_MULTIPLE      "gtk-dnd-multiple"
#define GTK_STOCK_EDIT      "gtk-edit"
#define GTK_STOCK_EXECUTE      "gtk-execute"
#define GTK_STOCK_FILE      "gtk-file"
#define GTK_STOCK_FIND      "gtk-find"
#define GTK_STOCK_FIND_AND_REPLACE      "gtk-find-and-replace"
#define GTK_STOCK_FLOPPY      "gtk-floppy"
#define GTK_STOCK_FULLSCREEN      "gtk-fullscreen"
#define GTK_STOCK_GO_BACK      "gtk-go-back"
#define GTK_STOCK_GO_DOWN      "gtk-go-down"
#define GTK_STOCK_GO_FORWARD      "gtk-go-forward"
#define GTK_STOCK_GO_UP      "gtk-go-up"
#define GTK_STOCK_GOTO_BOTTOM      "gtk-goto-bottom"
#define GTK_STOCK_GOTO_FIRST      "gtk-goto-first"
#define GTK_STOCK_GOTO_LAST      "gtk-goto-last"
#define GTK_STOCK_GOTO_TOP      "gtk-goto-top"
#define GTK_STOCK_HARDDISK      "gtk-harddisk"
#define GTK_STOCK_HELP      "gtk-help"
#define GTK_STOCK_HOME      "gtk-home"
#define GTK_STOCK_INDENT      "gtk-indent"
#define GTK_STOCK_INDEX      "gtk-index"
#define GTK_STOCK_INFO      "gtk-info"
#define GTK_STOCK_ITALIC      "gtk-italic"
#define GTK_STOCK_JUMP_TO      "gtk-jump-to"
#define GTK_STOCK_JUSTIFY_CENTER      "gtk-justify-center"
#define GTK_STOCK_JUSTIFY_FILL      "gtk-justify-fill"
#define GTK_STOCK_JUSTIFY_LEFT      "gtk-justify-left"
#define GTK_STOCK_JUSTIFY_RIGHT      "gtk-justify-right"
#define GTK_STOCK_LEAVE_FULLSCREEN      "gtk-leave-fullscreen"
#define GTK_STOCK_MEDIA_FORWARD      "gtk-media-forward"
#define GTK_STOCK_MEDIA_NEXT      "gtk-media-next"
#define GTK_STOCK_MEDIA_PAUSE      "gtk-media-pause"
#define GTK_STOCK_MEDIA_PLAY      "gtk-media-play"
#define GTK_STOCK_MEDIA_PREVIOUS      "gtk-media-previous"
#define GTK_STOCK_MEDIA_RECORD      "gtk-media-record"

```

```

#define GTK_STOCK_MEDIA_REWIND "gtk-media-rewind"
#define GTK_STOCK_MEDIA_STOP "gtk-media-stop"
#define GTK_STOCK_MISSING_IMAGE "gtk-missing-image"
#define GTK_STOCK_NETWORK "gtk-network"
#define GTK_STOCK_NEW "gtk-new"
#define GTK_STOCK_NO "gtk-no"
#define GTK_STOCK_OK "gtk-ok"
#define GTK_STOCK_OPEN "gtk-open"
#define GTK_STOCK_PASTE "gtk-paste"
#define GTK_STOCK_PREFERENCES "gtk-preferences"
#define GTK_STOCK_PRINT "gtk-print"
#define GTK_STOCK_PRINT_PREVIEW "gtk-print-preview"
#define GTK_STOCK_PROPERTIES "gtk-properties"
#define GTK_STOCK_QUIT "gtk-quit"
#define GTK_STOCK_REDO "gtk-redo"
#define GTK_STOCK_REFRESH "gtk-refresh"
#define GTK_STOCK_REMOVE "gtk-remove"
#define GTK_STOCK_REVERT_TO_SAVED "gtk-revert-to-saved"
#define GTK_STOCK_SAVE "gtk-save"
#define GTK_STOCK_SAVE_AS "gtk-save-as"
#define GTK_STOCK_SELECT_ALL "gtk-select-all"
#define GTK_STOCK_SELECT_COLOR "gtk-select-color"
#define GTK_STOCK_SELECT_FONT "gtk-select-font"
#define GTK_STOCK_SORT_ASCENDING "gtk-sort-ascending"
#define GTK_STOCK_SORT_DESCENDING "gtk-sort-descending"
#define GTK_STOCK_SPELL_CHECK "gtk-spell-check"
#define GTK_STOCK_STOP "gtk-stop"
#define GTK_STOCK_STRIKETHROUGH "gtk-strikethrough"
#define GTK_STOCK_UNDELETE "gtk-undelete"
#define GTK_STOCK_UNDERLINE "gtk-underline"
#define GTK_STOCK_UNDO "gtk-undo"
#define GTK_STOCK_UNINDENT "gtk-unindent"
#define GTK_STOCK_YES "gtk-yes"
#define GTK_STOCK_ZOOM_100 "gtk-zoom-100"
#define GTK_STOCK_ZOOM_FIT "gtk-zoom-fit"
#define GTK_STOCK_ZOOM_IN "gtk-zoom-in"
#define GTK_STOCK_ZOOM_OUT "gtk-zoom-out"
#define gtk_accel_label_get_accelerator_width
gtk_accel_label_get_accel_width
#define gtk_binding_entry_add gtk_binding_entry_clear
#define GTK_ICON_THEME_ERROR gtk_icon_theme_error_quark ()
#define GTK_PRINT_ERROR gtk_print_error_quark ()
#define GTK_SIGNAL_FUNC(f) G_CALLBACK(f)
#define GTK_CHECK_CLASS_CAST G_TYPE_CHECK_CLASS_CAST
#define GTK_CHECK_CLASS_TYPE G_TYPE_CHECK_CLASS_TYPE
#define GTK_CHECK_CAST G_TYPE_CHECK_INSTANCE_CAST
#define GTK_CHECK_TYPE G_TYPE_CHECK_INSTANCE_TYPE
#define GTK_CHECK_GET_CLASS G_TYPE_INSTANCE_GET_CLASS
#define GTK_PAPER_NAME_A3 "iso_a3"
#define GTK_PAPER_NAME_A4 "iso_a4"
#define GTK_PAPER_NAME_A5 "iso_a5"
#define GTK_PAPER_NAME_B5 "iso_b5"
#define GTK_PRINT_SETTINGS_MEDIA_TYPE "media-type"
#define GTK_PRINT_SETTINGS_N_COPIES "n-copies"
#define GTK_PAPER_NAME_EXECUTIVE "na_executive"
#define GTK_PAPER_NAME_LEGAL "na_legal"
#define GTK_PAPER_NAME_LETTER "na_letter"
#define GTK_PRINT_SETTINGS_NUMBER_UP "number-up"
#define GTK_PRINT_SETTINGS_NUMBER_UP_LAYOUT "number-up-layout"
#define GTK_PRINT_SETTINGS_ORIENTATION "orientation"
#define GTK_PRINT_SETTINGS_OUTPUT_BIN "output-bin"
#define GTK_PRINT_SETTINGS_OUTPUT_FILE_FORMAT "output-file-format"
#define GTK_PRINT_SETTINGS_OUTPUT_URI "output-uri"
#define GTK_PRINT_SETTINGS_PAGE_RANGES "page-ranges"
#define GTK_PRINT_SETTINGS_PAGE_SET "page-set"

```

```

#define GTK_PRINT_SETTINGS_PAPER_FORMAT "paper-format"
#define GTK_PRINT_SETTINGS_PAPER_HEIGHT "paper-height"
#define GTK_PRINT_SETTINGS_PAPER_WIDTH "paper-width"
#define GTK_PRINT_SETTINGS_PRINT_PAGES "print-pages"
#define GTK_PRINT_SETTINGS_PRINTER "printer"
#define GTK_PRINT_SETTINGS_PRINTER_LPI "printer-lpi"
#define GTK_PRINT_SETTINGS_QUALITY "quality"
#define GTK_PRINT_SETTINGS_RESOLUTION "resolution"
#define GTK_PRINT_SETTINGS_RESOLUTION_X "resolution-x"
#define GTK_PRINT_SETTINGS_RESOLUTION_Y "resolution-y"
#define GTK_PRINT_SETTINGS_REVERSE "reverse"
#define GTK_PRINT_SETTINGS_SCALE "scale"
#define GTK_PRINT_SETTINGS_USE_COLOR "use-color"
#define GTK_PRINT_SETTINGS_WIN32_DRIVER_EXTRA "win32-driver-extra"
#define GTK_PRINT_SETTINGS_WIN32_DRIVER_VERSION "win32-driver-version"

typedef struct _GtkIconSet GtkIconSet;
typedef struct _GtkObject {
    GObject parent_instance;
    guint32 flags;
} GtkObject;
typedef enum {
    GTK_RC_FG = 1,
    GTK_RC_BG = 2,
    GTK_RC_TEXT = 4,
    GTK_RC_BASE = 8
} GtkRcFlags;
typedef struct _GtkRcStyle {
    GObject parent_instance;
    gchar *name;
    gchar *bg_pixmap_name[5];
    PangoFontDescription *font_desc;
    GtkRcFlags color_flags[5];
    GdkColor fg[5];
    GdkColor bg[5];
    GdkColor text[5];
    GdkColor base[5];
    gint xthickness;
    gint ythickness;
    GArray *rc_properties;
    GSList *rc_style_lists;
    GSList *icon_factories;
    guint engine_specified:1;
} GtkRcStyle;
typedef struct _GtkStyle {
    GObject parent_instance;
    GdkColor fg[5];
    GdkColor bg[5];
    GdkColor light[5];
    GdkColor dark[5];
    GdkColor mid[5];
    GdkColor text[5];
    GdkColor base[5];
    GdkColor text_aa[5];
    GdkColor black;
    GdkColor white;
    PangoFontDescription *font_desc;
    gint xthickness;
    gint ythickness;
    GdkGC *fg_gc[5];
    GdkGC *bg_gc[5];
    GdkGC *light_gc[5];
    GdkGC *dark_gc[5];
    GdkGC *mid_gc[5];

```



```

    GdkGC *text_gc[5];
    GdkGC *base_gc[5];
    GdkGC *text_aa_gc[5];
    GdkGC *black_gc;
    GdkGC *white_gc;
    GdkPixmap *bg_pixmap[5];
    gint attach_count;
    gint depth;
    GdkColormap *colormap;
    GdkFont *private_font;
    PangoFontDescription *private_font_desc;
    GtkRcStyle *rc_style;
    GSList *styles;
    GArray *property_cache;
    GSList *icon_factories;
} GtkStyle;
typedef struct _GtkRequisition {
    gint width;
    gint height;
} GtkRequisition;
typedef GdkRectangle GtkAllocation;
typedef struct _GtkWidget {
    GObject object;
    guint16 private_flags;
    guint8 state;
    guint8 saved_state;
    gchar *name;
    GtkStyle *style;
    GtkRequisition requisition;
    GtkAllocation allocation;
    GdkWindow *window;
    GtkWidget *parent;
} GtkWidget;
typedef struct _GtkAdjustment {
    GObject parent_instance;
    gdouble lower;
    gdouble upper;
    gdouble value;
    gdouble step_increment;
    gdouble page_increment;
    gdouble page_size;
} GtkAdjustment;
typedef struct _GtkProgress {
    GtkWidget widget;
    GtkAdjustment *adjustment;
    GdkPixmap *offscreen_pixmap;
    gchar *format;
    gfloat x_align;
    gfloat y_align;
    guint show_text:1;
    guint activity_mode:1;
    guint use_text_format:1;
} GtkProgress;
typedef enum {
    GTK_PROGRESS_CONTINUOUS = 0,
    GTK_PROGRESS_DISCRETE = 1
} GtkProgressBarStyle;
typedef enum {
    GTK_PROGRESS_LEFT_TO_RIGHT = 0,
    GTK_PROGRESS_RIGHT_TO_LEFT = 1,
    GTK_PROGRESS_BOTTOM_TO_TOP = 2,
    GTK_PROGRESS_TOP_TO_BOTTOM = 3
} GtkProgressBarOrientation;
typedef struct _GtkProgressBar {
    GtkProgress progress;
    GtkProgressBarStyle bar_style;

```

```

    GtkProgressBarOrientation orientation;
    guint blocks;
    gint in_block;
    gint activity_pos;
    guint activity_step;
    guint activity_blocks;
    gdouble pulse_fraction;
    guint activity_dir:1;
    guint ellipsize:3;
} GtkProgressBar;
typedef struct _GtkTextTagTable {
    GObject parent_instance;
    GHashTable *hash;
    GSList *anonymous;
    gint anon_count;
    GSList *buffers;
} GtkTextTagTable;
typedef struct _GtkTextBTree GtkTextBTree;
typedef struct _GtkTextLogAttrCache GtkTextLogAttrCache;
typedef struct _GtkTextBuffer {
    GObject parent_instance;
    GtkTextTagTable *tag_table;
    GtkTextBTree *btree;
    GSList *clipboard_contents_buffers;
    GSList *selection_clipboards;
    GtkTextLogAttrCache *log_attr_cache;
    guint user_action_count;
    guint modified:1;
} GtkTextBuffer;
typedef struct _GtkTextIter {
    gpointer dummy1;
    gpointer dummy2;
    gint dummy3;
    gint dummy4;
    gint dummy5;
    gint dummy6;
    gint dummy7;
    gint dummy8;
    gpointer dummy9;
    gpointer dummy10;
    gint dummy11;
    gint dummy12;
    gint dummy13;
    gpointer dummy14;
} GtkTextIter;
typedef GType GtkType;
typedef struct _GtkContainer {
    GtkWidget widget;
    GtkWidget *focus_child;
    guint border_width:16;
    guint need_resize:1;
    guint resize_mode:2;
    guint reallocate_redraws:1;
    guint has_focus_chain:1;
} GtkContainer;
typedef struct _GtkBin {
    GtkContainer container;
    GtkWidget *child;
} GtkBin;
typedef struct _GtkComboBoxPrivate GtkComboBoxPrivate;
typedef struct _GtkComboBox {
    GtkBin parent_instance;
    GtkComboBoxPrivate *priv;
} GtkComboBox;
typedef struct _GtkMisc {
    GtkWidget widget;

```



```

    gfloat xalign;
    gfloat yalign;
    guint16 xpad;
    guint16 ypad;
} GtkMisc;
typedef struct _GtkBox {
    GtkContainer container;
    GList *children;
    gint16 spacing;
    guint homogeneous:1;
} GtkBox;
typedef struct _GtkHBox {
    GtkBox box;
} GtkHBox;
typedef struct _GtkStatusbar {
    GtkHBox parent_widget;
    GtkWidget *frame;
    GtkWidget *label;
    GSList *messages;
    GSList *keys;
    guint seq_context_id;
    guint seq_message_id;
    GdkWindow *grip_window;
    guint has_resize_grip:1;
} GtkStatusbar;
typedef struct _GtkTargetList {
    GList *list;
    guint ref_count;
} GtkTargetList;
typedef struct _GtkWindow {
    GtkBin bin;
    gchar *title;
    gchar *wmclass_name;
    gchar *wmclass_class;
    gchar *wm_role;
    GtkWidget *focus_widget;
    GtkWidget *default_widget;
    GtkWindow *transient_parent;
    GtkWindowGeometryInfo *geometry_info;
    GdkWindow *frame;
    GtkWindowGroup *group;
    guint16 configure_request_count;
    guint allow_shrink:1;
    guint allow_grow:1;
    guint configure_notify_received:1;
    guint need_default_position:1;
    guint need_default_size:1;
    guint position:3;
    guint type:4;
    guint has_user_ref_count:1;
    guint has_focus:1;
    guint modal:1;
    guint destroy_with_parent:1;
    guint has_frame:1;
    guint iconify_initially:1;
    guint stick_initially:1;
    guint maximize_initially:1;
    guint decorated:1;
    guint type_hint:3;
    guint gravity:5;
    guint is_active:1;
    guint has_toplevel_focus:1;
    guint frame_left;
    guint frame_top;
    guint frame_right;
    guint frame_bottom;

```

```

    guint keys_changed_handler;
    GdkModifierType mnemonic_modifier;
    GdkScreen *screen;
} GtkWidget;
typedef struct _GtkWindowGeometryInfo GtkWidgetGeometryInfo;
typedef struct _GtkWindowGroup {
    GObject parent_instance;
    GSList *grabs;
} GtkWindowGroup;
typedef struct _GtkTreeRowReference GtkTreeRowReference;
typedef enum {
    GTK_TREE_MODEL_ITERS_PERSIST = 1,
    GTK_TREE_MODEL_LIST_ONLY = 2
} GtkTreeModelFlags;
typedef struct _GtkTreeModel GtkTreeModel;
typedef struct _GtkTargetEntry {
    gchar *target;
    guint flags;
    guint info;
} GtkTargetEntry;
typedef struct _GtkIMContext {
    GObject parent_instance;
} GtkIMContext;
typedef struct _GtkEntry {
    GtkWidget widget;
    gchar *text;
    guint editable:1;
    guint visible:1;
    guint overwrite_mode:1;
    guint in_drag:1;
    guint16 text_length;
    guint16 text_max_length;
    GdkWindow *text_area;
    GtkIMContext *im_context;
    GtkWidget *popup_menu;
    gint current_pos;
    gint selection_bound;
    PangoLayout *cached_layout;
    guint cache_includes_preedit:1;
    guint need_im_reset:1;
    guint has_frame:1;
    guint activates_default:1;
    guint cursor_visible:1;
    guint in_click:1;
    guint is_cell_renderer:1;
    guint editing_canceled:1;
    guint mouse_cursor_obscured:1;
    guint select_words:1;
    guint select_lines:1;
    guint resolved_dir:4;
    guint truncate_multiline:1;
    guint button;
    guint blink_timeout;
    guint recompute_idle;
    gint scroll_offset;
    gint ascent;
    gint descent;
    guint16 text_size;
    guint16 n_bytes;
    guint16 preedit_length;
    guint16 preedit_cursor;
    gint dnd_position;
    gint drag_start_x;
    gint drag_start_y;
    gunichar invisible_char;
    gint width_chars;

```

```

} GtkEntry;
typedef struct _GtkComboBoxEntryPrivate GtkComboBoxEntryPrivate;
typedef struct _GtkComboBoxEntry {
    GtkComboBox parent_instance;
    GtkComboBoxEntryPrivate *priv;
} GtkComboBoxEntry;
typedef struct _GtkLayout {
    GtkContainer container;
    GList *children;
    guint width;
    guint height;
    GtkAdjustment *hadjustment;
    GtkAdjustment *vadjustment;
    GdkWindow *bin_window;
    GdkVisibilityState visibility;
    gint scroll_x;
    gint scroll_y;
    guint freeze_count;
} GtkLayout;
typedef struct _GtkTreeIter {
    gint stamp;
    gpointer user_data;
    gpointer user_data2;
    gpointer user_data3;
} GtkTreeIter;
typedef gint (*GtkTreeIterCompareFunc) (GtkTreeModel *, GtkTreeIter
*,
                                     GtkTreeIter *, gpointer);
typedef void (*GtkDestroyNotify) (gpointer);
typedef struct _GtkTreeStore {
    GObject parent;
    gint stamp;
    gpointer root;
    gpointer last;
    gint n_columns;
    gint sort_column_id;
    GList *sort_list;
    GtkSortType order;
    GType *column_headers;
    GtkTreeIterCompareFunc default_sort_func;
    gpointer default_sort_data;
    GtkDestroyNotify default_sort_destroy;
    guint columns_dirty:1;
} GtkTreeStore;
typedef struct _GtkDialog {
    GtkWindow window;
    GtkWidget *vbox;
    GtkWidget *action_area;
    GtkWidget *separator;
} GtkDialog;
typedef struct _GtkAboutDialog {
    GtkDialog parent_instance;
    gpointer private_data;
} GtkAboutDialog;
typedef void (*GtkAboutDialogActivateLinkFunc) (GtkAboutDialog *,
                                                const gchar *, gpointer);
typedef struct _GtkTreeSortable GtkTreeSortable;
typedef struct _GtkRangeLayout GtkRangeLayout;
typedef struct _GtkRangeStepTimer GtkRangeStepTimer;
typedef struct _GtkRange {
    GtkWidget widget;
    GtkAdjustment *adjustment;
    GtkUpdateType update_policy;
    guint inverted:1;
    guint flippable:1;
    guint has_stepper_a:1;

```

```

    guint has_stepper_b:1;
    guint has_stepper_c:1;
    guint has_stepper_d:1;
    guint need_recalc:1;
    guint slider_size_fixed:1;
    gint min_slider_size;
    GtkOrientation orientation;
    GdkRectangle range_rect;
    gint slider_start;
    gint slider_end;
    gint round_digits;
    guint trough_click_forward:1;
    guint update_pending:1;
    GtkRangeLayout *layout;
    GtkRangeStepTimer *timer;
    gint slide_initial_slider_position;
    gint slide_initial_coordinate;
    guint update_timeout_id;
    GdkWindow *event_window;
} GtkRange;
typedef struct _GtkAccelKey {
    guint accel_key;
    GdkModifierType accel_mods;
    guint accel_flags:16;
} GtkAccelKey;
typedef struct _GtkAccelGroupEntry {
    GtkAccelKey key;
    GClosure *closure;
    GQuark accel_path_quark;
} GtkAccelGroupEntry;
typedef struct _GtkAccelGroup {
    GObject parent;
    guint lock_count;
    GdkModifierType modifier_mask;
    GSList *acceleratables;
    guint n_accels;
    GtkAccelGroupEntry *priv_accels;
} GtkAccelGroup;
typedef struct _GtkSocket {
    GtkContainer container;
    guint16 request_width;
    guint16 request_height;
    guint16 current_width;
    guint16 current_height;
    GdkWindow *plug_window;
    GtkWidget *plug_widget;
    gshort xembed_version;
    guint same_app:1;
    guint focus_in:1;
    guint have_size:1;
    guint need_map:1;
    guint is_mapped:1;
    guint active:1;
    GtkAccelGroup *accel_group;
    GtkWidget *toplevel;
} GtkSocket;
typedef struct _GtkCellEditable GtkCellEditable;
typedef enum {
    GTK_TREE_VIEW_COLUMN_GROW_ONLY = 0,
    GTK_TREE_VIEW_COLUMN_AUTOSIZE = 1,
    GTK_TREE_VIEW_COLUMN_FIXED = 2
} GtkTreeViewColumnSizing;
typedef struct _GtkTreeViewColumn {
    GObject parent;
    GtkWidget *tree_view;
    GtkWidget *button;

```

```

GtkWidget *child;
GtkWidget *arrow;
GtkWidget *alignment;
GdkWindow *window;
GtkCellEditable *editable_widget;
gfloat xalign;
guint property_changed_signal;
gint spacing;
GtkTreeViewColumnSizing column_type;
gint requested_width;
gint button_request;
gint resized_width;
gint width;
gint fixed_width;
gint min_width;
gint max_width;
gint drag_x;
gint drag_y;
gchar *title;
GList *cell_list;
guint sort_clicked_signal;
guint sort_column_changed_signal;
gint sort_column_id;
GtkSortType sort_order;
guint visible:1;
guint resizable:1;
guint clickable:1;
guint dirty:1;
guint show_sort_indicator:1;
guint maybe_reordered:1;
guint reorderable:1;
guint use_resized_width:1;
guint expand:1;
} GtkTreeViewColumn;
typedef struct _GtkCellRenderer {
    GObject parent;
    gfloat xalign;
    gfloat yalign;
    gint width;
    gint height;
    guint16 xpad;
    guint16 ypad;
    guint mode:2;
    guint visible:1;
    guint is_expander:1;
    guint is_expanded:1;
    guint cell_background_set:1;
    guint sensitive:1;
    guint editing:1;
} GtkCellRenderer;
typedef struct _GtkButtonBox {
    GtkBox box;
    gint child_min_width;
    gint child_min_height;
    gint child_ipad_x;
    gint child_ipad_y;
    GtkButtonBoxStyle layout_style;
} GtkButtonBox;
typedef struct _GtkActionPrivate GtkActionPrivate;
typedef struct _GtkAction {
    GObject object;
    GtkActionPrivate *private_data;
} GtkAction;
typedef struct _GtkToggleActionPrivate GtkToggleActionPrivate;
typedef struct _GtkToggleAction {
    GtkAction parent;

```

```

    GtkToggleActionPrivate *private_data;
} GtkToggleAction;
typedef struct _GtkTextAppearance {
    GdkColor bg_color;
    GdkColor fg_color;
    GdkBitmap *bg_stipple;
    GdkBitmap *fg_stipple;
    gint rise;
    gpointer padding1;
    guint underline:4;
    guint strikethrough:1;
    guint draw_bg:1;
    guint inside_selection:1;
    guint is_text:1;
    guint pad1:1;
    guint pad2:1;
    guint pad3:1;
    guint pad4:1;
} GtkTextAppearance;
typedef struct _GtkTextAttributes {
    guint refcount;
    GtkTextAppearance appearance;
    GtkJustification justification;
    GtkTextDirection direction;
    PangoFontDescription *font;
    gdouble font_scale;
    gint left_margin;
    gint indent;
    gint right_margin;
    gint pixels_above_lines;
    gint pixels_below_lines;
    gint pixels_inside_wrap;
    PangoTabArray *tabs;
    GtkWrapMode wrap_mode;
    PangoLanguage *language;
    GdkColor *pg_bg_color;
    guint invisible:1;
    guint bg_full_height:1;
    guint editable:1;
    guint realized:1;
    guint pad1:1;
    guint pad2:1;
    guint pad3:1;
    guint pad4:1;
} GtkTextAttributes;
typedef struct _GtkTextTag {
    GObject parent_instance;
    GtkTextTagTable *table;
    char *name;
    int priority;
    GtkTextAttributes *values;
    guint bg_color_set:1;
    guint bg_stipple_set:1;
    guint fg_color_set:1;
    guint scale_set:1;
    guint fg_stipple_set:1;
    guint justification_set:1;
    guint left_margin_set:1;
    guint indent_set:1;
    guint rise_set:1;
    guint strikethrough_set:1;
    guint right_margin_set:1;
    guint pixels_above_lines_set:1;
    guint pixels_below_lines_set:1;
    guint pixels_inside_wrap_set:1;
    guint tabs_set:1;

```

```

    guint underline_set:1;
    guint wrap_mode_set:1;
    guint bg_full_height_set:1;
    guint invisible_set:1;
    guint editable_set:1;
    guint language_set:1;
    guint pad1:1;
    guint pad2:1;
    guint pad3:1;
} GtkTextTag;
typedef struct _GtkFileFilter GtkFileFilter;
typedef enum {
    GTK_FILE_FILTER_FILENAME = 1,
    GTK_FILE_FILTER_URI = 2,
    GTK_FILE_FILTER_DISPLAY_NAME = 4,
    GTK_FILE_FILTER_MIME_TYPE = 8
} GtkFileFilterFlags;
typedef struct _GtkFileFilterInfo {
    GtkFileFilterFlags contains;
    const gchar *filename;
    const gchar *uri;
    const gchar *display_name;
    const gchar *mime_type;
} GtkFileFilterInfo;
typedef struct _GtkIconSource GtkIconSource;
typedef struct _GtkToolItemPrivate GtkToolItemPrivate;
typedef struct _GtkToolItem {
    GtkBin parent;
    GtkToolItemPrivate *priv;
} GtkToolItem;
typedef struct _GtkToolButtonPrivate GtkToolButtonPrivate;
typedef struct _GtkToolButton {
    GtkToolItem parent;
    GtkToolButtonPrivate *priv;
} GtkToolButton;
typedef struct _GtkIconViewPrivate GtkIconViewPrivate;
typedef struct _GtkIconView {
    GtkContainer parent;
    GtkIconViewPrivate *priv;
} GtkIconView;
typedef struct _GtkLabelSelectionInfo GtkLabelSelectionInfo;
typedef struct _GtkLabel {
    GtkMisc misc;
    gchar *label;
    guint jtype:2;
    guint wrap:1;
    guint use_underline:1;
    guint use_markup:1;
    guint ellipsize:3;
    guint mnemonic_keyval;
    gchar *text;
    PangoAttrList *attrs;
    PangoAttrList *effective_attrs;
    PangoLayout *layout;
    GtkWidget *mnemonic_widget;
    GtkWindow *mnemonic_window;
    GtkLabelSelectionInfo *select_info;
} GtkLabel;
typedef struct _GtkSettingsPropertyValue GtkSettingsPropertyValue;
typedef struct _GtkRcContext GtkRcContext;
typedef struct _GtkSettings {
    GObject parent_instance;
    GData *queued_settings;
    GtkSettingsPropertyValue *property_values;
    GtkRcContext *rc_context;
    GdkScreen *screen;

```

```

} GtkSettings;
typedef struct _GtkUIManagerPrivate GtkUIManagerPrivate;
typedef struct _GtkUIManager {
    GObject parent;
    GtkUIManagerPrivate *private_data;
} GtkUIManager;
typedef struct _GtkItem {
    GtkBin bin;
} GtkItem;
typedef struct _GtkMenuItem {
    GtkItem item;
    GtkWidget *submenu;
    GdkWindow *event_window;
    guint16 toggle_size;
    guint16 accelerator_width;
    gchar *accel_path;
    quint show_submenu_indicator:1;
    quint submenu_placement:1;
    quint submenu_direction:1;
    quint right_justify:1;
    quint timer_from_keypress:1;
    quint timer;
} GtkMenuItem;
typedef struct _GtkCheckMenuItem {
    GtkMenuItem menu_item;
    quint active:1;
    quint always_show_toggle:1;
    quint inconsistent:1;
    quint draw_as_radio:1;
} GtkCheckMenuItem;
typedef struct _GtkRadioMenuItem {
    GtkCheckMenuItem check_menu_item;
    GSList *group;
} GtkRadioMenuItem;
typedef struct _GtkCellViewPrivate GtkCellViewPrivate;
typedef struct _GtkCellView {
    GtkWidget parent_instance;
    GtkCellViewPrivate *priv;
} GtkCellView;
typedef struct _GtkSelectionData {
    GdkAtom selection;
    GdkAtom target;
    GdkAtom type;
    gint format;
    guchar *data;
    gint length;
    GdkDisplay *display;
} GtkSelectionData;
typedef struct _GtkFileChooser GtkFileChooser;
typedef struct _GtkDrawingArea {
    GtkWidget widget;
    gpointer draw_data;
} GtkDrawingArea;
typedef struct _GtkCurve {
    GtkDrawingArea graph;
    gint cursor_type;
    gfloat min_x;
    gfloat max_x;
    gfloat min_y;
    gfloat max_y;
    GdkPixmap *pixmap;
    GtkCurveType curve_type;
    gint height;
    gint grab_point;
    gint last;
    gint num_points;

```



```

        GdkPoint *point;
        gint num_ctlpoints;
        gfloat *(ctlpoint)[2];
    } GtkCurve;
typedef struct _GtkNotebookPage GtkNotebookPage;
typedef struct _GtkNotebook {
    GtkContainer container;
    GtkNotebookPage *cur_page;
    GList *children;
    GList *first_tab;
    GList *focus_tab;
    GtkWidget *menu;
    GdkWindow *event_window;
    guint32 timer;
    guint16 tab_hborder;
    guint16 tab_vborder;
    guint show_tabs:1;
    guint homogeneous:1;
    guint show_border:1;
    guint tab_pos:2;
    guint scrollable:1;
    guint in_child:3;
    guint click_child:3;
    guint button:2;
    guint need_timer:1;
    guint child_has_focus:1;
    guint have_visible_child:1;
    guint focus_out:1;
    guint has_before_previous:1;
    guint has_before_next:1;
    guint has_after_previous:1;
    guint has_after_next:1;
} GtkNotebook;
typedef struct _GtkIconFactory {
    GObject parent_instance;
    GHashTable *icons;
} GtkIconFactory;
typedef struct _GtkRadioActionPrivate GtkRadioActionPrivate;
typedef struct _GtkRadioAction {
    GtkToggleAction parent;
    GtkRadioActionPrivate *private_data;
} GtkRadioAction;
typedef struct _GtkTextMark {
    GObject parent_instance;
    gpointer segment;
} GtkTextMark;
typedef struct _GtkVBox {
    GtkBox box;
} GtkVBox;
typedef struct _GtkColorSelection {
    GtkVBox parent_instance;
    gpointer private_data;
} GtkColorSelection;
typedef enum {
    GTK_CALENDAR_SHOW_HEADING = 1,
    GTK_CALENDAR_SHOW_DAY_NAMES = 2,
    GTK_CALENDAR_NO_MONTH_CHANGE = 4,
    GTK_CALENDAR_SHOW_WEEK_NUMBERS = 8,
    GTK_CALENDAR_WEEK_START_MONDAY = 16
} GtkCalendarDisplayOptions;
typedef struct _GtkCalendar {
    GtkWidget widget;
    GtkStyle *header_style;
    GtkStyle *label_style;
    gint month;
    gint year;

```

```

    gint selected_day;
    gint day_month[6][7];
    gint day[6][7];
    gint num_marked_dates;
    gint marked_date[31];
    GtkCalendarDisplayOptions display_flags;
    GdkColor marked_date_color[31];
    GdkGC *gc;
    GdkGC *xor_gc;
    gint focus_row;
    gint focus_col;
    gint highlight_row;
    gint highlight_col;
    gpointer private_data;
    gchar grow_space[32];
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCalendar;

typedef struct _GtkIconInfo GtkIconInfo;
typedef struct _GtkTreePath GtkTreePath;
typedef struct _GtkTreeDragSource GtkTreeDragSource;
typedef struct _GtkTableRowCol {
    guint16 requisition;
    guint16 allocation;
    guint16 spacing;
    guint need_expand:1;
    guint need_shrink:1;
    guint expand:1;
    guint shrink:1;
    guint empty:1;
} GtkTableRowCol;

typedef struct _GtkTable {
    GtkContainer container;
    GList *children;
    GtkTableRowCol *rows;
    GtkTableRowCol *cols;
    guint16 nrows;
    guint16 ncols;
    guint16 column_spacing;
    guint16 row_spacing;
    guint homogeneous:1;
} GtkTable;

typedef gboolean(*GtkFunction) (gpointer);
typedef void (*GtkSignalFunc) (void);
typedef struct _GtkArg GtkArg;
typedef void (*GtkCallbackMarshal) (GtkObject *, gpointer, guint,
                                     GtkArg *);

typedef struct _GtkMenuShell {
    GtkContainer container;
    GList *children;
    GtkWidget *active_menu_item;
    GtkWidget *parent_menu_shell;
    guint button;
    guint32 activate_time;
    guint active:1;
    guint have_grab:1;
    guint have_xgrab:1;
    guint ignore_leave:1;
    guint menu_flag:1;
    guint ignore_enter:1;
} GtkMenuShell;

typedef struct _GtkMenu {
    GtkMenuShell menu_shell;
    GtkWidget *parent_menu_item;

```

```

GtkWidget *old_active_menu_item;
GtkAccelGroup *accel_group;
gchar *accel_path;
GtkMenuPositionFunc position_func;
gpointer position_func_data;
guint toggle_size;
GtkWidget *toplevel;
GtkWidget *tearoff_window;
GtkWidget *tearoff_hbox;
GtkWidget *tearoff_scrollbar;
GtkAdjustment *tearoff_adjustment;
GdkWindow *view_window;
GdkWindow *bin_window;
gint scroll_offset;
gint saved_scroll_offset;
gint scroll_step;
guint timeout_id;
GdkRegion *navigation_region;
guint navigation_timeout;
guint needs_destruction_ref_count:1;
guint torn_off:1;
guint tearoff_active:1;
guint scroll_fast:1;
guint upper_arrow_visible:1;
guint lower_arrow_visible:1;
guint upper_arrow_prelight:1;
guint lower_arrow_prelight:1;
} GtkMenu;
typedef void (*GtkMenuPositionFunc) (GtkMenu *, gint *, gint *,
gboolean *,
gpointer);

typedef struct _GtkButton {
    GtkBin bin;
    GdkWindow *event_window;
    gchar *label_text;
    guint activate_timeout;
    guint constructed:1;
    guint in_button:1;
    guint button_down:1;
    guint relief:2;
    guint use_underline:1;
    guint use_stock:1;
    guint depressed:1;
    guint depress_on_activate:1;
    guint focus_on_click:1;
} GtkButton;
typedef struct _GtkToggleButton {
    GtkButton button;
    guint active:1;
    guint draw_indicator:1;
    guint inconsistent:1;
} GtkToggleButton;
typedef struct _GtkIconThemePrivate GtkIconThemePrivate;
typedef struct _GtkIconTheme {
    GObject parent_instance;
    GtkIconThemePrivate *priv;
} GtkIconTheme;
typedef struct _GtkTextWindow GtkTextWindow;
typedef struct _GtkTextPendingScroll GtkTextPendingScroll;
typedef struct _GtkTextView {
    GtkContainer parent_instance;
    struct _GtkTextLayout *layout;
    GtkTextBuffer *buffer;
    guint selection_drag_handler;
    guint scroll_timeout;
    gint pixels_above_lines;

```

```

    gint pixels_below_lines;
    gint pixels_inside_wrap;
    GtkWrapMode wrap_mode;
    GtkJustification justify;
    gint left_margin;
    gint right_margin;
    gint indent;
    PangoTabArray *tabs;
    guint editable:1;
    guint overwrite_mode:1;
    guint cursor_visible:1;
    guint need_im_reset:1;
    guint accepts_tab:1;
    guint reserved:1;
    guint onscreen_validated:1;
    guint mouse_cursor_obscured:1;
    GtkTextWindow *text_window;
    GtkTextWindow *left_window;
    GtkTextWindow *right_window;
    GtkTextWindow *top_window;
    GtkTextWindow *bottom_window;
    GtkAdjustment *hadjustment;
    GtkAdjustment *vadjustment;
    gint xoffset;
    gint yoffset;
    gint width;
    gint height;
    gint virtual_cursor_x;
    gint virtual_cursor_y;
    GtkTextMark *first_para_mark;
    gint first_para_pixels;
    GtkTextMark *dnd_mark;
    guint blink_timeout;
    guint first_validate_idle;
    guint incremental_validate_idle;
    GtkIMContext *im_context;
    GtkWidget *popup_menu;
    gint drag_start_x;
    gint drag_start_y;
    GSList *children;
    GtkTextPendingScroll *pending_scroll;
    gint pending_place_cursor_button;
} GtkTextView;
typedef enum {
    GTK_TEXT_WINDOW_PRIVATE = 0,
    GTK_TEXT_WINDOW_WIDGET = 1,
    GTK_TEXT_WINDOW_TEXT = 2,
    GTK_TEXT_WINDOW_LEFT = 3,
    GTK_TEXT_WINDOW_RIGHT = 4,
    GTK_TEXT_WINDOW_TOP = 5,
    GTK_TEXT_WINDOW_BOTTOM = 6
} GtkTextWindowType;
typedef struct _GtkObjectClass {
    GObjectClass parent_class;
    void (*set_arg) (GtkObject *, GtkArg *, guint);
    void (*get_arg) (GtkObject *, GtkArg *, guint);
    void (*destroy) (GtkObject *);
} GtkObjectClass;
typedef enum {
    GTK_WIDGET_HELP_TOOLTIP = 0,
    GTK_WIDGET_HELP_WHATS_THIS = 1
} GtkWidgetHelpType;
typedef struct _GtkWidgetClass {
    GtkObjectClass parent_class;
    guint activate_signal;
    guint set_scroll_adjustments_signal;

```

```

void (*dispatch_child_properties_changed) (GtkWidget *, guint,
                                           GParamSpec * *);

void (*show) (GtkWidget *);
void (*show_all) (GtkWidget *);
void (*hide) (GtkWidget *);
void (*hide_all) (GtkWidget *);
void (*map) (GtkWidget *);
void (*unmap) (GtkWidget *);
void (*realize) (GtkWidget *);
void (*unrealize) (GtkWidget *);
void (*size_request) (GtkWidget *, GtkRequisition *);
void (*size_allocate) (GtkWidget *, GtkAllocation *);
void (*state_changed) (GtkWidget *, GtkStateType);
void (*parent_set) (GtkWidget *, GtkWidget *);
void (*hierarchy_changed) (GtkWidget *, GtkWidget *);
void (*style_set) (GtkWidget *, GtkStyle *);
void (*direction_changed) (GtkWidget *, GtkTextDirection);
void (*grab_notify) (GtkWidget *, gboolean);
void (*child_notify) (GtkWidget *, GParamSpec *);
gboolean(*mnemonic_activate) (GtkWidget *, gboolean);
void (*grab_focus) (GtkWidget *);
gboolean(*focus) (GtkWidget *, GtkDirectionType);
gboolean(*event) (GtkWidget *, GdkEvent *);
gboolean(*button_press_event) (GtkWidget *, GdkEventButton *);
gboolean(*button_release_event) (GtkWidget *, GdkEventButton
*);
gboolean(*scroll_event) (GtkWidget *, GdkEventScroll *);
gboolean(*motion_notify_event) (GtkWidget *, GdkEventMotion *);
gboolean(*delete_event) (GtkWidget *, GdkEventAny *);
gboolean(*destroy_event) (GtkWidget *, GdkEventAny *);
gboolean(*expose_event) (GtkWidget *, GdkEventExpose *);
gboolean(*key_press_event) (GtkWidget *, GdkEventKey *);
gboolean(*key_release_event) (GtkWidget *, GdkEventKey *);
gboolean(*enter_notify_event) (GtkWidget *, GdkEventCrossing
*);
gboolean(*leave_notify_event) (GtkWidget *, GdkEventCrossing
*);
gboolean(*configure_event) (GtkWidget *, GdkEventConfigure *);
gboolean(*focus_in_event) (GtkWidget *, GdkEventFocus *);
gboolean(*focus_out_event) (GtkWidget *, GdkEventFocus *);
gboolean(*map_event) (GtkWidget *, GdkEventAny *);
gboolean(*unmap_event) (GtkWidget *, GdkEventAny *);
gboolean(*property_notify_event) (GtkWidget *,
GdkEventProperty *);
gboolean(*selection_clear_event) (GtkWidget *,
GdkEventSelection *);
gboolean(*selection_request_event) (GtkWidget *,
GdkEventSelection *);
gboolean(*selection_notify_event) (GtkWidget *,
GdkEventSelection *);
gboolean(*proximity_in_event) (GtkWidget *, GdkEventProximity
*);
gboolean(*proximity_out_event) (GtkWidget *, GdkEventProximity
*);
gboolean(*visibility_notify_event) (GtkWidget *,
GdkEventVisibility *);
gboolean(*client_event) (GtkWidget *, GdkEventClient *);
gboolean(*no_expose_event) (GtkWidget *, GdkEventAny *);
gboolean(*window_state_event) (GtkWidget *,
GdkEventWindowState *);
void (*selection_get) (GtkWidget *, GtkSelectionData *, guint,
guint);
void (*selection_received) (GtkWidget *, GtkSelectionData *,
guint);
void (*drag_begin) (GtkWidget *, GdkDragContext *);
void (*drag_end) (GtkWidget *, GdkDragContext *);

```

```

void (*drag_data_get) (GtkWidget *, GdkDragContext *,
                      GtkSelectionData *, guint, guint);
void (*drag_data_delete) (GtkWidget *, GdkDragContext *);
void (*drag_leave) (GtkWidget *, GdkDragContext *, guint);
gboolean(*drag_motion) (GtkWidget *, GdkDragContext *, gint,
gint,
                      guint);
gboolean(*drag_drop) (GtkWidget *, GdkDragContext *, gint, gint,
                      guint);
void (*drag_data_received) (GtkWidget *, GdkDragContext *, gint,
gint,
                      GtkSelectionData *, guint, guint);
gboolean(*popup_menu) (GtkWidget *);
gboolean(*show_help) (GtkWidget *, GtkWidgetHelpType);
AtkObject *(*get_accessible) (GtkWidget *);
void (*screen_changed) (GtkWidget *, GdkScreen *);
gboolean(*can_activate_accel) (GtkWidget *, guint);
gboolean(*grab_broken_event) (void);
void (*composited_changed) (GtkWidget *);
gboolean(*query_tooltip) (GtkWidget *, gint, gint, gboolean,
                      GtkTooltip *);
void (*_gtk_reserved5) (void);
void (*_gtk_reserved6) (void);
void (*_gtk_reserved7) (void);
} GtkWidgetClass;
typedef gboolean(*GtkRcPropertyParser) (const GParamSpec *,
const GString *, GValue *);

typedef enum {
    GTK_IMAGE_EMPTY = 0,
    GTK_IMAGE_PIXMAP = 1,
    GTK_IMAGE_IMAGE = 2,
    GTK_IMAGE_PIXBUF = 3,
    GTK_IMAGE_STOCK = 4,
    GTK_IMAGE_ICON_SET = 5,
    GTK_IMAGE_ANIMATION = 6,
    GTK_IMAGE_ICON_NAME = 7,
} GtkImageType;
typedef struct _GtkImagePixmapData {
    GdkPixmap *pixmap;
} GtkImagePixmapData;
typedef struct _GtkImageImageData {
    GdkImage *image;
} GtkImageImageData;
typedef struct _GtkImagePixbufData {
    GdkPixbuf *pixbuf;
} GtkImagePixbufData;
typedef struct _GtkImageStockData {
    gchar *stock_id;
} GtkImageStockData;
typedef struct _GtkImageIconSetData {
    GtkIconSet *icon_set;
} GtkImageIconSetData;
typedef struct _GtkImageAnimationData {
    GdkPixbufAnimation *anim;
    GdkPixbufAnimationIter *iter;
    guint frame_timeout;
} GtkImageAnimationData;
typedef struct _GtkImageIconNameData {
    gchar *icon_name;
    GdkPixbuf *pixbuf;
    guint theme_change_id;
} GtkImageIconNameData;
typedef struct _GtkImage {
    GtkMisc misc;
    GtkImageType storage_type;
    union {

```

```

        GtkImagePixmapData pixmap;
        GtkImageImageData image;
        GtkImagePixbufData pixbuf;
        GtkImageStockData stock;
        GtkImageIconSetData icon_set;
        GtkImageAnimationData anim;
        GtkImageIconNameData name;
    } data;
    GdkBitmap *mask;
    GtkIconSize icon_size;
} GtkImage;
typedef struct _GtkTreeViewPrivate GtkTreeViewPrivate;
typedef struct _GtkTreeView {
    GtkContainer parent;
    GtkTreeViewPrivate *priv;
} GtkTreeView;
typedef enum {
    GTK_UPDATE_ALWAYS = 0,
    GTK_UPDATE_IF_VALID = 1
} GtkSpinButtonUpdatePolicy;
typedef struct _GtkSpinButton {
    GtkEntry entry;
    GtkAdjustment *adjustment;
    GdkWindow *panel;
    guint32 timer;
    gdouble climb_rate;
    gdouble timer_step;
    GtkSpinButtonUpdatePolicy update_policy;
    guint in_child:2;
    guint click_child:2;
    guint button:2;
    guint need_timer:1;
    guint timer_calls:3;
    guint digits:10;
    guint numeric:1;
    guint wrap:1;
    guint snap_to_ticks:1;
} GtkSpinButton;
typedef struct _GtkCellLayout GtkCellLayout;
typedef struct _GtkFileSelection {
    GtkDialog parent_instance;
    GtkWidget *dir_list;
    GtkWidget *file_list;
    GtkWidget *selection_entry;
    GtkWidget *selection_text;
    GtkWidget *main_vbox;
    GtkWidget *ok_button;
    GtkWidget *cancel_button;
    GtkWidget *help_button;
    GtkWidget *history_pulldown;
    GtkWidget *history_menu;
    GList *history_list;
    GtkWidget *fileop_dialog;
    GtkWidget *fileop_entry;
    gchar *fileop_file;
    gpointer cpl_state;
    GtkWidget *fileop_c_dir;
    GtkWidget *fileop_del_file;
    GtkWidget *fileop_ren_file;
    GtkWidget *button_area;
    GtkWidget *action_area;
    GPtrArray *selected_names;
    gchar *last_selected;
} GtkFileSelection;
typedef struct _GtkTooltips {
    GObject parent_instance;

```

```

    GtkWidget *tip_window;
    GtkWidget *tip_label;
    GtkTooltipsData *active_tips_data;
    GList *tips_data_list;
    guint delay:30;
    guint enabled:1;
    guint have_grab:1;
    guint use_sticky_delay:1;
    gint timer_tag;
    GTimeVal last_popdown;
} GtkTooltips;
typedef struct _GtkTooltipsData {
    GtkTooltips *tooltips;
    GtkWidget *widget;
    gchar *tip_text;
    gchar *tip_private;
} GtkTooltipsData;
typedef gboolean(*GtkTreeViewRowSeparatorFunc) (GtkTreeModel *,
                                                GtkTreeIter *, gpointer);

typedef struct _GtkEntryCompletionPrivate
GtkEntryCompletionPrivate;
typedef struct _GtkEntryCompletion {
    GObject parent_instance;
    GtkEntryCompletionPrivate *priv;
} GtkEntryCompletion;
typedef struct _GtkClipboard GtkClipboard;
typedef struct _GtkTreeModelFilterPrivate
GtkTreeModelFilterPrivate;
typedef struct _GtkTreeModelFilter {
    GObject parent;
    GtkTreeModelFilterPrivate *priv;
} GtkTreeModelFilter;
typedef struct _GtkTreeModelSort {
    GObject parent;
    gpointer root;
    gint stamp;
    guint child_flags;
    GtkTreeModel *child_model;
    gint zero_ref_count;
    GList *sort_list;
    gint sort_column_id;
    GtkSortType order;
    GtkTreeIterCompareFunc default_sort_func;
    gpointer default_sort_data;
    GtkDestroyNotify default_sort_destroy;
    guint changed_id;
    guint inserted_id;
    guint has_child_toggled_id;
    guint deleted_id;
    guint reordered_id;
} GtkTreeModelSort;
typedef void (*GtkTreeCellDataFunc) (GtkTreeViewColumn *,
                                     GtkCellRenderer *, GtkTreeModel *,
                                     GtkTreeIter *, gpointer);

typedef struct _GtkColorButtonPrivate GtkColorButtonPrivate;
typedef struct _GtkColorButton {
    GtkButton button;
    GtkColorButtonPrivate *priv;
} GtkColorButton;
typedef enum {
    GTK_FILE_CHOOSER_ACTION_OPEN = 0,
    GTK_FILE_CHOOSER_ACTION_SAVE = 1,
    GTK_FILE_CHOOSER_ACTION_SELECT_FOLDER = 2,
    GTK_FILE_CHOOSER_ACTION_CREATE_FOLDER = 3
} GtkFileChooserAction;
typedef void (*GtkCallback) (GtkWidget *, gpointer);

```



```

typedef struct _GtkHandleBox {
    GtkBin bin;
    GdkWindow *bin_window;
    GdkWindow *float_window;
    GtkShadowType shadow_type;
    guint handle_position:2;
    guint float_window_mapped:1;
    guint child_detached:1;
    guint in_drag:1;
    guint shrink_on_detach:1;
    int snap_edge:3;
    gint deskoff_x;
    gint deskoff_y;
    GtkAllocation attach_allocation;
    GtkAllocation float_allocation;
} GtkHandleBox;
typedef struct _GtkActionGroupPrivate GtkActionGroupPrivate;
typedef struct _GtkActionGroup {
    GObject parent;
    GtkActionGroupPrivate *private_data;
} GtkActionGroup;
typedef struct _GtkBindingSet {
    gchar *set_name;
    gint priority;
    GSList *widget_path_specs;
    GSList *widget_class_specs;
    GSList *class_branch_specs;
    GtkBindingEntry *entries;
    GtkBindingEntry *current;
    guint parsed:1;
} GtkBindingSet;
typedef struct _GtkBindingEntry {
    guint keyval;
    GdkModifierType modifiers;
    GtkBindingSet *binding_set;
    guint destroyed:1;
    guint in_emission:1;
    GtkBindingEntry *set_next;
    GtkBindingEntry *hash_next;
    GtkBindingSignal *signals;
} GtkBindingEntry;
typedef struct _GtkBindingSignal {
    GtkBindingSignal *next;
    gchar *signal_name;
    guint n_args;
    GtkBindingArg *args;
} GtkBindingSignal;
typedef struct _GtkBindingArg {
    GType arg_type;
    union {
        glong long_data;
        gdouble double_data;
        gchar *string_data;
    } d;
} GtkBindingArg;
typedef struct _GtkScrolledWindow {
    GtkBin container;
    GtkWidget *hscrollbar;
    GtkWidget *vscrollbar;
    guint hscrollbar_policy:2;
    guint vscrollbar_policy:2;
    guint hscrollbar_visible:1;
    guint vscrollbar_visible:1;
    guint window_placement:2;
    guint focus_out:1;
    guint16 shadow_type;

```

```

} GtkScrolledWindow;
typedef struct _GtkFrame {
    GtkBin bin;
    GtkWidget *label_widget;
    gint16 shadow_type;
    gfloat label_xalign;
    gfloat label_yalign;
    GtkAllocation child_allocation;
} GtkFrame;
typedef struct _GtkTextChildAnchor {
    GObject parent_instance;
    gpointer segment;
} GtkTextChildAnchor;
typedef struct _GtkSeparatorToolItemPrivate
GtkSeparatorToolItemPrivate;
typedef struct _GtkSeparatorToolItem {
    GtkToolItem parent;
    GtkSeparatorToolItemPrivate *priv;
} GtkSeparatorToolItem;
typedef struct _GtkTreeSelection {
    GObject parent;
    GtkTreeView *tree_view;
    GtkSelectionMode type;
    GtkTreeSelectionFunc user_func;
    gpointer user_data;
    GtkDestroyNotify destroy;
} GtkTreeSelection;
typedef gboolean(*GtkTreeSelectionFunc) (GtkTreeSelection *,
                                           GtkTreeModel *, GtkTreePath *,
                                           gboolean, gpointer);
typedef struct _GtkExpanderPrivate GtkExpanderPrivate;
typedef struct _GtkExpander {
    GtkBin bin;
    GtkExpanderPrivate *priv;
} GtkExpander;
typedef enum {
    GTK_UI_MANAGER_AUTO = 0,
    GTK_UI_MANAGER_MENUBAR = 1,
    GTK_UI_MANAGER_MENU = 2,
    GTK_UI_MANAGER_TOOLBAR = 4,
    GTK_UI_MANAGER_PLACEHOLDER = 8,
    GTK_UI_MANAGER_POPUP = 16,
    GTK_UI_MANAGER_MENUITEM = 32,
    GTK_UI_MANAGER_TOOLITEM = 64,
    GTK_UI_MANAGER_SEPARATOR = 128,
    GTK_UI_MANAGER_ACCELERATOR = 256
} GtkUIManagerItemType;
typedef struct _GtkFontSelectionDialog {
    GtkDialog parent_instance;
    GtkWidget *fontsel;
    GtkWidget *main_vbox;
    GtkWidget *action_area;
    GtkWidget *ok_button;
    GtkWidget *apply_button;
    GtkWidget *cancel_button;
    gint dialog_width;
    gboolean auto_resize;
} GtkFontSelectionDialog;
typedef struct _GtkPanedPrivate GtkPanedPrivate;
typedef struct _GtkPaned {
    GtkContainer container;
    GtkWidget *child1;
    GtkWidget *child2;
    GdkWindow *handle;
    GdkGC *xor_gc;
    GdkCursorType cursor_type;

```

```

    GdkRectangle handle_pos;
    gint child1_size;
    gint last_allocation;
    gint min_position;
    gint max_position;
    guint position_set:1;
    guint in_drag:1;
    guint child1_shrink:1;
    guint child1_resize:1;
    guint child2_shrink:1;
    guint child2_resize:1;
    guint orientation:1;
    guint in_recursion:1;
    guint handle_prelit:1;
    GtkWidget *last_child1_focus;
    GtkWidget *last_child2_focus;
    GtkPanedPrivate *priv;
    gint drag_pos;
    gint original_position;
} GtkPaned;
typedef struct _GtkViewport {
    GtkBin bin;
    GtkShadowType shadow_type;
    GdkWindow *view_window;
    GdkWindow *bin_window;
    GtkAdjustment *hadjustment;
    GtkAdjustment *vadjustment;
} GtkViewport;
typedef struct _GtkScale {
    GtkRange range;
    gint digits;
    guint draw_value:1;
    guint value_pos:2;
} GtkScale;
typedef struct _GtkListStore {
    GObject parent;
    gint stamp;
    gpointer seq;
    gpointer _gtk_reserved1;
    GList *sort_list;
    gint n_columns;
    gint sort_column_id;
    GtkSortType order;
    GType *column_headers;
    gint length;
    GtkTreeIterCompareFunc default_sort_func;
    gpointer default_sort_data;
    GtkDestroyNotify default_sort_destroy;
    guint columns_dirty:1;
} GtkListStore;
typedef struct _GtkEditable GtkEditable;
typedef void (*GtkClipboardGetFunc) (GtkClipboard *,
    gpointer, gpointer);
typedef void (*GtkClipboardClearFunc) (GtkClipboard *, gpointer);
typedef struct _GtkRadioActionEntry {
    const gchar *name;
    const gchar *stock_id;
    const gchar *label;
    const gchar *accelerator;
    const gchar *tooltip;
    gint value;
} GtkRadioActionEntry;
typedef gboolean(*GtkTreeViewColumnDropFunc) (GtkTreeView *,
    GtkTreeViewColumn *,
    GtkTreeViewColumn *,

```

```

GtkTreeViewColumn *,
gpointer);

typedef struct _GtkEventBox {
    GtkBin bin;
} GtkEventBox;
typedef struct _GtkTreeDragDest GtkTreeDragDest;
typedef struct _GtkToggleActionEntry {
    const gchar *name;
    const gchar *stock_id;
    const gchar *label;
    const gchar *accelerator;
    const gchar *tooltip;
    GCallback callback;
    gboolean is_active;
} GtkToggleActionEntry;
typedef struct _GtkAccelLabel {
    GtkLabel label;
    guint gtk_reserved;
    guint accel_padding;
    GtkWidget *accel_widget;
    GClosure *accel_closure;
    GtkAccelGroup *accel_group;
    gchar *accel_string;
    guint16 accel_string_width;
} GtkAccelLabel;
typedef struct _GtkImageMenuItem {
    GtkMenuItem menu_item;
    GtkWidget *image;
} GtkImageMenuItem;
typedef struct _GtkPlug {
    GtkWindow window;
    GdkWindow *socket_window;
    GtkWidget *modality_window;
    GtkWindowGroup *modality_group;
    GHashTable *grabbed_keys;
    guint same_app:1;
} GtkPlug;
typedef struct _GtkFileChooserButtonPrivate
GtkFileChooserButtonPrivate;
typedef struct _GtkFileChooserButton {
    GtkHBox parent;
    GtkFileChooserButtonPrivate *priv;
} GtkFileChooserButton;
typedef struct _GtkStockItem {
    gchar *stock_id;
    gchar *label;
    GdkModifierType modifier;
    guint keyval;
    gchar *translation_domain;
} GtkStockItem;
typedef gchar *(*GtkTranslateFunc) (const gchar *, gpointer);
typedef struct _GtkToolbar {
    GtkContainer container;
    gint num_children;
    GList *children;
    GtkOrientation orientation;
    GtkToolbarStyle style;
    GtkIconSize icon_size;
    GtkTooltips *tooltips;
    gint button_maxw;
    gint button_maxh;
    guint style_set_connection;
    guint icon_size_connection;
    guint style_set:1;
    guint icon_size_set:1;
} GtkToolbar;

```

```

typedef void (*GtkCellLayoutDataFunc) (GtkCellLayout *,
GtkCellRenderer *,
                                GtkTreeModel *, GtkTreeIter *,
                                gpointer);
typedef struct _GtkCellRendererToggle {
    GtkCellRenderer parent;
    guint active:1;
    guint activatable:1;
    guint radio:1;
} GtkCellRendererToggle;
typedef void (*GtkClipboardImageReceivedFunc) (GtkClipboard *,
GdkPixbuf *,
                                gpointer);

typedef enum {
    GTK_CELL_RENDERER_SELECTED = 1,
    GTK_CELL_RENDERER_PRELIT = 2,
    GTK_CELL_RENDERER_INSENSITIVE = 4,
    GTK_CELL_RENDERER_SORTED = 8,
    GTK_CELL_RENDERER_FOCUSED = 16
} GtkCellRendererState;
typedef struct _GtkActionEntry {
    const gchar *name;
    const gchar *stock_id;
    const gchar *label;
    const gchar *accelerator;
    const gchar *tooltip;
    GCallback callback;
} GtkActionEntry;
typedef struct _GtkCheckButton {
    GtkToggleButton toggle_button;
} GtkCheckButton;
typedef struct _GtkRadioButton {
    GtkCheckButton check_button;
    GSList *group;
} GtkRadioButton;
typedef struct _GtkAlignment {
    GtkBin bin;
    gfloat xalign;
    gfloat yalign;
    gfloat xscale;
    gfloat yscale;
} GtkAlignment;
typedef struct _GtkContainerClass {
    GtkWidgetClass parent_class;
    void (*add) (GtkContainer *, GtkWidget *);
    void (*remove) (GtkContainer *, GtkWidget *);
    void (*check_resize) (GtkContainer *);
    void (*forall) (GtkContainer *, gboolean, GtkCallback,
gpointer);
    void (*set_focus_child) (GtkContainer *, GtkWidget *);
    GType(*child_type) (GtkContainer *);
    gchar *(*composite_name) (GtkContainer *, GtkWidget *);
    void (*set_child_property) (GtkContainer *, GtkWidget *, guint,
                                const GValue *, GParamSpec *);
    void (*get_child_property) (GtkContainer *, GtkWidget *, guint,
                                GValue *, GParamSpec *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkContainerClass;
typedef struct _GtkFontButtonPrivate GtkFontButtonPrivate;
typedef struct _GtkFontButton {
    GtkButton button;
    GtkFontButtonPrivate *priv;
} GtkFontButton;

```

```

typedef struct _GtkBorder {
    gint left;
    gint right;
    gint top;
    gint bottom;
} GtkBorder;
typedef struct _GtkMessageDialog {
    GtkDialog parent_instance;
    GtkWidget *image;
    GtkWidget *label;
} GtkMessageDialog;
typedef struct _GtkRulerMetric {
    gchar *metric_name;
    gchar *abbrev;
    gdouble pixels_per_unit;
    gdouble ruler_scale[10];
    gint subdivide[5];
} GtkRulerMetric;
typedef struct _GtkRuler {
    GtkWidget widget;
    GdkPixmap *backing_store;
    GdkGC *non_gr_exp_gc;
    GtkRulerMetric *metric;
    gint xsrc;
    gint ysrc;
    gint slider_size;
    gdouble lower;
    gdouble upper;
    gdouble position;
    gdouble max_size;
} GtkRuler;
typedef enum {
    GTK_ACCEL_VISIBLE = 1,
    GTK_ACCEL_LOCKED = 2,
    GTK_ACCEL_MASK = 7
} GtkAccelFlags;
typedef gboolean(*GtkTextCharPredicate) (gunichar, gpointer);
typedef struct _GtkMenuToolButtonPrivate GtkMenuToolButtonPrivate;
typedef struct _GtkMenuToolButton {
    GtkToolButton parent;
    GtkMenuToolButtonPrivate *priv;
} GtkMenuToolButton;
typedef struct _GtkToggleToolButtonPrivate
GtkToggleToolButtonPrivate;
typedef struct _GtkToggleToolButton {
    GtkToolButton parent;
    GtkToggleToolButtonPrivate *priv;
} GtkToggleToolButton;
typedef struct _GtkRadioToolButton {
    GtkToggleToolButton parent;
} GtkRadioToolButton;
typedef struct _GtkSizeGroup {
    GObject parent_instance;
    GSList *widgets;
    guint8 mode;
    guint have_width:1;
    guint have_height:1;
    GtkRequisition requisition;
} GtkSizeGroup;
typedef enum {
    GTK_SIZE_GROUP_NONE = 0,
    GTK_SIZE_GROUP_HORIZONTAL = 1,
    GTK_SIZE_GROUP_VERTICAL = 2,
    GTK_SIZE_GROUP_BOTH = 3
} GtkSizeGroupMode;
typedef enum {

```

```

    GTK_DIALOG_MODAL = 1,
    GTK_DIALOG_DESTROY_WITH_PARENT = 2,
    GTK_DIALOG_NO_SEPARATOR = 4
} GtkDialogFlags;
typedef void (*GtkTreeSelectionForeachFunc) (GtkTreeModel *,
GtkTreePath *,
                                           GtkTreeIter *, gpointer);

typedef enum {
    GTK_ICON_LOOKUP_NO_SVG = 1,
    GTK_ICON_LOOKUP_FORCE_SVG = 2,
    GTK_ICON_LOOKUP_USE_BUILTIN = 4
} GtkIconLookupFlags;
typedef gboolean(*GtkTreeViewSearchEqualFunc) (GtkTreeModel *,
gint,
                                           const gchar *,
                                           GtkTreeIter *, gpointer);

typedef struct _GtkIMContextSimple {
    GtkIMContext object;
    GSList *tables;
    guint compose_buffer[8];
    gunichar tentative_match;
    gint tentative_match_len;
    guint in_hex_sequence:1;
} GtkIMContextSimple;
typedef void (*GtkClipboardTargetsReceivedFunc) (GtkClipboard *,
GdkAtom *,
                                           gint, gpointer);

typedef struct _GtkFontSelection {
    GtkVBox parent_instance;
    GtkWidget *font_entry;
    GtkWidget *family_list;
    GtkWidget *font_style_entry;
    GtkWidget *face_list;
    GtkWidget *size_entry;
    GtkWidget *size_list;
    GtkWidget *pixels_button;
    GtkWidget *points_button;
    GtkWidget *filter_button;
    GtkWidget *preview_entry;
    PangoFontFamily *family;
    PangoFontFace *face;
    gint size;
    GdkFont *font;
} GtkFontSelection;
typedef void (*GtkClipboardTextReceivedFunc) (GtkClipboard *,
                                           const gchar *, gpointer);
typedef gboolean(*GtkAccelGroupFindFunc) (GtkAccelKey *, GClosure
*,
                                           gpointer);
typedef void (*GtkTreeDestroyCountFunc) (GtkTreeView *, GtkTreePath
*,
                                           gint, gpointer);

typedef enum {
    GTK_TREE_VIEW_DROP_BEFORE = 0,
    GTK_TREE_VIEW_DROP_AFTER = 1,
    GTK_TREE_VIEW_DROP_INTRO_OR_BEFORE = 2,
    GTK_TREE_VIEW_DROP_INTRO_OR_AFTER = 3
} GtkTreeViewDropPosition;
typedef struct _GtkAspectFrame {
    GtkFrame frame;
    gfloat xalign;
    gfloat yalign;
    gfloat ratio;
    gboolean obey_child;
    GtkAllocation center_allocation;
} GtkAspectFrame;

```

```

typedef gboolean(*GtkFileFilterFunc) (const GtkFileFilterInfo *
                                     filter_info, gpointer data);
typedef struct _GtkIMMulticontextPrivate GtkIMMulticontextPrivate;
typedef struct _GtkIMMulticontext {
    GtkIMContext object;
    GtkIMContext *slave;
    GtkIMMulticontextPrivate *priv;
    const gchar *context_id;
} GtkIMMulticontext;
typedef enum {
    GTK_BUTTONS_NONE = 0,
    GTK_BUTTONS_OK = 1,
    GTK_BUTTONS_CLOSE = 2,
    GTK_BUTTONS_CANCEL = 3,
    GTK_BUTTONS_YES_NO = 4,
    GTK_BUTTONS_OK_CANCEL = 5
} GtkButtonType;
typedef void (*GtkAccelMapForeach) (gpointer, const gchar *, guint,
                                     GdkModifierType, gboolean);
typedef void (*GtkColorSelectionChangePaletteWithScreenFunc)
(GdkScreen *,
 const
 GdkColor *,
 gint);
typedef void (*GtkClipboardReceivedFunc) (GtkClipboard *,
                                         GtkSelectionData *, gpointer);
typedef void (*GtkTreeViewMappingFunc) (GtkTreeView *, GtkTreePath
*,
                                         gpointer);
typedef gboolean(*GtkTreeModelFilterVisibleFunc) (GtkTreeModel *,
                                                  GtkTreeIter *, gpointer);
typedef enum {
    GTK_TEXT_SEARCH_VISIBLE_ONLY = 1,
    GTK_TEXT_SEARCH_TEXT_ONLY = 2
} GtkTextSearchFlags;
typedef struct _GtkFixed {
    GtkContainer container;
    GList *children;
} GtkFixed;
typedef enum {
    GTK_DEST_DEFAULT_MOTION = 1,
    GTK_DEST_DEFAULT_HIGHLIGHT = 2,
    GTK_DEST_DEFAULT_DROP = 4,
    GTK_DEST_DEFAULT_ALL = 7
} GtkDestDefaults;
typedef gint(*GtkKeySnoopFunc) (GtkWidget *, GdkEventKey *,
                                gpointer);
typedef enum {
    GTK_SPIN_STEP_FORWARD = 0,
    GTK_SPIN_STEP_BACKWARD = 1,
    GTK_SPIN_PAGE_FORWARD = 2,
    GTK_SPIN_PAGE_BACKWARD = 3,
    GTK_SPIN_HOME = 4,
    GTK_SPIN_END = 5,
    GTK_SPIN_USER_DEFINED = 6
} GtkSpinType;
typedef void (*GtkMenuDetachFunc) (GtkWidget *, GtkMenu *);
typedef struct _GtkInvisible {
    GtkWidget widget;
    gboolean has_user_ref_count;
    GdkScreen *screen;
} GtkInvisible;
typedef void (*GtkTextTagTableForeach) (GtkTextTag *, gpointer);
typedef gboolean(*GtkEntryCompletionMatchFunc) (GtkEntryCompletion
*,
const gchar *,

```



```

GtkTreeIter *, gpointer);

typedef struct _GtkAccessible {
    AtkObject parent;
    GtkWidget *widget;
} GtkAccessible;
typedef struct _GtkArrow {
    GtkMisc misc;
    gint16 arrow_type;
    gint16 shadow_type;
} GtkArrow;
typedef void (*GtkIconViewForeachFunc) (GtkIconView *, GtkTreePath
*,
                                     gpointer);
typedef void (*GtkTreeModelFilterModifyFunc) (GtkTreeModel *,
                                              GtkTreeIter *, GValue *,
                                              gint, gpointer);

typedef struct _GtkCellRendererText {
    GtkCellRenderer parent;
    gchar *text;
    PangoFontDescription *font;
    gdouble font_scale;
    PangoColor foreground;
    PangoColor background;
    PangoAttrList *extra_attrs;
    PangoUnderline underline_style;
    gint rise;
    gint fixed_height_rows;
    guint strikethrough:1;
    guint editable:1;
    guint scale_set:1;
    guint foreground_set:1;
    guint background_set:1;
    guint underline_set:1;
    guint rise_set:1;
    guint strikethrough_set:1;
    guint editable_set:1;
    guint calc_fixed_height:1;
} GtkCellRendererText;
typedef gboolean(*GtkTreeModelForeachFunc) (GtkTreeModel *,
                                             GtkTreePath *,
                                             GtkTreeIter *, gpointer);

typedef struct _GtkSettingsValue {
    gchar *origin;
    GValue value;
} GtkSettingsValue;
typedef struct _GtkAccelMap GtkAccelMap;
typedef struct _GtkBinClass {
    GtkContainerClass parent_class;
} GtkBinClass;
typedef struct _GtkWindowClass {
    GtkBinClass parent_class;
    void (*set_focus) (GtkWindow *, GtkWidget *);
    gboolean(*frame_event) (GtkWindow *, GdkEvent *);
    void (*activate_focus) (GtkWindow *);
    void (*activate_default) (GtkWindow *);
    void (*move_focus) (GtkWindow *, GtkDirectionType);
    void (*keys_changed) (GtkWindow *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkWindowClass;
typedef struct _GtkRangeClass {
    GtkWidgetClass parent_class;
    gchar *slider_detail;
    gchar *stepper_detail;

```

```

    void (*value_changed) (GtkRange *);
    void (*adjust_bounds) (GtkRange *, gdouble);
    void (*move_slider) (GtkRange *, GtkScrollType);
    void (*get_range_border) (GtkRange *, GtkBorder *);
    gboolean(*change_value) (GtkRange *, GtkScrollType, gdouble);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
} GtkRangeClass;
typedef struct _GtkScaleClass {
    GtkRangeClass parent_class;
    gchar *(*format_value) (GtkScale *, gdouble);
    void (*draw_value) (GtkScale *);
    void (*get_layout_offsets) (GtkScale *, gint *, gint *);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkScaleClass;
typedef struct _GtkBoxClass {
    GtkContainerClass parent_class;
} GtkBoxClass;
typedef struct _GtkMiscClass {
    GtkWidgetClass parent_class;
} GtkMiscClass;
typedef struct _GtkHRuler {
    GtkRuler ruler;
} GtkHRuler;
typedef struct _GtkHBoxClass {
    GtkBoxClass parent_class;
} GtkHBoxClass;
typedef struct _GtkPanedClass {
    GtkContainerClass parent_class;
    gboolean(*cycle_child_focus) (GtkPaned *, gboolean);
    gboolean(*toggle_handle_focus) (GtkPaned *);
    gboolean(*move_handle) (GtkPaned *, GtkScrollType);
    gboolean(*cycle_handle_focus) (GtkPaned *, gboolean);
    gboolean(*accept_position) (GtkPaned *);
    gboolean(*cancel_position) (GtkPaned *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkPanedClass;
typedef struct _GtkHPanedClass {
    GtkPanedClass parent_class;
} GtkHPanedClass;
typedef struct _GtkSizeGroupClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkSizeGroupClass;
typedef struct _GtkViewportClass {
    GtkBinClass parent_class;
    void (*set_scroll_adjustments) (GtkViewport *, GtkAdjustment *,
                                    GtkAdjustment *);
} GtkViewportClass;
typedef struct _GtkEditableClass {
    GTypeInterface base_iface;
    void (*insert_text) (GtkEditable *, const gchar *, gint, gint
*);
    void (*delete_text) (GtkEditable *, gint, gint);
    void (*changed) (GtkEditable *);
    void (*do_insert_text) (GtkEditable *, const gchar *, gint, gint
*);

```

```

void (*do_delete_text) (GtkEditable *, gint, gint);
gchar *(*get_chars) (GtkEditable *, gint, gint);
void (*set_selection_bounds) (GtkEditable *, gint, gint);
gboolean(*get_selection_bounds) (GtkEditable *, gint *, gint
*);
void (*set_position) (GtkEditable *, gint);
gint(*get_position) (GtkEditable *);
} GtkEditableClass;
typedef enum {
    GTK_NOTEBOOK_TAB_FIRST = 0,
    GTK_NOTEBOOK_TAB_LAST = 1
} GtkNotebookTab;
typedef struct _GtkDialogClass {
    GtkWindowClass parent_class;
    void (*response) (GtkDialog *, gint);
    void (*close) (GtkDialog *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkDialogClass;
typedef struct _GtkCellRendererPixbuf {
    GtkCellRenderer parent;
    GdkPixbuf *pixbuf;
    GdkPixbuf *pixbuf_expander_open;
    GdkPixbuf *pixbuf_expander_closed;
} GtkCellRendererPixbuf;
typedef struct _GtkTreeViewColumnClass {
    GObjectClass parent_class;
    void (*clicked) (GtkTreeViewColumn *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTreeViewColumnClass;
typedef struct _GtkRulerClass {
    GtkWidgetClass parent_class;
    void (*draw_ticks) (GtkRuler *);
    void (*draw_pos) (GtkRuler *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkRulerClass;
typedef struct _GtkHRulerClass {
    GtkRulerClass parent_class;
} GtkHRulerClass;
typedef struct _GtkAccelMapClass GtkAccelMapClass;
typedef struct _GtkInvisibleClass {
    GtkWidgetClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkInvisibleClass;
typedef struct _GtkVRulerClass {
    GtkRulerClass parent_class;
} GtkVRulerClass;
typedef struct _GtkToolItemClass {
    GtkBinClass parent_class;
    gboolean(*create_menu_proxy) (GtkToolItem *);
    void (*toolbar_reconfigured) (GtkToolItem *);
    gboolean(*set_tooltip) (GtkToolItem *, GtkTooltips *, const
gchar *,
                                const gchar *);
    void (*_gtk_reserved1) (void);

```

```

        void (*_gtk_reserved2) (void);
        void (*_gtk_reserved3) (void);
        void (*_gtk_reserved4) (void);
    } GtkToolItemClass;
typedef struct _GtkWindowGroupClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkWindowGroupClass;
typedef struct _GtkWidgetAuxInfo {
    gint x;
    gint y;
    gint width;
    gint height;
    guint x_set:1;
    guint y_set:1;
} GtkWidgetAuxInfo;
typedef struct _GtkItemClass {
    GtkBinClass parent_class;
    void (*select) (GtkItem *);
    void (*deselect) (GtkItem *);
    void (*toggle) (GtkItem *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkItemClass;
typedef struct _GtkMenuItemClass {
    GtkItemClass parent_class;
    guint hide_on_activate:1;
    void (*activate) (GtkMenuItem *);
    void (*activate_item) (GtkMenuItem *);
    void (*toggle_size_request) (GtkMenuItem *, gint *);
    void (*toggle_size_allocate) (GtkMenuItem *, gint);
    void (*set_label) (void);
    const gchar *(*get_label) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkMenuItemClass;
typedef struct _GtkCheckMenuItemClass {
    GtkMenuItemClass parent_class;
    void (*toggled) (GtkCheckMenuItem *);
    void (*draw_indicator) (GtkCheckMenuItem *, GdkRectangle *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCheckMenuItemClass;
typedef struct _GtkRcProperty {
    GQuark type_name;
    GQuark property_name;
    gchar *origin;
    GValue value;
} GtkRcProperty;
typedef struct _GtkFontSelectionDialogClass {
    GtkDialogClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkFontSelectionDialogClass;
typedef struct _GtkFileChooserDialogPrivate
GtkFileChooserDialogPrivate;
typedef struct _GtkFileChooserDialog {

```

```

    GtkDialog parent_instance;
    GtkFileChooserDialogPrivate *priv;
} GtkFileChooserDialog;
typedef struct _GtkActionClass {
    GObjectClass parent_class;
    void (*activate) (GtkAction *);
    GType menu_item_type;
    GType toolbar_item_type;
    GtkWidget *(*create_menu_item) (GtkAction *);
    GtkWidget *(*create_tool_item) (GtkAction *);
    void (*connect_proxy) (GtkAction *, GtkWidget *);
    void (*disconnect_proxy) (GtkAction *, GtkWidget *);
    GtkWidget *(*create_menu) (GtkAction *);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkActionClass;
typedef struct _GtkToggleActionClass {
    GtkActionClass parent_class;
    void (*toggled) (GtkToggleAction *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkToggleActionClass;
typedef struct _GtkVBoxClass {
    GtkBoxClass parent_class;
} GtkVBoxClass;
typedef struct _GtkButtonBoxClass {
    GtkBoxClass parent_class;
} GtkButtonBoxClass;
typedef struct _GtkHButtonBoxClass {
    GtkButtonBoxClass parent_class;
} GtkHButtonBoxClass;
typedef struct _GtkThemeEngine GtkThemeEngine;
typedef struct _GtkTreeStoreClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTreeStoreClass;
typedef struct _GtkBoxChild {
    GtkWidget *widget;
    guint16 padding;
    guint expand:1;
    guint fill:1;
    guint pack:1;
    guint is_secondary:1;
} GtkBoxChild;
typedef struct _GtkHButtonBox {
    GtkButtonBox button_box;
} GtkHButtonBox;
typedef struct _GtkAccelGroupClass {
    GObjectClass parent_class;
    void (*accel_changed) (GtkAccelGroup *, guint, GdkModifierType,
                           GClosure *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkAccelGroupClass;
typedef struct _GtkSeparator {
    GtkWidget widget;
} GtkSeparator;
typedef struct _GtkButtonClass {

```

```

    GtkBinClass parent_class;
    void (*pressed) (GtkButton *);
    void (*released) (GtkButton *);
    void (*clicked) (GtkButton *);
    void (*enter) (GtkButton *);
    void (*leave) (GtkButton *);
    void (*activate) (GtkButton *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkButtonClass;
typedef void (*GtkWindowKeysForeachFunc) (GtkWindow *, guint,
                                           GdkModifierType, gboolean,
                                           gpointer);

typedef struct _GtkIconFactoryClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkIconFactoryClass;
typedef struct _GtkTreeSelectionClass {
    GObjectClass parent_class;
    void (*changed) (GtkTreeSelection *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTreeSelectionClass;
typedef struct _GtkMenuShellClass {
    GtkContainerClass parent_class;
    guint submenu_placement:1;
    void (*deactivate) (GtkMenuShell *);
    void (*selection_done) (GtkMenuShell *);
    void (*move_current) (GtkMenuShell *, GtkMenuDirectionType);
    void (*activate_current) (GtkMenuShell *, gboolean);
    void (*cancel) (GtkMenuShell *);
    void (*select_item) (GtkMenuShell *, GtkWidget *);
    void (*insert) (GtkMenuShell *, GtkWidget *, gint);
    gint (*get_popup_delay) (GtkMenuShell *);
    gboolean (*move_selected) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
} GtkMenuShellClass;
typedef struct _GtkTreeDragSourceIface {
    GTypeInterface g_iface;
    gboolean (*row_draggable) (GtkTreeDragSource *, GtkTreePath *);
    gboolean (*drag_data_get) (GtkTreeDragSource *, GtkTreePath *,
                              GtkSelectionData *);
    gboolean (*drag_data_delete) (GtkTreeDragSource *, GtkTreePath
    *);
} GtkTreeDragSourceIface;
typedef struct _GtkLabelClass {
    GtkMiscClass parent_class;
    void (*move_cursor) (GtkLabel *, GtkMovementStep, gint,
    gboolean);
    void (*copy_clipboard) (GtkLabel *);
    void (*populate_popup) (GtkLabel *, GtkMenu *);
    gboolean (*activate_link) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkLabelClass;
typedef struct _GtkScrollbar {
    GtkRange range;

```

```

} GtkWidgetScrollbar;
typedef struct _GtkWidgetShapeInfo {
    gint16 offset_x;
    gint16 offset_y;
    GdkBitmap *shape_mask;
} GtkWidgetShapeInfo;
typedef struct _GtkActionGroupClass {
    GObjectClass parent_class;
    GtkAction *(*get_action) (GtkActionGroup *, const gchar *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkActionGroupClass;
typedef struct _GtkSettingsClass {
    GObjectClass parent_class;
} GtkSettingsClass;
typedef struct _GtkIMContextClass {
    GObjectClass parent_class;
    void (*preedit_start) (GtkIMContext *);
    void (*preedit_end) (GtkIMContext *);
    void (*preedit_changed) (GtkIMContext *);
    void (*commit) (GtkIMContext *, const gchar *);
    gboolean(*retrieve_surrounding) (GtkIMContext *);
    gboolean(*delete_surrounding) (GtkIMContext *, gint, gint);
    void (*set_client_window) (GtkIMContext *, GdkWindow *);
    void (*get_preedit_string) (GtkIMContext *, gchar *,
                               PangoAttrList **, gint *);
    gboolean(*filter_keypress) (GtkIMContext *, GdkEventKey *);
    void (*focus_in) (GtkIMContext *);
    void (*focus_out) (GtkIMContext *);
    void (*reset) (GtkIMContext *);
    void (*set_cursor_location) (GtkIMContext *, GdkRectangle *);
    void (*set_use_preedit) (GtkIMContext *, gboolean);
    void (*set_surrounding) (GtkIMContext *, const gchar *, gint,
                             gint);
    gboolean(*get_surrounding) (GtkIMContext *, gchar *, gint *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
} GtkIMContextClass;
typedef struct _GtkCellRendererClass {
    GObjectClass parent_class;
    void (*get_size) (GtkCellRenderer *, GtkWidget *, GdkRectangle
*,
                    gint *, gint *, gint *, gint *);
    void (*render) (GtkCellRenderer *, GdkDrawable *, GtkWidget *,
                    GdkRectangle *, GdkRectangle *, GdkRectangle *,
                    GtkCellRendererState);
    gboolean(*activate) (GtkCellRenderer *, GdkEvent *, GtkWidget
*,
                        const gchar *, GdkRectangle *, GdkRectangle
*,
                        GtkCellRendererState);
    GtkCellEditable *(*start_editing) (GtkCellRenderer *, GdkEvent
*,
                                      GtkWidget *, const gchar *,
                                      GdkRectangle *, GdkRectangle *,
                                      GtkCellRendererState);
    void (*editing_canceled) (GtkCellRenderer *);
    void (*editing_started) (GtkCellRenderer *, GtkCellEditable *,
                             const gchar *);
    void (*_gtk_reserved1) (void);

```

```

        void (*_gtk_reserved2) (void);
    } GtkCellRendererClass;
typedef struct _GtkToolButtonClass {
    GtkToolItemClass parent_class;
    GType button_type;
    void (*clicked) (GtkToolButton *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkToolButtonClass;
typedef struct _GtkMenuToolButtonClass {
    GtkToolButtonClass parent_class;
    void (*show_menu) (GtkMenuToolButton *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkMenuToolButtonClass;
typedef struct _GtkVSeparator {
    GtkSeparator separator;
} GtkVSeparator;
typedef struct _GtkDrawingAreaClass {
    GtkWidgetClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkDrawingAreaClass;
typedef struct _GtkTargetPair {
    GdkAtom target;
    guint flags;
    guint info;
} GtkTargetPair;
typedef struct _GtkCellViewClass {
    GtkWidgetClass parent_class;
} GtkCellViewClass;
typedef struct _GtkCellLayoutIface {
    GTypeInterface g_iface;
    void (*pack_start) (GtkCellLayout *, GtkCellRenderer *,
gboolean);
    void (*pack_end) (GtkCellLayout *, GtkCellRenderer *, gboolean);
    void (*clear) (GtkCellLayout *);
    void (*add_attribute) (GtkCellLayout *, GtkCellRenderer *,
                        const gchar *, gint);
    void (*set_cell_data_func) (GtkCellLayout *, GtkCellRenderer *,
                        GtkCellLayoutDataFunc, gpointer,
                        GDestroyNotify);
    void (*clear_attributes) (GtkCellLayout *, GtkCellRenderer *);
    void (*reorder) (GtkCellLayout *, GtkCellRenderer *, gint);
} GtkCellLayoutIface;
typedef struct _GtkAlignmentPrivate GtkAlignmentPrivate;
typedef struct _GtkFixedClass {
    GtkContainerClass parent_class;
} GtkFixedClass;
typedef struct _GtkHScaleClass {
    GtkScaleClass parent_class;
} GtkHScaleClass;
typedef struct _GtkScrollbarClass {
    GtkRangeClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkScrollbarClass;
typedef struct _GtkAccelLabelClass {

```



```

    GtkLabelClass parent_class;
    gchar *signal_quote1;
    gchar *signal_quote2;
    gchar *mod_name_shift;
    gchar *mod_name_control;
    gchar *mod_name_alt;
    gchar *mod_separator;
    gchar *accel_seperator;
    guint latin1_to_char:1;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkAccelLabelClass;
typedef void (*GtkModuleDisplayInitFunc) (GdkDisplay *);
typedef struct _GtkTextBufferClass {
    GObjectClass parent_class;
    void (*insert_text) (GtkTextBuffer *, GtkTextIter *, const gchar
*,
                        gint);
    void (*insert_pixbuf) (GtkTextBuffer *, GtkTextIter *,
GdkPixbuf *);
    void (*insert_child_anchor) (GtkTextBuffer *, GtkTextIter *,
                                GtkTextChildAnchor *);
    void (*delete_range) (GtkTextBuffer *, GtkTextIter *,
GtkTextIter *);
    void (*changed) (GtkTextBuffer *);
    void (*modified_changed) (GtkTextBuffer *);
    void (*mark_set) (GtkTextBuffer *, const GtkTextIter *,
GtkTextMark *);
    void (*mark_deleted) (GtkTextBuffer *, GtkTextMark *);
    void (*apply_tag) (GtkTextBuffer *, GtkTextTag *, const
GtkTextIter *,
                        const GtkTextIter *);
    void (*remove_tag) (GtkTextBuffer *, GtkTextTag *, const
GtkTextIter *,
                        const GtkTextIter *);
    void (*begin_user_action) (GtkTextBuffer *);
    void (*end_user_action) (GtkTextBuffer *);
    void (*paste_done) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
} GtkTextBufferClass;
typedef struct _GtkVRuler {
    GtkRuler ruler;
} GtkVRuler;
typedef struct _GtkSocketClass {
    GtkContainerClass parent_class;
    void (*plug_added) (GtkSocket *);
    gboolean(*plug_removed) (GtkSocket *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkSocketClass;
typedef struct _GtkIconThemeClass {
    GObjectClass parent_class;
    void (*changed) (GtkIconTheme *);
} GtkIconThemeClass;
typedef struct _GtkVPaned {
    GtkPaned paned;
} GtkVPaned;
typedef struct _GtkTableChild {

```

```

    GtkWidget *widget;
    guint16 left_attach;
    guint16 right_attach;
    guint16 top_attach;
    guint16 bottom_attach;
    guint16 xpadding;
    guint16 ypadding;
    guint xexpand:1;
    guint yexpand:1;
    guint xshrink:1;
    guint yshrink:1;
    guint xfill:1;
    guint yfill:1;
} GtkTableChild;
typedef struct _GtkArrowClass {
    GtkMiscClass parent_class;
} GtkArrowClass;
typedef struct _GtkHScrollbar {
    GtkScrollbar scrollbar;
} GtkHScrollbar;
typedef struct _GtkFileChooserWidgetPrivate
GtkFileChooserWidgetPrivate;
typedef struct _GtkFileChooserWidget {
    GtkVBox parent_instance;
    GtkFileChooserWidgetPrivate *priv;
} GtkFileChooserWidget;
typedef struct _GtkPlugClass {
    GtkWindowClass parent_class;
    void (*embedded) (GtkPlug *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkPlugClass;
typedef struct _GtkCellRendererPixbufClass {
    GtkCellRendererClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCellRendererPixbufClass;
typedef struct _GtkVButtonBoxClass {
    GtkButtonBoxClass parent_class;
} GtkVButtonBoxClass;
typedef struct _GtkTextChildAnchorClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTextChildAnchorClass;
typedef struct _GtkCellRendererCombo {
    GtkCellRendererText parent;
    GtkTreeModel *model;
    gint text_column;
    gboolean has_entry;
    guint focus_out_id;
} GtkCellRendererCombo;
typedef struct _GtkTreeModelIface {
    GTypeInterface g_iface;
    void (*row_changed) (GtkTreeModel *, GtkTreePath *, GtkTreeIter
*);
    void (*row_inserted) (GtkTreeModel *, GtkTreePath *,
GtkTreeIter *);
    void (*row_has_child_toggled) (GtkTreeModel *, GtkTreePath *,
GtkTreeIter *);

```

```

void (*row_deleted) (GtkTreeModel *, GtkTreePath *);
void (*rows_reordered) (GtkTreeModel *, GtkTreePath *,
GtkTreeIter *,
gint *);
GtkTreeModelFlags (*get_flags) (GtkTreeModel *);
gint (*get_n_columns) (GtkTreeModel *);
GType (*get_column_type) (GtkTreeModel *, gint);
gboolean (*get_iter) (GtkTreeModel *, GtkTreeIter *,
GtkTreePath *);
GtkTreePath (*get_path) (GtkTreeModel *, GtkTreeIter *);
void (*get_value) (GtkTreeModel *, GtkTreeIter *, gint, GValue
*);
gboolean (*iter_next) (GtkTreeModel *, GtkTreeIter *);
gboolean (*iter_children) (GtkTreeModel *, GtkTreeIter *,
GtkTreeIter *);
gboolean (*iter_has_child) (GtkTreeModel *, GtkTreeIter *);
gint (*iter_n_children) (GtkTreeModel *, GtkTreeIter *);
gboolean (*iter_nth_child) (GtkTreeModel *, GtkTreeIter *,
GtkTreeIter *, gint);
gboolean (*iter_parent) (GtkTreeModel *, GtkTreeIter *,
GtkTreeIter *);
void (*ref_node) (GtkTreeModel *, GtkTreeIter *);
void (*unref_node) (GtkTreeModel *, GtkTreeIter *);
} GtkTreeModelIface;
typedef struct _GtkAccessibleClass {
AtkObjectClass parent_class;
void (*connect_widget_destroyed) (GtkAccessible *);
void (*_gtk_reserved1) (void);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
} GtkAccessibleClass;
typedef struct _GtkCellRendererTextClass {
GtkCellRendererClass parent_class;
void (*edited) (GtkCellRendererText *, const gchar *, const
gchar *);
void (*_gtk_reserved1) (void);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
} GtkCellRendererTextClass;
typedef struct _GtkVScaleClass {
GtkScaleClass parent_class;
} GtkVScaleClass;
typedef struct _GtkCurveClass {
GtkDrawingAreaClass parent_class;
void (*curve_type_changed) (GtkCurve *);
void (*_gtk_reserved1) (void);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
} GtkCurveClass;
typedef struct _GtkSeparatorToolItemClass {
GtkToolItemClass parent_class;
void (*_gtk_reserved1) (void);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
} GtkSeparatorToolItemClass;
typedef struct _GtkStyleClass {
GObjectClass parent_class;
void (*realize) (GtkStyle *);
void (*unrealize) (GtkStyle *);
void (*copy) (GtkStyle *, GtkStyle *);
GtkStyle *(*clone) (GtkStyle *);
void (*init_from_rc) (GtkStyle *, GtkRcStyle *);

```

```

void (*set_background) (GtkStyle *, GdkWindow *, GtkStateType);
GdkPixbuf *(*render_icon) (GtkStyle *, const GtkIconSource *,
                             GtkTextDirection,
                             GtkStateType,
                             GtkIconSize,
                             GtkWidget *, const gchar *);
void (*draw_hline) (GtkStyle *, GdkWindow *, GtkStateType,
                    GdkRectangle *, GtkWidget *, const gchar *,
                    gint,
                    gint, gint);
void (*draw_vline) (GtkStyle *, GdkWindow *, GtkStateType,
                    GdkRectangle *, GtkWidget *, const gchar *,
                    gint,
                    gint, gint);
void (*draw_shadow) (GtkStyle *, GdkWindow *, GtkStateType,
                    GtkShadowType, GdkRectangle *, GtkWidget *,
                    const gchar *, gint, gint, gint, gint);
void (*draw_polygon) (GtkStyle *, GdkWindow *, GtkStateType,
                    GtkShadowType, GdkRectangle *, GtkWidget *,
                    const gchar *, GdkPoint *, gint, gboolean);
void (*draw_arrow) (GtkStyle *, GdkWindow *, GtkStateType,
                    GtkShadowType, GdkRectangle *, GtkWidget *,
                    const gchar *, GtkArrowType, gboolean, gint,
                    gint,
                    gint, gint);
void (*draw_diamond) (GtkStyle *, GdkWindow *, GtkStateType,
                    GtkShadowType, GdkRectangle *, GtkWidget *,
                    const gchar *, gint, gint, gint, gint);
void (*draw_string) (GtkStyle *, GdkWindow *, GtkStateType,
                    GdkRectangle *, GtkWidget *, const gchar *,
                    gint,
                    gint, const gchar *);
void (*draw_box) (GtkStyle *, GdkWindow *, GtkStateType,
                  GtkShadowType,
                  GdkRectangle *, GtkWidget *, const gchar *, gint,
                  gint, gint, gint);
void (*draw_flat_box) (GtkStyle *, GdkWindow *, GtkStateType,
                      GtkShadowType, GdkRectangle *, GtkWidget *,
                      const gchar *, gint, gint, gint, gint);
void (*draw_check) (GtkStyle *, GdkWindow *, GtkStateType,
                   GtkShadowType, GdkRectangle *, GtkWidget *,
                   const gchar *, gint, gint, gint, gint);
void (*draw_option) (GtkStyle *, GdkWindow *, GtkStateType,
                    GtkShadowType, GdkRectangle *, GtkWidget *,
                    const gchar *, gint, gint, gint, gint);
void (*draw_tab) (GtkStyle *, GdkWindow *, GtkStateType,
                 GtkShadowType,
                 GdkRectangle *, GtkWidget *, const gchar *, gint,
                 gint, gint, gint);
void (*draw_shadow_gap) (GtkStyle *, GdkWindow *, GtkStateType,
                        GtkShadowType, GdkRectangle *, GtkWidget *,
                        const gchar *, gint, gint, gint, gint,
                        GtkPositionType, gint, gint);
void (*draw_box_gap) (GtkStyle *, GdkWindow *, GtkStateType,
                     GtkShadowType, GdkRectangle *, GtkWidget *,
                     const gchar *, gint, gint, gint, gint,
                     GtkPositionType, gint, gint);
void (*draw_extension) (GtkStyle *, GdkWindow *, GtkStateType,
                       GtkShadowType, GdkRectangle *, GtkWidget *,
                       const gchar *, gint, gint, gint, gint,
                       GtkPositionType);
void (*draw_focus) (GtkStyle *, GdkWindow *, GtkStateType,
                   GdkRectangle *, GtkWidget *, const gchar *,
                   gint,
                   gint, gint, gint);
void (*draw_slider) (GtkStyle *, GdkWindow *, GtkStateType,
                    GtkShadowType, GdkRectangle *, GtkWidget *,

```

```

        const gchar *, gint, gint, gint, gint,
        GtkOrientation);
void (*draw_handle) (GtkStyle *, GdkWindow *, GtkStateType,
        GtkShadowType, GdkRectangle *, GtkWidget *,
        const gchar *, gint, gint, gint, gint,
        GtkOrientation);
void (*draw_expander) (GtkStyle *, GdkWindow *, GtkStateType,
        GdkRectangle *, GtkWidget *, const gchar *,
        gint, gint, GtkExpanderStyle);
void (*draw_layout) (GtkStyle *, GdkWindow *, GtkStateType,
gboolean,
        GdkRectangle *, GtkWidget *, const gchar *,
gint,
        gint, PangoLayout *);
void (*draw_resize_grip) (GtkStyle *, GdkWindow *, GtkStateType,
        GdkRectangle *, GtkWidget *, const gchar
*,
        GdkWindowEdge, gint, gint, gint, gint);
void (*get_text_area_size) (void);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
void (*_gtk_reserved5) (void);
void (*_gtk_reserved6) (void);
void (*_gtk_reserved7) (void);
void (*_gtk_reserved8) (void);
void (*_gtk_reserved9) (void);
void (*_gtk_reserved10) (void);
void (*_gtk_reserved11) (void);
void (*_gtk_reserved12) (void);
} GtkStyleClass;
typedef struct _GtkSeparatorMenuItemClass {
    GtkMenuItemClass parent_class;
} GtkSeparatorMenuItemClass;
typedef struct _GtkIMMulticontextClass {
    GtkIMContextClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkIMMulticontextClass;
typedef struct _GtkToggleToolButtonClass {
    GtkToolButtonClass parent_class;
    void (*toggled) (GtkToggleToolButton *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkToggleToolButtonClass;
typedef struct _GtkRadioToolButtonClass {
    GtkToggleToolButtonClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkRadioToolButtonClass;
typedef struct _GtkCellRendererProgressClass {
    GtkCellRendererClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCellRendererProgressClass;
typedef struct _GtkVScrollbar {
    GtkScrollbar scrollbar;
} GtkVScrollbar;

```

```

typedef struct _GtkVPanedClass {
    GtkPanedClass parent_class;
} GtkVPanedClass;
typedef struct _GtkToggleButtonClass {
    GtkButtonClass parent_class;
    void (*toggled) (GtkToggleButton *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkToggleButtonClass;
typedef struct _GtkCellRendererToggleClass {
    GtkCellRendererClass parent_class;
    void (*toggled) (GtkCellRendererToggle *, const gchar *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCellRendererToggleClass;
typedef struct _GtkVButtonBox {
    GtkButtonBox button_box;
} GtkVButtonBox;
typedef struct _GtkTreeDragDestIface {
    GTypeInterface g_iface;
    gboolean(*drag_data_received) (GtkTreeDragDest *, GtkTreePath
*,
                                GtkSelectionData *);
    gboolean(*row_drop_possible) (GtkTreeDragDest *, GtkTreePath *,
                                GtkSelectionData *);
} GtkTreeDragDestIface;
typedef struct _GtkColorSelectionClass {
    GtkVBoxClass parent_class;
    void (*color_changed) (GtkColorSelection *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkColorSelectionClass;
typedef struct _GtkCellRendererProgressPrivate {
    GtkCellRendererProgressPrivate;
} GtkCellRendererProgressPrivate;
typedef struct _GtkCellRendererProgress {
    GtkCellRenderer parent_instance;
    GtkCellRendererProgressPrivate *priv;
} GtkCellRendererProgress;
typedef struct _GtkImageClass {
    GtkMiscClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkImageClass;
typedef struct _GtkMessageDialogClass {
    GtkDialogClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkMessageDialogClass;
typedef struct _GtkTreeSortableIface {
    GTypeInterface g_iface;
    void (*sort_column_changed) (GtkTreeSortable *);
    gboolean(*get_sort_column_id) (GtkTreeSortable *, gint *,
                                GtkSortType *);
    void (*set_sort_column_id) (GtkTreeSortable *, gint,
                                GtkSortType);
}

```

```

    void (*set_sort_func) (GtkTreeSortable *, gint,
GtkTreeIterCompareFunc,
                           gpointer, GtkDestroyNotify);
    void (*set_default_sort_func) (GtkTreeSortable *,
                                   GtkTreeIterCompareFunc, gpointer,
                                   GtkDestroyNotify);
    gboolean(*has_default_sort_func) (GtkTreeSortable *);
} GtkTreeSortableIface;
typedef struct _GtkCalendarClass {
    GtkWidgetClass parent_class;
    void (*month_changed) (GtkCalendar *);
    void (*day_selected) (GtkCalendar *);
    void (*day_selected_double_click) (GtkCalendar *);
    void (*prev_month) (GtkCalendar *);
    void (*next_month) (GtkCalendar *);
    void (*prev_year) (GtkCalendar *);
    void (*next_year) (GtkCalendar *);
} GtkCalendarClass;
typedef struct _GtkEntryClass {
    GtkWidgetClass parent_class;
    void (*populate_popup) (GtkEntry *, GtkMenu *);
    void (*activate) (GtkEntry *);
    void (*move_cursor) (GtkEntry *, GtkMovementStep, gint,
gboolean);
    void (*insert_at_cursor) (GtkEntry *, const gchar *);
    void (*delete_from_cursor) (GtkEntry *, GtkDeleteType, gint);
    void (*backspace) (GtkEntry *);
    void (*cut_clipboard) (GtkEntry *);
    void (*copy_clipboard) (GtkEntry *);
    void (*paste_clipboard) (GtkEntry *);
    void (*toggle_overwrite) (GtkEntry *);
    void (*get_text_area_size) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
} GtkEntryClass;
typedef struct _GtkSpinButtonClass {
    GtkEntryClass parent_class;
    gint(*input) (GtkSpinButton *, gdouble *);
    gint(*output) (GtkSpinButton *);
    void (*value_changed) (GtkSpinButton *);
    void (*change_value) (GtkSpinButton *, GtkScrollType);
    void (*cursor_on_match) (GtkSpinButton *);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkSpinButtonClass;
typedef struct _GtkInputDialog {
    GtkDialog dialog;
    GtkWidget *axis_list;
    GtkWidget *axis_listbox;
    GtkWidget *mode_optionmenu;
    GtkWidget *close_button;
    GtkWidget *save_button;
    GtkWidget *axis_items[7];
    GdkDevice *current_device;
    GtkWidget *keys_list;
    GtkWidget *keys_listbox;
} GtkInputDialog;
typedef struct _GtkInputDialogClass {
    GtkDialogClass parent_class;
    void (*enable_device) (GtkInputDialog *, GdkDevice *);
    void (*disable_device) (GtkInputDialog *, GdkDevice *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
}

```

```

} GtkInputDialogClass;
typedef struct _GtkEntryCompletionClass {
    GObjectClass parent_class;
    gboolean(*match_selected) (GtkEntryCompletion *, GtkTreeModel
*,
                                GtkTreeIter *);
    void (*action_activated) (GtkEntryCompletion *, gint);
    gboolean(*insert_prefix) (GtkEntryCompletion *, const gchar *);
    gboolean(*cursor_on_match) (GtkEntryCompletion *, GtkTreeModel
*,
                                GtkTreeIter *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
} GtkEntryCompletionClass;
typedef struct _GtkAlignmentClass {
    GtkBinClass parent_class;
} GtkAlignmentClass;
typedef struct _GtkTreeModelFilterClass {
    GObjectClass parent_class;
    void (*_gtk_reserved0) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
} GtkTreeModelFilterClass;
typedef struct _GtkRcStyleClass {
    GObjectClass parent_class;
    GtkRcStyle *(*create_rc_style) (GtkRcStyle *);
    guint(*parse) (GtkRcStyle *, GtkSettings *, GScanner *);
    void (*merge) (GtkRcStyle *, GtkRcStyle *);
    GtkStyle *(*create_style) (GtkRcStyle *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkRcStyleClass;
typedef struct _GtkUIManagerClass {
    GObjectClass parent_class;
    void (*add_widget) (GtkUIManager *, GtkWidget *);
    void (*actions_changed) (GtkUIManager *);
    void (*connect_proxy) (GtkUIManager *, GtkAction *, GtkWidget
*);
    void (*disconnect_proxy) (GtkUIManager *, GtkAction *,
GtkWidget *);
    void (*pre_activate) (GtkUIManager *, GtkAction *);
    void (*post_activate) (GtkUIManager *, GtkAction *);
    GtkWidget *(*get_widget) (GtkUIManager *, const gchar *);
    GtkAction *(*get_action) (GtkUIManager *, const gchar *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
} GtkUIManagerClass;
typedef struct _GtkSeparatorClass {
    GtkWidgetClass parent_class;
} GtkSeparatorClass;
typedef struct _GtkVSeparatorClass {
    GtkSeparatorClass parent_class;
} GtkVSeparatorClass;
typedef struct _GtkCheckButtonClass {
    GtkToggleButtonClass parent_class;
    void (*draw_indicator) (GtkCheckButton *, GdkRectangle *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCheckButtonClass;
typedef struct _GtkRadioMenuItemClass {
    GtkCheckMenuItemClass parent_class;

```



```

void (*group_changed) (GtkRadioMenuItem *);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
} GtkRadioMenuItemClass;
typedef struct _GtkTextViewClass {
    GtkContainerClass parent_class;
    void (*set_scroll_adjustments) (GtkTextView *, GtkAdjustment *,
                                    GtkAdjustment *);
    void (*populate_popup) (GtkTextView *, GtkMenu *);
    void (*move_cursor) (GtkTextView *, GtkMovementStep, gint,
gboolean);
    void (*page_horizontally) (GtkTextView *, gint, gboolean);
    void (*set_anchor) (GtkTextView *);
    void (*insert_at_cursor) (GtkTextView *, const gchar *);
    void (*delete_from_cursor) (GtkTextView *, GtkDeleteType, gint);
    void (*backspace) (GtkTextView *);
    void (*cut_clipboard) (GtkTextView *);
    void (*copy_clipboard) (GtkTextView *);
    void (*paste_clipboard) (GtkTextView *);
    void (*toggle_overwrite) (GtkTextView *);
    void (*move_focus) (GtkTextView *, GtkDirectionType);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
    void (*_gtk_reserved7) (void);
} GtkTextViewClass;
typedef void (*GtkColorSelectionChangePaletteFunc) (const GdkColor
*,
                                                    gint);

typedef struct _GtkTreeModelSortClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTreeModelSortClass;
typedef struct _GtkCellEditableIface {
    GTypeInterface g_iface;
    void (*editing_done) (GtkCellEditable *);
    void (*remove_widget) (GtkCellEditable *);
    void (*start_editing) (GtkCellEditable *, GdkEvent *);
} GtkCellEditableIface;
typedef struct _GtkToolbarClass {
    GtkContainerClass parent_class;
    void (*orientation_changed) (GtkToolbar *, GtkOrientation);
    void (*style_changed) (GtkToolbar *, GtkToolbarStyle);
    gboolean(*popup_context_menu) (GtkToolbar *, gint, gint, gint);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
} GtkToolbarClass;
typedef struct _GtkHandleBoxClass {
    GtkBinClass parent_class;
    void (*child_attached) (GtkHandleBox *, GtkWidget *);
    void (*child_detached) (GtkHandleBox *, GtkWidget *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkHandleBoxClass;
typedef struct _GtkHSeparator {
    GtkSeparator separator;

```

```

} GtkHSeparator;
typedef struct _GtkTextTagTableClass {
    GObjectClass parent_class;
    void (*tag_changed) (GtkTextTagTable *, GtkTextTag *, gboolean);
    void (*tag_added) (GtkTextTagTable *, GtkTextTag *);
    void (*tag_removed) (GtkTextTagTable *, GtkTextTag *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTextTagTableClass;
typedef struct _GtkComboBoxClass {
    GtkBinClass parent_class;
    void (*changed) (GtkComboBox *);
    gchar *(*get_active_text) (GtkComboBox *);
    void (*_gtk_reserved0) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
} GtkComboBoxClass;
typedef struct _GtkComboBoxEntryClass {
    GtkComboBoxClass parent_class;
    void (*_gtk_reserved0) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
} GtkComboBoxEntryClass;
typedef struct _GtkFontSelectionClass {
    GtkVBoxClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkFontSelectionClass;
typedef struct _GtkFrameClass {
    GtkBinClass parent_class;
    void (*compute_child_allocation) (GtkFrame *, GtkAllocation *);
} GtkFrameClass;
typedef struct _GtkHScrollbarClass {
    GtkScrollbarClass parent_class;
} GtkHScrollbarClass;
typedef struct _GtkFileSelectionClass {
    GtkDialogClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkFileSelectionClass;
typedef struct _GtkScrolledWindowClass {
    GtkBinClass parent_class;
    gint scrollbar_spacing;
    void (*scroll_child) (GtkScrolledWindow *, GtkScrollType,
        gboolean);
    void (*move_focus_out) (GtkScrolledWindow *, GtkDirectionType);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkScrolledWindowClass;
typedef struct _GtkTextMarkClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTextMarkClass;
typedef struct _GtkToolbarPrivate GtkToolbarPrivate;

```

```

typedef struct _GtkVScale {
    GtkScale scale;
} GtkVScale;
typedef struct _GtkFixedChild {
    GtkWidget *widget;
    gint x;
    gint y;
} GtkFixedChild;
typedef struct _GtkLayoutClass {
    GtkContainerClass parent_class;
    void (*set_scroll_adjustments) (GtkLayout *, GtkAdjustment *,
                                    GtkAdjustment *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkLayoutClass;
typedef struct _GtkEventBoxClass {
    GtkBinClass parent_class;
} GtkEventBoxClass;
typedef struct _GtkIMContextSimpleClass {
    GtkIMContextClass parent_class;
} GtkIMContextSimpleClass;
typedef struct _GtkExpanderClass {
    GtkBinClass parent_class;
    void (*activate) (GtkExpander *);
} GtkExpanderClass;
typedef struct _GtkProgressClass {
    GtkWidgetClass parent_class;
    void (*paint) (GtkProgress *);
    void (*update) (GtkProgress *);
    void (*act_mode_enter) (GtkProgress *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkProgressClass;
typedef void (*GtkModuleInitFunc) (gint *, gchar * **);
typedef struct _GtkFileChooserDialogClass {
    GtkDialogClass parent_class;
} GtkFileChooserDialogClass;
typedef struct _GtkVScrollbarClass {
    GtkScrollbarClass parent_class;
} GtkVScrollbarClass;
typedef struct _GtkTableClass {
    GtkContainerClass parent_class;
} GtkTableClass;
typedef struct _GtkFontButtonClass {
    GtkButtonClass parent_class;
    void (*font_set) (GtkFontButton *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkFontButtonClass;
typedef struct _GtkHSeparatorClass {
    GtkSeparatorClass parent_class;
} GtkHSeparatorClass;
typedef struct _GtkColorSelectionDialogClass {
    GtkDialogClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkColorSelectionDialogClass;
typedef struct _GtkColorButtonClass {

```

```

        GtkButtonClass parent_class;
        void (*color_set) (GtkColorButton *);
        void (*_gtk_reserved1) (void);
        void (*_gtk_reserved2) (void);
        void (*_gtk_reserved3) (void);
        void (*_gtk_reserved4) (void);
    } GtkColorButtonClass;
typedef struct _GtkMenuBar {
    GtkMenuShell menu_shell;
} GtkMenuBar;
typedef struct _GtkMenuBarClass {
    GtkMenuShellClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkMenuBarClass;
typedef struct _GtkColorSelectionDialog {
    GtkDialog parent_instance;
    GtkWidget *colorsel;
    GtkWidget *ok_button;
    GtkWidget *cancel_button;
    GtkWidget *help_button;
} GtkColorSelectionDialog;
typedef struct _GtkStatusbarClass {
    GtkHBoxClass parent_class;
    gpointer reserved;
    void (*text_pushed) (GtkStatusbar *, guint, const gchar *);
    void (*text_popped) (GtkStatusbar *, guint, const gchar *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkStatusbarClass;
typedef struct _GtkSeparatorMenuItem {
    GtkMenuItem menu_item;
} GtkSeparatorMenuItem;
typedef struct _GtkAboutDialogClass {
    GtkDialogClass parent_class;
    gboolean(*activate_link) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkAboutDialogClass;
typedef struct _GtkListStoreClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkListStoreClass;
typedef struct _GtkFileChooserButtonClass {
    GtkHBoxClass parent_class;
    void (*file_set) (GtkFileChooserButton * fc);
    void *_gtk_reserved2;
    void *_gtk_reserved3;
    void *_gtk_reserved4;
    void *_gtk_reserved5;
    void *_gtk_reserved6;
    void *_gtk_reserved7;
    void *_gtk_reserved8;
} GtkFileChooserButtonClass;
typedef enum {
    GTK_TOOLBAR_SPACE_EMPTY = 0,
    GTK_TOOLBAR_SPACE_LINE = 1
} GtkToolbarSpaceStyle;

```

```

typedef struct _GtkMenuClass {
    GtkMenuShellClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkMenuClass;
typedef struct _GtkHScale {
    GtkScale scale;
} GtkHScale;
typedef struct _GtkGammaCurveClass {
    GtkVBoxClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkGammaCurveClass;
typedef struct _GtkCellRendererComboClass {
    GtkCellRendererTextClass parent;
} GtkCellRendererComboClass;
typedef struct _GtkTearoffMenuItem {
    GtkMenuItem menu_item;
    guint torn_off:1;
} GtkTearoffMenuItem;
typedef struct _GtkNotebookClass {
    GtkContainerClass parent_class;
    void (*switch_page) (GtkNotebook *, GtkNotebookPage *, guint);
    gboolean(*select_page) (GtkNotebook *, gboolean);
    gboolean(*focus_tab) (GtkNotebook *, GtkNotebookTab);
    void (*change_current_page) (GtkNotebook *, gint);
    void (*move_focus_out) (GtkNotebook *, GtkDirectionType);
    gboolean(*reorder_tab) (void);
    gint(*insert_page) (void);
    GtkNotebook *(*create_window) (void);
    void (*_gtk_reserved4) (void);
} GtkNotebookClass;
typedef struct _GtkImageMenuItemClass {
    GtkMenuItemClass parent_class;
} GtkImageMenuItemClass;
typedef struct _GtkTooltipsClass {
    GObjectClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTooltipsClass;
typedef struct _GtkTreeViewClass {
    GtkContainerClass parent_class;
    void (*set_scroll_adjustments) (GtkTreeView *, GtkAdjustment *,
                                    GtkAdjustment *);
    void (*row_activated) (GtkTreeView *, GtkTreePath *,
                           GtkTreeViewColumn *);
    gboolean(*test_expand_row) (GtkTreeView *, GtkTreeIter *,
                                GtkTreePath *);
    gboolean(*test_collapse_row) (GtkTreeView *, GtkTreeIter *,
                                  GtkTreePath *);
    void (*row_expanded) (GtkTreeView *, GtkTreeIter *, GtkTreePath
                          *);
    void (*row_collapsed) (GtkTreeView *, GtkTreeIter *,
                           GtkTreePath *);
    void (*columns_changed) (GtkTreeView *);
    void (*cursor_changed) (GtkTreeView *);
    gboolean(*move_cursor) (GtkTreeView *, GtkMovementStep, gint);
    gboolean(*select_all) (GtkTreeView *);
    gboolean(*unselect_all) (GtkTreeView *);
    gboolean(*select_cursor_row) (GtkTreeView *, gboolean);

```

```

gboolean(*toggle_cursor_row) (GtkTreeView *);
gboolean(*expand_collapse_cursor_row) (GtkTreeView *, gboolean,
                                       gboolean, gboolean);
gboolean(*select_cursor_parent) (GtkTreeView *);
gboolean(*start_interactive_search) (GtkTreeView *);
void (*_gtk_reserved0) (void);
void (*_gtk_reserved1) (void);
void (*_gtk_reserved2) (void);
void (*_gtk_reserved3) (void);
void (*_gtk_reserved4) (void);
} GtkTreeViewClass;
typedef struct _GtkFileChooserWidgetClass {
    GtkVBoxClass parent_class;
} GtkFileChooserWidgetClass;
typedef struct _GtkTearoffMenuItemClass {
    GtkMenuItemClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTearoffMenuItemClass;
typedef struct _GtkRadioButtonClass {
    GtkCheckButtonClass parent_class;
    void (*group_changed) (GtkRadioButton *);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkRadioButtonClass;
typedef struct _GtkAdjustmentClass {
    GObjectClass parent_class;
    void (*changed) (GtkAdjustment *);
    void (*value_changed) (GtkAdjustment *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkAdjustmentClass;
typedef struct _GtkHPaned {
    GtkPaned paned;
} GtkHPaned;
typedef struct _GtkRadioActionClass {
    GtkToggleActionClass parent_class;
    void (*changed) (GtkRadioAction *, GtkRadioAction *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkRadioActionClass;
typedef struct _GtkTextTagClass {
    GObjectClass parent_class;
    gboolean(*event) (GtkTextTag *, GObject *, GdkEvent *,
                    const GtkTextIter *);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkTextTagClass;
typedef struct _GtkIconViewClass {
    GtkContainerClass parent_class;
    void (*set_scroll_adjustments) (GtkIconView *, GtkAdjustment *,
                                   GtkAdjustment *);
    void (*item_activated) (GtkIconView *, GtkTreePath *);
    void (*selection_changed) (GtkIconView *);
    void (*select_all) (GtkIconView *);
    void (*unselect_all) (GtkIconView *);
    void (*select_cursor_item) (GtkIconView *);

```

```

void (*toggle_cursor_item) (GtkIconView *);
gboolean(*move_cursor) (GtkIconView *, GtkMovementStep, gint);
gboolean(*activate_cursor_item) (GtkIconView *);
} GtkIconViewClass;
typedef struct _GtkProgressBarClass {
    GtkProgressClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkProgressBarClass;
typedef struct _GtkAspectFrameClass {
    GtkFrameClass parent_class;
} GtkAspectFrameClass;
typedef struct _GtkGammaCurve {
    GtkVBox vbox;
    GtkWidget *table;
    GtkWidget *curve;
    GtkWidget *button[5];
    gfloat gamma;
    GtkWidget *gamma_dialog;
    GtkWidget *gamma_text;
} GtkGammaCurve;
typedef gboolean(*GtkAccelGroupActivate) (GtkAccelGroup *, GObject
*,
                                         guint, GdkModifierType);

enum {
    GTK_TREE_SORTABLE_DEFAULT_SORT_COLUMN_ID = -1,
    GTK_TREE_SORTABLE_UNSORTED_SORT_COLUMN_ID = -2
};
typedef enum {
    GTK_DEBUG_MISC = 1 << 0,
    GTK_DEBUG_PLUGSOCKET = 1 << 1,
    GTK_DEBUG_TEXT = 1 << 2,
    GTK_DEBUG_TREE = 1 << 3,
    GTK_DEBUG_UPDATES = 1 << 4,
    GTK_DEBUG_KEYBINDINGS = 1 << 5,
    GTK_DEBUG_MULTIHEAD = 1 << 6,
    GTK_DEBUG_MODULES = 1 << 7,
    GTK_DEBUG_GEOMETRY = 1 << 8,
    GTK_DEBUG_ICONTHEME = 1 << 9
} GtkDebugFlag;
typedef enum {
    GTK_CELL_RENDERER_MODE_INERT,
    GTK_CELL_RENDERER_MODE_ACTIVATABLE,
    GTK_CELL_RENDERER_MODE_EDITABLE
} GtkCellRendererMode;
typedef enum {
    GTK_ICON_THEME_NOT_FOUND,
    GTK_ICON_THEME_FAILED
} GtkIconThemeError;
typedef enum {
    GTK_RESPONSE_NONE = -1,
    GTK_RESPONSE_REJECT = -2,
    GTK_RESPONSE_ACCEPT = -3,
    GTK_RESPONSE_DELETE_EVENT = -4,
    GTK_RESPONSE_OK = -5,
    GTK_RESPONSE_CANCEL = -6,
    GTK_RESPONSE_CLOSE = -7,
    GTK_RESPONSE_YES = -8,
    GTK_RESPONSE_NO = -9,
    GTK_RESPONSE_APPLY = -10,
    GTK_RESPONSE_HELP = -11
} GtkResponseType;
typedef enum {
    GTK_TARGET_SAME_APP = 1 << 0,

```

```

    GTK_TARGET_SAME_WIDGET = 1 << 1
} GtkTargetFlags;
typedef enum {
    GTK_FILE_CHOOSER_ERROR_NONEXISTENT,
    GTK_FILE_CHOOSER_ERROR_BAD_FILENAME
} GtkFileChooserError;
typedef enum {
    GTK_IN_DESTRUCTION = 1 << 0,
    GTK_FLOATING = 1 << 1,
    GTK_RESERVED_1 = 1 << 2,
    GTK_RESERVED_2 = 1 << 3
} GtkObjectFlags;
typedef enum {
    GTK_RC_TOKEN_INVALID = G_TOKEN_LAST,
    GTK_RC_TOKEN_INCLUDE,
    GTK_RC_TOKEN_NORMAL,
    GTK_RC_TOKEN_ACTIVE,
    GTK_RC_TOKEN_PRELIGHT,
    GTK_RC_TOKEN_SELECTED,
    GTK_RC_TOKEN_INSENSITIVE,
    GTK_RC_TOKEN_FG,
    GTK_RC_TOKEN_BG,
    GTK_RC_TOKEN_TEXT,
    GTK_RC_TOKEN_BASE,
    GTK_RC_TOKEN_XTHICKNESS,
    GTK_RC_TOKEN_YTHICKNESS,
    GTK_RC_TOKEN_FONT,
    GTK_RC_TOKEN_FONTSET,
    GTK_RC_TOKEN_FONT_NAME,
    GTK_RC_TOKEN_BG_PIXMAP,
    GTK_RC_TOKEN_PIXMAP_PATH,
    GTK_RC_TOKEN_STYLE,
    GTK_RC_TOKEN_BINDING,
    GTK_RC_TOKEN_BIND,
    GTK_RC_TOKEN_WIDGET,
    GTK_RC_TOKEN_WIDGET_CLASS,
    GTK_RC_TOKEN_CLASS,
    GTK_RC_TOKEN_LOWEST,
    GTK_RC_TOKEN_GTK,
    GTK_RC_TOKEN_APPLICATION,
    GTK_RC_TOKEN_THEME,
    GTK_RC_TOKEN_RC,
    GTK_RC_TOKEN_HIGHEST,
    GTK_RC_TOKEN_ENGINE,
    GTK_RC_TOKEN_MODULE_PATH,
    GTK_RC_TOKEN_IM_MODULE_PATH,
    GTK_RC_TOKEN_IM_MODULE_FILE,
    GTK_RC_TOKEN_STOCK,
    GTK_RC_TOKEN_LTR,
    GTK_RC_TOKEN_RTL,
    GTK_RC_TOKEN_LAST
} GtkRcTokenType;
typedef enum {
    GTK_TOPLEVEL = 1 << 4,
    GTK_NO_WINDOW = 1 << 5,
    GTK_REALIZED = 1 << 6,
    GTK_MAPPED = 1 << 7,
    GTK_VISIBLE = 1 << 8,
    GTK_SENSITIVE = 1 << 9,
    GTK_PARENT_SENSITIVE = 1 << 10,
    GTK_CAN_FOCUS = 1 << 11,
    GTK_HAS_FOCUS = 1 << 12,
    GTK_CAN_DEFAULT = 1 << 13,
    GTK_HAS_DEFAULT = 1 << 14,
    GTK_HAS_GRAB = 1 << 15,
    GTK_RC_STYLE = 1 << 16,

```



```

    GTK_COMPOSITE_CHILD = 1 << 17,
    GTK_NO_REPARENT = 1 << 18,
    GTK_APP_PAINTABLE = 1 << 19,
    GTK_RECEIVES_DEFAULT = 1 << 20,
    GTK_DOUBLE_BUFFERED = 1 << 21,
    GTK_NO_SHOW_ALL = 1 << 22
} GtkWidgetFlags;
typedef enum {
    GTK_ICON_VIEW_NO_DROP,
    GTK_ICON_VIEW_DROP_INTRO,
    GTK_ICON_VIEW_DROP_LEFT,
    GTK_ICON_VIEW_DROP_RIGHT,
    GTK_ICON_VIEW_DROP_ABOVE,
    GTK_ICON_VIEW_DROP_BELOW
} GtkIconViewDropPosition;
typedef struct _GtkPageSetup GtkPageSetup;
typedef struct _GtkPrintContext GtkPrintContext;
typedef struct _GtkPaperSize GtkPaperSize;
typedef struct _GtkPrintOperationClass {
    GObjectClass parent_class;
    void (*done) (void);
    void (*begin_print) (void);
    gboolean (*paginate) (void);
    void (*request_page_setup) (void);
    void (*draw_page) (void);
    void (*end_print) (void);
    void (*status_changed) (void);
    GtkWidget *(*create_custom_widget) (void);
    void (*custom_widget_apply) (void);
    gboolean (*preview) (void);
    void (*update_custom_widget) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
    void (*_gtk_reserved7) (void);
} GtkPrintOperationClass;
typedef struct _GtkPrintOperationPrivate GtkPrintOperationPrivate;
typedef struct _GtkPrintOperation {
    GObject parent_instance;
    GtkPrintOperationPrivate *priv;
} GtkPrintOperation;
typedef enum {
    GTK_PRINT_STATUS_INITIAL,
    GTK_PRINT_STATUS_PREPARING,
    GTK_PRINT_STATUS_GENERATING_DATA,
    GTK_PRINT_STATUS_SENDING_DATA,
    GTK_PRINT_STATUS_PENDING,
    GTK_PRINT_STATUS_PENDING_ISSUE,
    GTK_PRINT_STATUS_PRINTING,
    GTK_PRINT_STATUS_FINISHED,
    GTK_PRINT_STATUS_FINISHED_ABORTED
} GtkPrintStatus;
typedef enum {
    GTK_PRINT_OPERATION_RESULT_ERROR,
    GTK_PRINT_OPERATION_RESULT_APPLY,
    GTK_PRINT_OPERATION_RESULT_CANCEL,
    GTK_PRINT_OPERATION_RESULT_IN_PROGRESS
} GtkPrintOperationResult;
typedef enum {
    GTK_PRINT_OPERATION_ACTION_PRINT_DIALOG,
    GTK_PRINT_OPERATION_ACTION_PRINT,
    GTK_PRINT_OPERATION_ACTION_PREVIEW,
    GTK_PRINT_OPERATION_ACTION_EXPORT
} GtkPrintOperationAction;

```

```

typedef enum {
    GTK_PRINT_ERROR_GENERAL,
    GTK_PRINT_ERROR_INTERNAL_ERROR,
    GTK_PRINT_ERROR_NOMEM,
    GTK_PRINT_ERROR_INVALID_FILE
} GtkPrintError;
typedef void (*GtkPageSetupDoneFunc) (GtkPageSetup *, gpointer);
typedef struct _GtkPrintSettings GtkPrintSettings;
typedef void (*GtkPrintSettingsFunc) (const char *, const char *,
                                      gpointer);

typedef struct _GtkPageRange {
    gint start;
    gint end;
} GtkPageRange;
typedef struct _GtkLinkButton {
    GtkWidget parent_instance;
    GtkLinkButtonPrivate *priv;
} GtkLinkButton;
typedef struct _GtkLinkButtonClass {
    GtkWidgetClass parent_class;
    void (*_gtk_padding1) (void);
    void (*_gtk_padding2) (void);
    void (*_gtk_padding3) (void);
    void (*_gtk_padding4) (void);
} GtkLinkButtonClass;
typedef struct _GtkLinkButtonPrivate GtkLinkButtonPrivate;
typedef void (*GtkLinkButtonUriFunc) (GtkLinkButton *, const char
*,
                                      gpointer);

typedef GtkWidget (*GtkNotebookWindowCreationFunc) (GtkNotebook
*,
                                                    GtkWidget *, gint,
                                                    gint, gpointer);

typedef enum {
    GTK_ASSISTANT_PAGE_CONTENT,
    GTK_ASSISTANT_PAGE_INTRO,
    GTK_ASSISTANT_PAGE_CONFIRM,
    GTK_ASSISTANT_PAGE_SUMMARY,
    GTK_ASSISTANT_PAGE_PROGRESS
} GtkAssistantPageType;
typedef struct _GtkAssistant {
    GtkWidget parent;
    GtkWidget *cancel;
    GtkWidget *forward;
    GtkWidget *back;
    GtkWidget *apply;
    GtkWidget *close;
    GtkWidget *last;
    GtkAssistantPrivate *priv;
} GtkAssistant;
typedef struct _GtkAssistantPrivate GtkAssistantPrivate;
typedef struct _GtkAssistantClass {
    GtkWidgetClass parent_class;
    void (*prepare) (void);
    void (*apply) (void);
    void (*close) (void);
    void (*cancel) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
} GtkAssistantClass;
typedef gint (*GtkAssistantPageFunc) (gint, gpointer);
typedef enum {
    GTK_RECENT_SORT_NONE,

```

```

    GTK_RECENT_SORT_MRU,
    GTK_RECENT_SORT_LRU,
    GTK_RECENT_SORT_CUSTOM
} GtkRecentSortType;
typedef gint (*GtkRecentSortFunc) (GtkRecentInfo *, GtkRecentInfo *,
                                   gpointer);
typedef struct _GtkRecentChooser GtkRecentChooser;
typedef struct _GtkRecentChooserIface {
    GTypeInterface base_iface;
    gboolean (*set_current_uri) (void);
    gchar * (*get_current_uri) (void);
    gboolean (*select_uri) (void);
    void (*unselect_uri) (void);
    void (*select_all) (void);
    void (*unselect_all) (void);
    GList * (*get_items) (void);
    GtkRecentManager * (*get_recent_manager) (void);
    void (*add_filter) (void);
    void (*remove_filter) (void);
    GSList * (*list_filters) (void);
    void (*set_sort_func) (void);
    void (*item_activated) (void);
    void (*selection_changed) (void);
} GtkRecentChooserIface;
typedef enum {
    GTK_RECENT_CHOOSER_ERROR_NOT_FOUND,
    GTK_RECENT_CHOOSER_ERROR_INVALID_URI
} GtkRecentChooserError;
typedef struct _GtkRecentChooserDialog {
    GtkDialog parent_instance;
    GtkRecentChooserDialogPrivate *priv;
} GtkRecentChooserDialog;
typedef struct _GtkRecentChooserDialogClass {
    GtkDialogClass parent_class;
} GtkRecentChooserDialogClass;
typedef struct _GtkRecentChooserDialogPrivate {
    GtkRecentChooserDialogPrivate;
} GtkRecentChooserDialogPrivate;
typedef struct _GtkRecentChooserWidget {
    GtkVBox parent_instance;
    GtkRecentChooserWidgetPrivate *priv;
} GtkRecentChooserWidget;
typedef struct _GtkRecentChooserWidgetClass {
    GtkVBoxClass parent_class;
} GtkRecentChooserWidgetClass;
typedef struct _GtkRecentChooserWidgetPrivate {
    GtkRecentChooserWidgetPrivate;
} GtkRecentChooserWidgetPrivate;
typedef struct _GtkRecentChooserMenu {
    GtkMenu parent_instance;
    GtkRecentChooserMenuPrivate *priv;
} GtkRecentChooserMenu;
typedef struct _GtkRecentChooserMenuClass {
    GtkMenuClass parent_class;
    void (*gtk_recent1) (void);
    void (*gtk_recent2) (void);
    void (*gtk_recent3) (void);
    void (*gtk_recent4) (void);
} GtkRecentChooserMenuClass;
typedef struct _GtkRecentChooserMenuPrivate {
    GtkRecentChooserMenuPrivate;
} GtkRecentChooserMenuPrivate;
typedef void (*GtkClipboardRichTextReceivedFunc) (GtkClipboard *,
    GdkAtom,
    const unsigned char *,
    gsize, gpointer);

typedef struct _GtkCellRendererSpin {
    GtkCellRendererText parent;
} GtkCellRendererSpin;

```

```

typedef struct _GtkCellRendererSpinClass {
    GtkCellRendererTextClass parent;
} GtkCellRendererSpinClass;
typedef struct _GtkCellRendererSpinPrivate
GtkCellRendererSpinPrivate;
typedef struct _GtkCellRendererAccel {
    GtkCellRendererText parent;
    guint accel_key;
    GdkModifierType accel_mods;
    guint keycode;
    GtkCellRendererAccelMode accel_mode;
    GtkWidget *edit_widget;
    GtkWidget *grab_widget;
    GtkWidget *sizing_label;
} GtkCellRendererAccel;
typedef struct _GtkCellRendererAccelClass {
    GtkCellRendererTextClass parent_class;
    void (*accel_edited) (void);
    void (*accel_cleared) (void);
    void (*_gtk_reserved0) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
} GtkCellRendererAccelClass;
typedef enum {
    GTK_CELL_RENDERER_ACCEL_MODE_GTK,
    GTK_CELL_RENDERER_ACCEL_MODE_OTHER
} GtkCellRendererAccelMode;
typedef void (*GtkTreeViewSearchPositionFunc) (GtkTreeView *,
GtkWidget *,
gpointer);

typedef struct _GtkTooltip GtkTooltip;
extern const gchar *gtk_about_dialog_get_artists(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_authors(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_comments(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_copyright(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_documenters(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_license(GtkAboutDialog *
about);
extern GdkPixbuf *gtk_about_dialog_get_logo(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_logo_icon_name(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_name(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_translator_credits(GtkAboutDialog *
about);
extern GType gtk_about_dialog_get_type(void);
extern const gchar *gtk_about_dialog_get_version(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_website(GtkAboutDialog *
about);
extern const gchar *gtk_about_dialog_get_website_label(GtkAboutDialog *
about);

```

```

extern gboolean gtk_about_dialog_get_wrap_license(GtkAboutDialog *
about);
extern GtkWidget *gtk_about_dialog_new(void);
extern void gtk_about_dialog_set_artists(GtkAboutDialog * about,
const gchar * *artists);
extern void gtk_about_dialog_set_authors(GtkAboutDialog * about,
const gchar * *authors);
extern void gtk_about_dialog_set_comments(GtkAboutDialog * about,
const gchar * comments);
extern void gtk_about_dialog_set_copyright(GtkAboutDialog * about,
const gchar * copyright);
extern void gtk_about_dialog_set_documenters(GtkAboutDialog *
about,
const gchar * *documenters);
extern GtkAboutDialogActivateLinkFunc
gtk_about_dialog_set_email_hook(GtkAboutDialogActivateLinkFunc
func,
gpointer data, GDestroyNotify destroy);
extern void gtk_about_dialog_set_license(GtkAboutDialog * about,
const gchar * license);
extern void gtk_about_dialog_set_logo(GtkAboutDialog * about,
GdkPixbuf * logo);
extern void gtk_about_dialog_set_logo_icon_name(GtkAboutDialog *
about,
const gchar * icon_name);
extern void gtk_about_dialog_set_name(GtkAboutDialog * about,
const gchar * name);
extern void gtk_about_dialog_set_translator_credits(GtkAboutDialog
* about,
const gchar *
translator_credits);
extern GtkAboutDialogActivateLinkFunc
gtk_about_dialog_set_url_hook(GtkAboutDialogActivateLinkFunc func,
gpointer data, GDestroyNotify destroy);
extern void gtk_about_dialog_set_version(GtkAboutDialog * about,
const gchar * version);
extern void gtk_about_dialog_set_website(GtkAboutDialog * about,
const gchar * website);
extern void gtk_about_dialog_set_website_label(GtkAboutDialog *
about,
const gchar *
website_label);
extern void gtk_about_dialog_set_wrap_license(GtkAboutDialog *
about,
gboolean wrap_license);
extern GType gtk_accel_flags_get_type(void);
extern gboolean gtk_accel_group_activate(GtkAccelGroup *
accel_group,
GQuark accel_quark,
GObject * acceleratable,
guint accel_key,
GdkModifierType accel_mods);
extern void gtk_accel_group_connect(GtkAccelGroup * accel_group,
guint accel_key,
GdkModifierType accel_mods,
GtkAccelFlags accel_flags,
GClosure * closure);
extern void gtk_accel_group_connect_by_path(GtkAccelGroup *
accel_group,
const gchar * accel_path,
GClosure * closure);
extern gboolean gtk_accel_group_disconnect(GtkAccelGroup *
accel_group,
GClosure * closure);
extern gboolean gtk_accel_group_disconnect_key(GtkAccelGroup *
accel_group,

```

```

                                guint accel_key,
                                GdkModifierType

accel_mods);
extern      GtkAccelKey      *gtk_accel_group_find(GtkAccelGroup      *
accel_group,

                                GtkAccelGroupFindFunc find_func,
                                gpointer data);
extern GtkAccelGroup *gtk_accel_group_from_accel_closure(GClosure
*
                                closure);

extern GType gtk_accel_group_get_type(void);
extern void gtk_accel_group_lock(GtkAccelGroup * accel_group);
extern GtkAccelGroup *gtk_accel_group_new(void);
extern GtkAccelGroupEntry *gtk_accel_group_query(GtkAccelGroup *
                                accel_group,
                                guint accel_key,
                                GdkModifierType
                                accel_mods,
                                guint * n_entries);
extern void gtk_accel_group_unlock(GtkAccelGroup * accel_group);
extern gboolean gtk_accel_groups_activate(GObject * object,
                                guint accel_key,
                                GdkModifierType accel_mods);
extern GSList *gtk_accel_groups_from_object(GObject * object);
extern GtkWidget *gtk_accel_label_get_accel_widget(GtkAccelLabel *
                                accel_label);
extern      guint      gtk_accel_label_get_accel_width(GtkAccelLabel *
accel_label);
extern GType gtk_accel_label_get_type(void);
extern GtkWidget *gtk_accel_label_new(const gchar * string);
extern      gboolean      gtk_accel_label_refetch(GtkAccelLabel      *
accel_label);
extern void      gtk_accel_label_set_accel_closure(GtkAccelLabel *
accel_label,

                                GClosure * accel_closure);
extern void      gtk_accel_label_set_accel_widget(GtkAccelLabel *
accel_label,

                                GtkWidget * accel_widget);
extern void gtk_accel_map_add_entry(const gchar * accel_path,
                                guint accel_key,
                                GdkModifierType accel_mods);
extern void gtk_accel_map_add_filter(const gchar * filter_pattern);
extern gboolean gtk_accel_map_change_entry(const gchar * accel_path,
                                guint accel_key,
                                GdkModifierType accel_mods,
                                gboolean replace);
extern void gtk_accel_map_foreach(gpointer data,
                                GtkAccelMapForeach foreach_func);
extern void gtk_accel_map_foreach_unfiltered(gpointer data,
                                GtkAccelMapForeach
                                foreach_func);
extern GtkAccelMap *gtk_accel_map_get(void);
extern GType gtk_accel_map_get_type(void);
extern void gtk_accel_map_load(const gchar * file_name);
extern void gtk_accel_map_load_fd(gint fd);
extern void gtk_accel_map_load_scanner(GScanner * scanner);
extern void gtk_accel_map_lock_path(const gchar * accel_path);
extern gboolean gtk_accel_map_lookup_entry(const gchar * accel_path,
                                GtkAccelKey * key);
extern void gtk_accel_map_save(const gchar * file_name);
extern void gtk_accel_map_save_fd(gint fd);
extern void gtk_accel_map_unlock_path(const gchar * accel_path);
extern guint gtk_accelerator_get_default_mod_mask(void);
extern gchar *gtk_accelerator_get_label(guint accelerator_key,
                                GdkModifierType
                                accelerator_mods);

```

```

extern gchar *gtk_accelerator_name(guint accelerator_key,
                                   GdkModifierType accelerator_mods);
extern void gtk_accelerator_parse(const gchar * accelerator,
                                  guint * accelerator_key,
                                  GdkModifierType * accelerator_mods);
extern void gtk_accelerator_set_default_mod_mask(GdkModifierType
                                                  default_mod_mask);
extern gboolean gtk_accelerator_valid(guint keyval,
                                      GdkModifierType modifiers);
extern void gtk_accessible_connect_widget_destroyed(GtkAccessible
*
                                   accessible);
extern GType gtk_accessible_get_type(void);
extern void gtk_action_activate(GtkAction * action);
extern void gtk_action_block_activate_from(GtkAction * action,
                                           GtkWidget * proxy);
extern void gtk_action_connect_accelerator(GtkAction * action);
extern void gtk_action_connect_proxy(GtkAction * action,
                                     GtkWidget * proxy);
extern GtkWidget *gtk_action_create_icon(GtkAction * action,
                                         GtkIconSize icon_size);
extern GtkWidget *gtk_action_create_menu_item(GtkAction * action);
extern GtkWidget *gtk_action_create_tool_item(GtkAction * action);
extern void gtk_action_disconnect_accelerator(GtkAction * action);
extern void gtk_action_disconnect_proxy(GtkAction * action,
                                       GtkWidget * proxy);
extern GClosure *gtk_action_get_accel_closure(GtkAction * action);
extern const gchar *gtk_action_get_accel_path(GtkAction * action);
extern const gchar *gtk_action_get_name(GtkAction * action);
extern GSList *gtk_action_get_proxies(GtkAction * action);
extern gboolean gtk_action_get_sensitive(GtkAction * action);
extern GType gtk_action_get_type(void);
extern gboolean gtk_action_get_visible(GtkAction * action);
extern void gtk_action_group_add_action(GtkActionGroup *
action_group,
                                       GtkAction * action);
extern void gtk_action_group_add_action_with_accel(GtkActionGroup
*
                                       action_group,
                                       GtkAction * action,
                                       const gchar *
                                       accelerator);
extern void gtk_action_group_add_actions(GtkActionGroup *
action_group,
                                       const GtkActionEntry * entries,
                                       guint n_entries,
                                       gpointer user_data);
extern void gtk_action_group_add_actions_full(GtkActionGroup *
action_group,
                                       const GtkActionEntry *
                                       entries, guint n_entries,
                                       gpointer user_data,
                                       GDestroyNotify destroy);
extern void gtk_action_group_add_radio_actions(GtkActionGroup *
action_group,
                                       const GtkRadioActionEntry *
                                       entries, guint n_entries,
                                       gint value,
                                       GCallback on_change,
                                       gpointer user_data);
extern void gtk_action_group_add_radio_actions_full(GtkActionGroup
*
                                       action_group,
                                       const
                                       GtkRadioActionEntry *
                                       entries,

```

```

        guint n_entries,
        gint value,
        GCallback on_change,
        gpointer user_data,
        GDestroyNotify
        destroy);
extern void gtk_action_group_add_toggle_actions(GtkActionGroup *
        action_group,
        const GtkToggleActionEntry
        * entries, guint n_entries,
        gpointer user_data);
extern
        void
gtk_action_group_add_toggle_actions_full(GtkActionGroup *
        action_group,
        const
        GtkToggleActionEntry
        *
        entries,
        guint n_entries,
        gpointer user_data,
        GDestroyNotify
        destroy);
extern GtkAction *gtk_action_group_get_action(GtkActionGroup *
        action_group,
        const gchar * action_name);
extern const gchar *gtk_action_group_get_name(GtkActionGroup *
        action_group);
extern gboolean gtk_action_group_get_sensitive(GtkActionGroup *
        action_group);
extern GType gtk_action_group_get_type(void);
extern gboolean gtk_action_group_get_visible(GtkActionGroup *
        action_group);
extern GList *gtk_action_group_list_actions(GtkActionGroup *
        action_group);
extern GtkActionGroup *gtk_action_group_new(const gchar * name);
extern void gtk_action_group_remove_action(GtkActionGroup *
        action_group,
        GtkAction * action);
extern void gtk_action_group_set_sensitive(GtkActionGroup *
        action_group,
        gboolean sensitive);
extern void gtk_action_group_set_translate_func(GtkActionGroup *
        action_group,
        GtkTranslateFunc func,
        gpointer data,
        GDestroyNotify notify);
extern void gtk_action_group_set_translation_domain(GtkActionGroup
        *
        action_group,
        const gchar * domain);
extern void gtk_action_group_set_visible(GtkActionGroup *
        action_group,
        gboolean visible);
extern
        const
        gchar
*gtk_action_group_translate_string(GtkActionGroup *
        action_group,
        const gchar *
        string);
extern gboolean gtk_action_is_sensitive(GtkAction * action);
extern gboolean gtk_action_is_visible(GtkAction * action);
extern GtkAction *gtk_action_new(const gchar * name, const gchar *
        label,
        const gchar * tooltip,
        const gchar * stock_id);
extern void gtk_action_set_accel_group(GtkAction * action,
        GtkAccelGroup * accel_group);

```



```

extern void gtk_action_set_accel_path(GtkAction * action,
                                     const gchar * accel_path);
extern void gtk_action_set_sensitive(GtkAction * action,
                                     gboolean sensitive);
extern void gtk_action_set_visible(GtkAction * action, gboolean
visible);
extern void gtk_action_unblock_activate_from(GtkAction * action,
                                             GtkWidget * proxy);
extern void gtk_adjustment_changed(GtkAdjustment * adjustment);
extern void gtk_adjustment_clamp_page(GtkAdjustment * adjustment,
                                       gdouble lower, gdouble upper);
extern GType gtk_adjustment_get_type(void);
extern gdouble gtk_adjustment_get_value(GtkAdjustment *
adjustment);
extern GObject *gtk_adjustment_new(gdouble value, gdouble lower,
                                   gdouble upper, gdouble
step_increment,
                                   gdouble page_increment,
                                   gdouble page_size);
extern void gtk_adjustment_set_value(GtkAdjustment * adjustment,
                                     gdouble value);
extern void gtk_adjustment_value_changed(GtkAdjustment *
adjustment);
extern void gtk_alignment_get_padding(GtkAlignment * alignment,
                                       guint * padding_top,
                                       guint * padding_bottom,
                                       guint * padding_left,
                                       guint * padding_right);
extern GType gtk_alignment_get_type(void);
extern GtkWidget *gtk_alignment_new(gfloat xalign, gfloat yalign,
                                    gfloat xscale, gfloat yscale);
extern void gtk_alignment_set(GtkAlignment * alignment, gfloat
xalign,
                                gfloat yalign, gfloat xscale, gfloat
yscale);
extern void gtk_alignment_set_padding(GtkAlignment * alignment,
                                       guint padding_top,
                                       guint padding_bottom,
                                       guint padding_left,
                                       guint padding_right);
extern gboolean gtk_alternative_dialog_button_order(GdkScreen *
screen);
extern GType gtk_anchor_type_get_type(void);
extern GType gtk_arg_flags_get_type(void);
extern GType gtk_arrow_get_type(void);
extern GtkWidget *gtk_arrow_new(GtkArrowType arrow_type,
                                GtkShadowType shadow_type);
extern void gtk_arrow_set(GtkArrow * arrow, GtkArrowType arrow_type,
                           GtkShadowType shadow_type);
extern GType gtk_arrow_type_get_type(void);
extern GType gtk_aspect_frame_get_type(void);
extern GtkWidget *gtk_aspect_frame_new(const gchar * label, gfloat
xalign,
                                       gfloat yalign, gfloat ratio,
                                       gboolean obey_child);
extern void gtk_aspect_frame_set(GtkAspectRatio * aspect_frame,
                                  gfloat xalign, gfloat yalign,
                                  gfloat ratio, gboolean obey_child);
extern void gtk_assistant_add_action_widget(GtkAssistant *
assistant,
                                             GtkWidget * child);
extern gint gtk_assistant_append_page(GtkAssistant * assistant,
                                       GtkWidget * page);
extern gint gtk_assistant_get_current_page(GtkAssistant *
assistant);
extern gint gtk_assistant_get_n_pages(GtkAssistant * assistant);

```

```

extern GtkWidget *gtk_assistant_get_nth_page(GtkAssistant *
assistant,
                                           gint page_num);
extern gboolean gtk_assistant_get_page_complete(GtkAssistant *
assistant,
                                           GtkWidget * page);
extern GdkPixbuf *gtk_assistant_get_page_header_image(GtkAssistant
*
                                           assistant,
                                           GtkWidget * page);
extern GdkPixbuf *gtk_assistant_get_page_side_image(GtkAssistant *
assistant,
                                           GtkWidget * page);
extern const char *gtk_assistant_get_page_title(GtkAssistant *
assistant,
                                           GtkWidget * page);
extern
                                           GtkAssistantPageType
gtk_assistant_get_page_type(GtkAssistant *
assistant,
                                           GtkWidget * page);

extern GType gtk_assistant_get_type(void);
extern gint gtk_assistant_insert_page(GtkAssistant * assistant,
                                           GtkWidget * page, gint position);
extern GtkWidget *gtk_assistant_new(void);
extern GType gtk_assistant_page_type_get_type(void);
extern gint gtk_assistant_prepend_page(GtkAssistant * assistant,
                                           GtkWidget * page);
extern void gtk_assistant_remove_action_widget(GtkAssistant *
assistant,
                                           GtkWidget * child);
extern void gtk_assistant_set_current_page(GtkAssistant *
assistant,
                                           gint page_num);
extern void gtk_assistant_set_forward_page_func(GtkAssistant *
assistant,
                                           GtkAssistantPageFunc
                                           page_func, gpointer data,
                                           GDestroyNotify destroy);
extern void gtk_assistant_set_page_complete(GtkAssistant *
assistant,
                                           GtkWidget * page,
                                           gboolean complete);
extern void gtk_assistant_set_page_header_image(GtkAssistant *
assistant,
                                           GtkWidget * page,
                                           GdkPixbuf * pixbuf);
extern void gtk_assistant_set_page_side_image(GtkAssistant *
assistant,
                                           GtkWidget * page,
                                           GdkPixbuf * pixbuf);
extern void gtk_assistant_set_page_title(GtkAssistant * assistant,
                                           GtkWidget * page,
                                           const char *title);
extern void gtk_assistant_set_page_type(GtkAssistant * assistant,
                                           GtkWidget * page,
                                           GtkAssistantPageType type);
extern void gtk_assistant_update_buttons_state(GtkAssistant *
assistant);
extern GType gtk_attach_options_get_type(void);
extern GtkWidget *gtk_bin_get_child(GtkBin * bin);
extern GType gtk_bin_get_type(void);
extern const guint gtk_binary_age;
extern void gtk_binding_entry_add_signal(GtkBindingSet *
binding_set,
                                           guint keyval,
                                           GdkModifierType modifiers,

```

```

        const gchar * signal_name,
        guint n_args, ...);
extern void gtk_binding_entry_add_signall(GtkBindingSet *
binding_set,
        guint keyval,
        GdkModifierType modifiers,
        const gchar * signal_name,
        GSList * binding_args);
extern void gtk_binding_entry_clear(GtkBindingSet * binding_set,
        guint keyval,
        GdkModifierType modifiers);
extern void gtk_binding_entry_remove(GtkBindingSet * binding_set,
        guint keyval,
        GdkModifierType modifiers);
extern guint gtk_binding_parse_binding(GScanner * scanner);
extern gboolean gtk_binding_set_activate(GtkBindingSet
binding_set,
        guint keyval,
        GdkModifierType modifiers,
        GObject * object);
extern void gtk_binding_set_add_path(GtkBindingSet * binding_set,
        GtkPathType path_type,
        const gchar * path_pattern,
        GtkPathPriorityType priority);
extern GtkBindingSet *gtk_binding_set_by_class(gpointer
object_class);
extern GtkBindingSet *gtk_binding_set_find(const gchar * set_name);
extern GtkBindingSet *gtk_binding_set_new(const gchar * set_name);
extern gboolean gtk_bindings_activate(GObject * object, guint
keyval,
        GdkModifierType modifiers);
extern gboolean gtk_bindings_activate_event(GObject * object,
        GdkEventKey * event);
extern GtkBorder *gtk_border_copy(const GtkBorder * border_);
extern void gtk_border_free(GtkBorder * border_);
extern GType gtk_border_get_type(void);
extern gboolean gtk_box_get_homogeneous(GtkBox * box);
extern gint gtk_box_get_spacing(GtkBox * box);
extern GType gtk_box_get_type(void);
extern void gtk_box_pack_end(GtkBox * box, GtkWidget * child,
        gboolean expand, gboolean fill,
        guint padding);
extern void gtk_box_pack_end_defaults(GtkBox * box, GtkWidget *
widget);
extern void gtk_box_pack_start(GtkBox * box, GtkWidget * child,
        gboolean expand, gboolean fill,
        guint padding);
extern void gtk_box_pack_start_defaults(GtkBox * box, GtkWidget *
widget);
extern void gtk_box_query_child_packing(GtkBox * box, GtkWidget *
child,
        gboolean * expand, gboolean *
        fill,
        guint * padding,
        GtkPackType * pack_type);
extern void gtk_box_reorder_child(GtkBox * box, GtkWidget * child,
        gint position);
extern void gtk_box_set_child_packing(GtkBox * box, GtkWidget *
child,
        gboolean expand, gboolean fill,
        guint padding,
        GtkPackType pack_type);
extern void gtk_box_set_homogeneous(GtkBox * box, gboolean
homogeneous);
extern void gtk_box_set_spacing(GtkBox * box, gint spacing);

```

```

extern gboolean gtk_button_box_get_child_secondary(GtkButtonBox *
widget,
                                           GtkWidget * child);
extern GtkWidgetStyle gtk_button_box_get_layout(GtkButtonBox *
widget);
extern GType gtk_button_box_get_type(void);
extern void gtk_button_box_set_child_secondary(GtkButtonBox *
widget,
                                           GtkWidget * child,
                                           gboolean is_secondary);
extern void gtk_button_box_set_layout(GtkButtonBox * widget,
                                           GtkWidgetStyle layout_style);
extern GType gtk_button_box_style_get_type(void);
extern void gtk_button_clicked(GtkButton * button);
extern void gtk_button_enter(GtkButton * button);
extern void gtk_button_get_alignment(GtkButton * button, gfloat *
xalign,
                                           gfloat * yalign);
extern gboolean gtk_button_get_focus_on_click(GtkButton * button);
extern GtkWidget *gtk_button_get_image(GtkButton * button);
extern GtkPositionType gtk_button_get_image_position(GtkButton *
button);
extern const gchar *gtk_button_get_label(GtkButton * button);
extern GtkReliefStyle gtk_button_get_relief(GtkButton * button);
extern GType gtk_button_get_type(void);
extern gboolean gtk_button_get_use_stock(GtkButton * button);
extern gboolean gtk_button_get_use_underline(GtkButton * button);
extern void gtk_button_leave(GtkButton * button);
extern GtkWidget *gtk_button_new(void);
extern GtkWidget *gtk_button_new_from_stock(const gchar * stock_id);
extern GtkWidget *gtk_button_new_with_label(const gchar * label);
extern GtkWidget *gtk_button_new_with_mnemonic(const gchar * label);
extern void gtk_button_pressed(GtkButton * button);
extern void gtk_button_released(GtkButton * button);
extern void gtk_button_set_alignment(GtkButton * button, gfloat
xalign,
                                           gfloat yalign);
extern void gtk_button_set_focus_on_click(GtkButton * button,
                                           gboolean focus_on_click);
extern void gtk_button_set_image(GtkButton * button, GtkWidget *
image);
extern void gtk_button_set_image_position(GtkButton * button,
                                           GtkPositionType position);
extern void gtk_button_set_label(GtkButton * button, const gchar *
label);
extern void gtk_button_set_relief(GtkButton * button,
                                           GtkReliefStyle newstyle);
extern void gtk_button_set_use_stock(GtkButton * button,
                                           gboolean use_stock);
extern void gtk_button_set_use_underline(GtkButton * button,
                                           gboolean use_underline);
extern GType gtk_buttons_type_get_type(void);
extern void gtk_calendar_clear_marks(GtkCalendar * calendar);
extern GType gtk_calendar_display_options_get_type(void);
extern void gtk_calendar_get_date(GtkCalendar * calendar, guint *
year,
                                           guint * month, guint * day);
extern GtkCalendarDisplayOptions
gtk_calendar_get_display_options(GtkCalendar * calendar);
extern GType gtk_calendar_get_type(void);
extern gboolean gtk_calendar_mark_day(GtkCalendar * calendar, guint
day);
extern GtkWidget *gtk_calendar_new(void);
extern void gtk_calendar_select_day(GtkCalendar * calendar, guint
day);
extern gboolean gtk_calendar_select_month(GtkCalendar * calendar,

```

```

                                guint month, guint year);
extern void gtk_calendar_set_display_options(GtkCalendar *
calendar,
                                GtkCalendarDisplayOptions
                                flags);
extern gboolean gtk_calendar_unmark_day(GtkCalendar * calendar,
guint day);
extern void gtk_cell_editable_editing_done(GtkCellEditable *
cell_editable);
extern GType gtk_cell_editable_get_type(void);
extern void gtk_cell_editable_remove_widget(GtkCellEditable *
cell_editable);
extern void gtk_cell_editable_start_editing(GtkCellEditable *
cell_editable,
GdkEvent * event);
extern void gtk_cell_layout_add_attribute(GtkCellLayout *
cell_layout,
                                GtkCellRenderer * cell,
                                const gchar * attribute,
                                gint column);
extern void gtk_cell_layout_clear(GtkCellLayout * cell_layout);
extern void gtk_cell_layout_clear_attributes(GtkCellLayout *
cell_layout,
                                GtkCellRenderer * cell);
extern GType gtk_cell_layout_get_type(void);
extern void gtk_cell_layout_pack_end(GtkCellLayout * cell_layout,
                                GtkCellRenderer * cell,
                                gboolean expand);
extern void gtk_cell_layout_pack_start(GtkCellLayout * cell_layout,
                                GtkCellRenderer * cell,
                                gboolean expand);
extern void gtk_cell_layout_reorder(GtkCellLayout * cell_layout,
                                GtkCellRenderer * cell,
                                gint
                                position);
extern void gtk_cell_layout_set_attributes(GtkCellLayout *
cell_layout,
                                GtkCellRenderer * cell, ...);
extern void gtk_cell_layout_set_cell_data_func(GtkCellLayout *
cell_layout,
                                GtkCellRenderer * cell,
                                GtkCellLayoutDataFunc func,
                                gpointer func_data,
                                GDestroyNotify destroy);
extern GType gtk_cell_renderer_accel_get_type(void);
extern GType gtk_cell_renderer_accel_mode_get_type(void);
extern GtkCellRenderer *gtk_cell_renderer_accel_new(void);
extern gboolean gtk_cell_renderer_activate(GtkCellRenderer * cell,
                                GdkEvent * event,
                                GtkWidget * widget,
                                const gchar * path,
                                const GdkRectangle *
                                background_area,
                                const GdkRectangle * cell_area,
                                GtkCellRendererState flags);
extern GType gtk_cell_renderer_combo_get_type(void);
extern GtkCellRenderer *gtk_cell_renderer_combo_new(void);
extern void gtk_cell_renderer_get_fixed_size(GtkCellRenderer *
cell,
                                gint * width, gint * height);
extern void gtk_cell_renderer_get_size(GtkCellRenderer * cell,
                                GtkWidget * widget,
                                const GdkRectangle * cell_area,
                                gint * x_offset, gint * y_offset,
                                gint * width, gint * height);
extern GType gtk_cell_renderer_get_type(void);
extern GType gtk_cell_renderer_mode_get_type(void);

```

```

extern GType gtk_cell_renderer_pixbuf_get_type(void);
extern GtkWidget *gtk_cell_renderer_pixbuf_new(void);
extern GType gtk_cell_renderer_progress_get_type(void);
extern GtkWidget *gtk_cell_renderer_progress_new(void);
extern void gtk_cell_renderer_render(GtkCellRenderer * cell,
                                     GdkWindow * window,
                                     GtkWidget * widget,
                                     const GdkRectangle *
background_area,
                                     const GdkRectangle * cell_area,
                                     const GdkRectangle * expose_area,
                                     GtkWidget * widget,
                                     const GdkRectangle *
                                     GdkCellRendererState flags);
extern void gtk_cell_renderer_set_fixed_size(GtkCellRenderer *
cell,
                                             gint width, gint height);
extern GType gtk_cell_renderer_spin_get_type(void);
extern GtkWidget *gtk_cell_renderer_spin_new(void);
extern GtkWidget *gtk_cell_renderer_start_editing(GtkCellRenderer *
cell,
GdkEvent * event,
GtkWidget * widget,
const gchar * path,
const GdkRectangle
* background_area,
const GdkRectangle
* cell_area,

GtkCellRendererState
flags);
extern GType gtk_cell_renderer_state_get_type(void);
extern void gtk_cell_renderer_stop_editing(GtkCellRenderer * cell,
gboolean canceled);
extern GType gtk_cell_renderer_text_get_type(void);
extern GtkWidget *gtk_cell_renderer_text_new(void);
extern void
gtk_cell_renderer_text_set_fixed_height_from_font(GtkCellRenderer
Text *
render,
gint number_of_rows);
extern gboolean
gtk_cell_renderer_toggle_get_active(GtkCellRendererToggle *
toggle);
extern gboolean
gtk_cell_renderer_toggle_get_radio(GtkCellRendererToggle *
toggle);
extern GType gtk_cell_renderer_toggle_get_type(void);
extern GtkWidget *gtk_cell_renderer_toggle_new(void);
extern void
gtk_cell_renderer_toggle_set_active(GtkCellRendererToggle *
toggle, gboolean setting);
extern void
gtk_cell_renderer_toggle_set_radio(GtkCellRendererToggle *
toggle, gboolean radio);
extern GList *gtk_cell_view_get_cell_renderers(GtkCellView *
cell_view);
extern GtkTreePath *gtk_cell_view_get_displayed_row(GtkCellView *
cell_view);
extern gboolean gtk_cell_view_get_size_of_row(GtkCellView *
cell_view,
GtkTreePath * path,
GtkRequisition *
requisition);
extern GType gtk_cell_view_get_type(void);
extern GtkWidget *gtk_cell_view_new(void);

```

```

extern GtkWidget *gtk_cell_view_new_with_markup(const gchar *
markup);
extern GtkWidget *gtk_cell_view_new_with_pixbuf(GdkPixbuf *
pixbuf);
extern GtkWidget *gtk_cell_view_new_with_text(const gchar * text);
extern void gtk_cell_view_set_background_color(GtkCellView *
cell_view,
const GdkColor * color);
extern void gtk_cell_view_set_displayed_row(GtkCellView *
cell_view,
GtkTreePath * path);
extern void gtk_cell_view_set_model(GtkCellView * cell_view,
GtkTreeModel * model);
extern GType gtk_check_button_get_type(void);
extern GtkWidget *gtk_check_button_new(void);
extern GtkWidget *gtk_check_button_new_with_label(const gchar *
label);
extern GtkWidget *gtk_check_button_new_with_mnemonic(const gchar *
label);
extern gboolean gtk_check_menu_item_get_active(GtkCheckMenuItem *
check_menu_item);
extern
gboolean
gtk_check_menu_item_get_draw_as_radio(GtkCheckMenuItem *
check_menu_item);
extern
gboolean
gtk_check_menu_item_get_inconsistent(GtkCheckMenuItem *
check_menu_item);
extern GType gtk_check_menu_item_get_type(void);
extern GtkWidget *gtk_check_menu_item_new(void);
extern GtkWidget *gtk_check_menu_item_new_with_label(const gchar *
label);
extern GtkWidget *gtk_check_menu_item_new_with_mnemonic(const
gchar *
label);
extern void gtk_check_menu_item_set_active(GtkCheckMenuItem *
check_menu_item,
gboolean is_active);
extern void gtk_check_menu_item_set_draw_as_radio(GtkCheckMenuItem
*
check_menu_item,
gboolean draw_as_radio);
extern void gtk_check_menu_item_set_inconsistent(GtkCheckMenuItem
*
check_menu_item,
gboolean setting);
extern void gtk_check_menu_item_toggled(GtkCheckMenuItem *
check_menu_item);
extern const gchar *gtk_check_version(guint required_major,
guint required_minor,
guint required_micro);
extern void gtk_clipboard_clear(GtkClipboard * clipboard);
extern GtkWidget *gtk_clipboard_get(GdkAtom selection);
extern GdkDisplay *gtk_clipboard_get_display(GtkClipboard *
clipboard);
extern GtkWidget *gtk_clipboard_get_for_display(GdkDisplay *
display,
GdkAtom selection);
extern GObject *gtk_clipboard_get_owner(GtkClipboard * clipboard);
extern GType gtk_clipboard_get_type(void);
extern void gtk_clipboard_request_contents(GtkClipboard *
clipboard,
GdkAtom target,
GtkClipboardReceivedFunc
callback, gpointer user_data);
extern void gtk_clipboard_request_image(GtkClipboard * clipboard,
GtkClipboardImageReceivedFunc

```

```

                                callback, gpointer user_data);
extern void gtk_clipboard_request_rich_text(GtkClipboard *
clipboard,
                                GtkTextBuffer * buffer,

                                GtkClipboardRichTextReceivedFunc
                                callback, gpointer user_data);
extern void gtk_clipboard_request_targets(GtkClipboard * clipboard,

                                GtkClipboardTargetsReceivedFunc
                                callback, gpointer user_data);
extern void gtk_clipboard_request_text(GtkClipboard * clipboard,
                                GtkClipboardTextReceivedFunc
                                callback, gpointer user_data);
extern void gtk_clipboard_set_can_store(GtkClipboard * clipboard,
                                const GtkTargetEntry * targets,
                                gint n_targets);
extern void gtk_clipboard_set_image(GtkClipboard * clipboard,
                                GdkPixbuf * pixbuf);
extern void gtk_clipboard_set_text(GtkClipboard * clipboard,
                                const gchar * text, gint len);
extern gboolean gtk_clipboard_set_with_data(GtkClipboard *
clipboard,
                                const GtkTargetEntry *
targets,
                                guint n_targets,
                                GtkClipboardGetFunc get_func,
                                GtkClipboardClearFunc
                                clear_func,
                                gpointer user_data);
extern gboolean gtk_clipboard_set_with_owner(GtkClipboard *
clipboard,
                                const GtkTargetEntry *
targets, guint n_targets,
                                GtkClipboardGetFunc get_func,
                                GtkClipboardClearFunc
                                clear_func, GObject * owner);
extern void gtk_clipboard_store(GtkClipboard * clipboard);
extern void gtk_clipboard_wait_for_contents(GtkClipboard *
clipboard,
                                GdkAtom target);
extern GdkPixbuf *gtk_clipboard_wait_for_image(GtkClipboard *
clipboard);
extern guint8 *gtk_clipboard_wait_for_rich_text(GtkClipboard *
clipboard,
                                GtkTextBuffer * buffer,
                                GdkAtom * format,
                                gsize * length);
extern gboolean gtk_clipboard_wait_for_targets(GtkClipboard *
clipboard,
                                GdkAtom * *targets,
                                gint * n_targets);
extern gchar *gtk_clipboard_wait_for_text(GtkClipboard *
clipboard);
extern gboolean gtk_clipboard_wait_is_image_available(GtkClipboard
*
                                clipboard);
extern gboolean
gtk_clipboard_wait_is_rich_text_available(GtkClipboard *
                                clipboard,
                                GtkTextBuffer *
                                buffer);
extern gboolean
gtk_clipboard_wait_is_target_available(GtkClipboard *
                                clipboard,

```



```

                                GdkAtom target);
extern gboolean gtk_clipboard_wait_is_text_available(GtkClipboard
*
                                clipboard);
extern guint16  gtk_color_button_get_alpha(GtkColorButton  *
color_button);
extern void     gtk_color_button_get_color(GtkColorButton  *
color_button,
                                GdkColor * color);
extern const gchar *gtk_color_button_get_title(GtkColorButton *
                                color_button);
extern GType gtk_color_button_get_type(void);
extern gboolean gtk_color_button_get_use_alpha(GtkColorButton *
                                color_button);
extern GtkWidget *gtk_color_button_new(void);
extern GtkWidget *gtk_color_button_new_with_color(const GdkColor *
color);
extern void     gtk_color_button_set_alpha(GtkColorButton *
color_button,
                                guint16 alpha);
extern void     gtk_color_button_set_color(GtkColorButton  *
color_button,
                                const GdkColor * color);
extern void     gtk_color_button_set_title(GtkColorButton  *
color_button,
                                const gchar * title);
extern void     gtk_color_button_set_use_alpha(GtkColorButton *
color_button,
                                gboolean use_alpha);
extern GType gtk_color_selection_dialog_get_type(void);
extern GtkWidget *gtk_color_selection_dialog_new(const gchar *
title);
extern
                                guint16
gtk_color_selection_get_current_alpha(GtkColorSelection *
                                colorsel);
extern
                                void
gtk_color_selection_get_current_color(GtkColorSelection *
                                colorsel,
                                GdkColor * color);
extern gboolean
gtk_color_selection_get_has_opacity_control(GtkColorSelection *
colorsel);
extern
                                gboolean
gtk_color_selection_get_has_palette(GtkColorSelection *
                                colorsel);
extern
                                guint16
gtk_color_selection_get_previous_alpha(GtkColorSelection *
                                colorsel);
extern
                                void
gtk_color_selection_get_previous_color(GtkColorSelection *
                                colorsel,
                                GdkColor * color);
extern GType gtk_color_selection_get_type(void);
extern gboolean gtk_color_selection_is_adjusting(GtkColorSelection
*
                                colorsel);
extern GtkWidget *gtk_color_selection_new(void);
extern gboolean  gtk_color_selection_palette_from_string(const
gchar * str,
                                GdkColor * *colors,
                                gint * n_colors);
extern gchar *gtk_color_selection_palette_to_string(const GdkColor
*
                                colors, gint n_colors);
extern GtkColorSelectionChangePaletteWithScreenFunc
gtk_color_selection_set_change_palette_with_screen_hook

```

```

(GtkColorSelectionChangePaletteWithScreenFunc func);
extern void
gtk_color_selection_set_current_alpha(GtkColorSelection *
                                     colorsel, guint16 alpha);
extern void
gtk_color_selection_set_current_color(GtkColorSelection *
                                     colorsel,
                                     const GdkColor * color);
extern void
gtk_color_selection_set_has_opacity_control(GtkColorSelection *
                                           colorsel,
                                           gboolean
                                           has_opacity);
extern void gtk_color_selection_set_has_palette(GtkColorSelection
*
                                           colorsel,
                                           gboolean has_palette);
extern void
gtk_color_selection_set_previous_alpha(GtkColorSelection *
                                       colorsel,
                                       guint16 alpha);
extern void
gtk_color_selection_set_previous_color(GtkColorSelection *
                                       colorsel,
                                       const GdkColor * color);
extern void gtk_combo_box_append_text(GtkComboBox * combo_box,
                                       const gchar * text);
extern gint gtk_combo_box_entry_get_text_column(GtkComboBoxEntry *
                                                entry_box);
extern GType gtk_combo_box_entry_get_type(void);
extern GtkWidget *gtk_combo_box_entry_new(void);
extern GtkWidget *gtk_combo_box_entry_new_text(void);
extern GtkWidget *gtk_combo_box_entry_new_with_model(GtkTreeModel
* model,
                                                    gint text_column);
extern void gtk_combo_box_entry_set_text_column(GtkComboBoxEntry *
                                                entry_box,
                                                gint text_column);
extern gint gtk_combo_box_get_active(GtkComboBox * combo_box);
extern gboolean gtk_combo_box_get_active_iter(GtkComboBox *
                                              combo_box,
                                              GtkTreeIter * iter);
extern gchar *gtk_combo_box_get_active_text(GtkComboBox *
                                             combo_box);
extern gboolean gtk_combo_box_get_add_tearoffs(GtkComboBox *
                                              combo_box);
extern gint gtk_combo_box_get_column_span_column(GtkComboBox *
                                                  combo_box);
extern gboolean gtk_combo_box_get_focus_on_click(GtkComboBox *
                                                  combo_box);
extern GtkTreeModel *gtk_combo_box_get_model(GtkComboBox *
                                              combo_box);
extern AtkObject *gtk_combo_box_get_popup_accessible(GtkComboBox *
                                                     combo_box);
extern GtkTreeViewRowSeparatorFunc
gtk_combo_box_get_row_separator_func(GtkComboBox * combo_box);
extern gint gtk_combo_box_get_row_span_column(GtkComboBox *
                                              combo_box);
extern const gchar *gtk_combo_box_get_title(GtkComboBox *
                                             combo_box);
extern GType gtk_combo_box_get_type(void);
extern gint gtk_combo_box_get_wrap_width(GtkComboBox * combo_box);
extern void gtk_combo_box_insert_text(GtkComboBox * combo_box,
                                       gint position, const gchar * text);
extern GtkWidget *gtk_combo_box_new(void);
extern GtkWidget *gtk_combo_box_new_text(void);

```

```

extern GtkWidget *gtk_combo_box_new_with_model(GtkTreeModel *
model);
extern void gtk_combo_box_popdown(GtkComboBox * combo_box);
extern void gtk_combo_box_popup(GtkComboBox * combo_box);
extern void gtk_combo_box_prepend_text(GtkComboBox * combo_box,
const gchar * text);
extern void gtk_combo_box_remove_text(GtkComboBox * combo_box,
gint position);
extern void gtk_combo_box_set_active(GtkComboBox * combo_box, gint
index_);
extern void gtk_combo_box_set_active_iter(GtkComboBox * combo_box,
GtkTreeIter * iter);
extern void gtk_combo_box_set_add_tearoffs(GtkComboBox * combo_box,
gboolean add_tearoffs);
extern void gtk_combo_box_set_column_span_column(GtkComboBox *
combo_box,
gint column_span);
extern void gtk_combo_box_set_focus_on_click(GtkComboBox * combo,
gboolean focus_on_click);
extern void gtk_combo_box_set_model(GtkComboBox * combo_box,
GtkTreeModel * model);
extern void gtk_combo_box_set_row_separator_func(GtkComboBox *
combo_box,

GtkTreeViewRowSeparatorFunc
func, gpointer data,
GtkDestroyNotify destroy);
extern void gtk_combo_box_set_row_span_column(GtkComboBox *
combo_box,
gint row_span);
extern void gtk_combo_box_set_title(GtkComboBox * combo_box,
const gchar * title);
extern void gtk_combo_box_set_wrap_width(GtkComboBox * combo_box,
gint width);
extern GType gtk_combo_get_type(void);
extern void gtk_container_add(GtkContainer * container,
GtkWidget * widget);
extern void gtk_container_add_with_properties(GtkContainer *
container,
GtkWidget * widget,
const gchar *
first_prop_name, ...);
extern void gtk_container_check_resize(GtkContainer * container);
extern void gtk_container_child_get(GtkContainer * container,
GtkWidget * child,
const gchar * first_prop_name, ...);
extern void gtk_container_child_get_property(GtkContainer *
container,
GtkWidget * child,
const gchar * property_name,
GValue * value);
extern void gtk_container_child_get_valist(GtkContainer *
container,
GtkWidget * child,
const gchar *
first_property_name,
va_list var_args);
extern void gtk_container_child_set(GtkContainer * container,
GtkWidget * child,
const gchar * first_prop_name, ...);
extern void gtk_container_child_set_property(GtkContainer *
container,
GtkWidget * child,
const gchar * property_name,
const GValue * value);

```

```

extern void gtk_container_child_set_valist(GtkContainer *
container,
                                GtkWidget * child,
                                const gchar *
                                first_property_name,
                                va_list var_args);
extern GType gtk_container_child_type(GtkContainer * container);
extern GParamSpec *
*gtk_container_class_find_child_property(GObjectClass *
                                cclass,
                                const gchar *
                                property_name);
extern void
gtk_container_class_install_child_property(GtkContainerClass *
                                cclass,
                                guint property_id,
                                GParamSpec * pspec);
extern GParamSpec *
*gtk_container_class_list_child_properties(GObjectClass
                                * cclass,
                                guint *
                                n_properties);
extern void gtk_container_forall(GtkContainer * container,
                                GtkCallback callback,
                                gpointer callback_data);
extern void gtk_container_foreach(GtkContainer * container,
                                GtkCallback callback,
                                gpointer callback_data);
extern guint gtk_container_get_border_width(GtkContainer *
container);
extern GList *gtk_container_get_children(GtkContainer * container);
extern gboolean gtk_container_get_focus_chain(GtkContainer *
container,
                                GList * *focusable_widgets);
extern GtkAdjustment *
*gtk_container_get_focus_hadjustment(GtkContainer *
                                container);
extern GtkAdjustment *
*gtk_container_get_focus_vadjustment(GtkContainer *
                                container);
extern GtkResizeMode gtk_container_get_resize_mode(GtkContainer *
container);
extern GType gtk_container_get_type(void);
extern void gtk_container_propagate_expose(GtkContainer *
container,
                                GtkWidget * child,
                                GdkEventExpose * event);
extern void gtk_container_remove(GtkContainer * container,
                                GtkWidget * widget);
extern void gtk_container_resize_children(GtkContainer *
container);
extern void gtk_container_set_border_width(GtkContainer *
container,
                                guint border_width);
extern void gtk_container_set_focus_chain(GtkContainer * container,
                                GList * focusable_widgets);
extern void gtk_container_set_focus_child(GtkContainer * container,
                                GtkWidget * child);
extern void gtk_container_set_focus_hadjustment(GtkContainer *
container,
                                GtkAdjustment *
                                adjustment);
extern void gtk_container_set_focus_vadjustment(GtkContainer *
container,
                                GtkAdjustment *
                                adjustment);

```

```

extern void gtk_container_set_reallocate_redraws(GtkContainer *
container,
                                gboolean needs_redraws);
extern void gtk_container_set_resize_mode(GtkContainer * container,
                                GtkResizeMode resize_mode);
extern void gtk_container_unset_focus_chain(GtkContainer *
container);
extern GType gtk_corner_type_get_type(void);
extern GType gtk_curve_get_type(void);
extern void gtk_curve_get_vector(GtkCurve * curve, int veclen,
                                gfloat * vector);
extern GtkWidget *gtk_curve_new(void);
extern void gtk_curve_reset(GtkCurve * curve);
extern void gtk_curve_set_curve_type(GtkCurve * curve, GtkCurveType
type);
extern void gtk_curve_set_gamma(GtkCurve * curve, gfloat gamma);
extern void gtk_curve_set_range(GtkCurve * curve, gfloat min_x,
                                gfloat max_x, gfloat min_y, gfloat
max_y);
extern void gtk_curve_set_vector(GtkCurve * curve, int veclen,
                                gfloat * vector);
extern GType gtk_curve_type_get_type(void);
extern GType gtk_debug_flag_get_type(void);
extern guint gtk_debug_flags;
extern GType gtk_delete_type_get_type(void);
extern GType gtk_dest_defaults_get_type(void);
extern void gtk_dialog_add_action_widget(GtkDialog * dialog,
                                GtkWidget * child,
                                gint response_id);
extern GtkWidget *gtk_dialog_add_button(GtkDialog * dialog,
                                const gchar * button_text,
                                gint response_id);
extern void gtk_dialog_add_buttons(GtkDialog * dialog,
                                const gchar * first_button_text, ...);
extern GType gtk_dialog_flags_get_type(void);
extern gboolean gtk_dialog_get_has_separator(GtkDialog * dialog);
extern gint gtk_dialog_get_response_for_widget(GtkDialog * dialog,
                                GtkWidget * widget);
extern GType gtk_dialog_get_type(void);
extern GtkWidget *gtk_dialog_new(void);
extern GtkWidget *gtk_dialog_new_with_buttons(const gchar * title,
                                GtkWindow * parent,
                                GtkDialogFlags flags,
                                const gchar *
                                first_button_text, ...);
extern void gtk_dialog_response(GtkDialog * dialog, gint
response_id);
extern gint gtk_dialog_run(GtkDialog * dialog);
extern void gtk_dialog_set_alternative_button_order(GtkDialog *
dialog,
                                gint first_response_id,
                                ...);
extern void
gtk_dialog_set_alternative_button_order_from_array(GtkDialog *
dialog,
                                gint
                                n_params,
                                gint *
                                new_order);
extern void gtk_dialog_set_default_response(GtkDialog * dialog,
                                gint response_id);
extern void gtk_dialog_set_has_separator(GtkDialog * dialog,
                                gboolean setting);
extern void gtk_dialog_set_response_sensitive(GtkDialog * dialog,
                                gint response_id,
                                gboolean setting);

```

```

extern GType gtk_direction_type_get_type(void);
extern void gtk_disable_setlocale(void);
extern GdkDragContext *gtk_drag_begin(GtkWidget * widget,
                                     GtkTargetList * targets,
                                     GdkDragAction actions, gint button,
                                     GdkEvent * event);
extern gboolean gtk_drag_check_threshold(GtkWidget * widget, gint
start_x,
                                     gint start_y, gint current_x,
                                     gint current_y);
extern void gtk_drag_dest_add_image_targets(GtkWidget * widget);
extern void gtk_drag_dest_add_text_targets(GtkWidget * widget);
extern void gtk_drag_dest_add_uri_targets(GtkWidget * widget);
extern GdkAtom gtk_drag_dest_find_target(GtkWidget * widget,
                                     GdkDragContext * context,
                                     GtkTargetList * target_list);
extern GtkTargetList *gtk_drag_dest_get_target_list(GtkWidget *
widget);
extern gboolean gtk_drag_dest_get_track_motion(GtkWidget * widget);
extern void gtk_drag_dest_set(GtkWidget * widget, GtkDestDefaults
flags,
                                     const GtkTargetEntry * targets,
                                     gint n_targets, GdkDragAction actions);
extern void gtk_drag_dest_set_proxy(GtkWidget * widget,
                                     GdkWindow * proxy_window,
                                     GdkDragProtocol protocol,
                                     gboolean use_coordinates);
extern void gtk_drag_dest_set_target_list(GtkWidget * widget,
                                     GtkTargetList * target_list);
extern void gtk_drag_dest_set_track_motion(GtkWidget * widget,
                                     gboolean track_motion);
extern void gtk_drag_dest_unset(GtkWidget * widget);
extern void gtk_drag_finish(GdkDragContext * context, gboolean
success,
                                     gboolean del, guint32 time_);
extern void gtk_drag_get_data(GtkWidget * widget, GdkDragContext *
context,
                                     GdkAtom target, guint32 time_);
extern GtkWidget *gtk_drag_get_source_widget(GdkDragContext *
context);
extern void gtk_drag_highlight(GtkWidget * widget);
extern void gtk_drag_set_icon_default(GdkDragContext * context);
extern void gtk_drag_set_icon_name(GdkDragContext * context,
                                     const gchar * icon_name, gint hot_x,
                                     gint hot_y);
extern void gtk_drag_set_icon_pixbuf(GdkDragContext * context,
                                     GdkPixbuf * pixbuf, gint hot_x,
                                     gint hot_y);
extern void gtk_drag_set_icon_pixmap(GdkDragContext * context,
                                     GdkColormap * colormap,
                                     GdkPixmap * pixmap, GdkBitmap * mask,
                                     gint hot_x, gint hot_y);
extern void gtk_drag_set_icon_stock(GdkDragContext * context,
                                     const gchar * stock_id, gint hot_x,
                                     gint hot_y);
extern void gtk_drag_set_icon_widget(GdkDragContext * context,
                                     GtkWidget * widget, gint hot_x,
                                     gint hot_y);
extern void gtk_drag_source_add_image_targets(GtkWidget * widget);
extern void gtk_drag_source_add_text_targets(GtkWidget * widget);
extern void gtk_drag_source_add_uri_targets(GtkWidget * widget);
extern GtkTargetList *gtk_drag_source_get_target_list(GtkWidget *
widget);
extern void gtk_drag_source_set(GtkWidget * widget,
                                     GdkModifierType start_button_mask,
                                     const GtkTargetEntry * targets,

```

```

        gint n_targets, GdkDragAction actions);
extern void gtk_drag_source_set_icon(GtkWidget * widget,
        GdkColormap * colormap,
        GdkPixmap * pixmap, GdkBitmap *
mask);
extern void gtk_drag_source_set_icon_name(GtkWidget * widget,
        const gchar * icon_name);
extern void gtk_drag_source_set_icon_pixbuf(GtkWidget * widget,
        GdkPixbuf * pixbuf);
extern void gtk_drag_source_set_icon_stock(GtkWidget * widget,
        const gchar * stock_id);
extern void gtk_drag_source_set_target_list(GtkWidget * widget,
        GtkTargetList * target_list);
extern void gtk_drag_source_unset(GtkWidget * widget);
extern void gtk_drag_unhighlight(GtkWidget * widget);
extern void gtk_draw_insertion_cursor(GtkWidget * widget,
        GdkDrawable * drawable,
        const GdkRectangle * area,
        const GdkRectangle * location,
        gboolean is_primary,
        GtkTextDirection direction,
        gboolean draw_arrow);
extern GType gtk_drawing_area_get_type(void);
extern GtkWidget *gtk_drawing_area_new(void);
extern void gtk_editable_copy_clipboard(GtkEditable * editable);
extern void gtk_editable_cut_clipboard(GtkEditable * editable);
extern void gtk_editable_delete_selection(GtkEditable * editable);
extern void gtk_editable_delete_text(GtkEditable * editable,
        gint start_pos, gint end_pos);
extern gchar *gtk_editable_get_chars(GtkEditable * editable,
        gint start_pos, gint end_pos);
extern gboolean gtk_editable_get_editable(GtkEditable * editable);
extern gint gtk_editable_get_position(GtkEditable * editable);
extern gboolean gtk_editable_get_selection_bounds(GtkEditable *
editable,
        gint * start,
        gint * end);
extern GType gtk_editable_get_type(void);
extern void gtk_editable_insert_text(GtkEditable * editable,
        const gchar * new_text,
        gint new_text_length,
        gint * position);
extern void gtk_editable_paste_clipboard(GtkEditable * editable);
extern void gtk_editable_select_region(GtkEditable * editable, gint
start,
        gint end);
extern void gtk_editable_set_editable(GtkEditable * editable,
        gboolean is_editable);
extern void gtk_editable_set_position(GtkEditable * editable,
        gint position);
extern void gtk_entry_completion_complete(GtkEntryCompletion *
completion);
extern void gtk_entry_completion_delete_action(GtkEntryCompletion
*
        completion, gint index );
extern
        GtkWidget
*gtk_entry_completion_get_entry(GtkEntryCompletion *
        completion);
extern gboolean
gtk_entry_completion_get_inline_completion(GtkEntryCompletion *
        completion);
extern
        gint
gtk_entry_completion_get_minimum_key_length(GtkEntryCompletion
        * completion);
extern
        GtkTreeModel
*gtk_entry_completion_get_model(GtkEntryCompletion *

```

```

                                completion);

extern gboolean
gtk_entry_completion_get_popup_completion(GtkEntryCompletion *
completion);
extern
                                gboolean
gtk_entry_completion_get_popup_set_width(GtkEntryCompletion
                                * completion);

extern gboolean
gtk_entry_completion_get_popup_single_match(GtkEntryCompletion *
                                completion);

extern
                                gint
gtk_entry_completion_get_text_column(GtkEntryCompletion *
                                completion);
extern GType gtk_entry_completion_get_type(void);
extern
                                void
gtk_entry_completion_insert_action_markup(GtkEntryCompletion *
                                completion,
                                gint index_,
                                const gchar *
                                markup);

extern
                                void
gtk_entry_completion_insert_action_text(GtkEntryCompletion *
                                completion,
                                gint index_,
                                const gchar * text);
extern void gtk_entry_completion_insert_prefix(GtkEntryCompletion
*
                                completion);

extern GtkEntryCompletion *gtk_entry_completion_new(void);
extern
                                void
gtk_entry_completion_set_inline_completion(GtkEntryCompletion *
                                completion,
                                gboolean
                                inline_completion);
extern void gtk_entry_completion_set_match_func(GtkEntryCompletion
*
                                completion,

                                GtkEntryCompletionMatchFunc
                                func, gpointer func_data,
                                GDestroyNotify
                                func_notify);

extern
                                void
gtk_entry_completion_set_minimum_key_length(GtkEntryCompletion
                                * completion,
                                gint length);
extern void gtk_entry_completion_set_model(GtkEntryCompletion *
completion,
                                GtkTreeModel * model);
extern
                                void
gtk_entry_completion_set_popup_completion(GtkEntryCompletion *
                                completion,
                                gboolean
                                popup_completion);

extern
                                void
gtk_entry_completion_set_popup_set_width(GtkEntryCompletion *
                                completion,
                                gboolean
                                popup_set_width);

extern
                                void
gtk_entry_completion_set_popup_single_match(GtkEntryCompletion
                                * completion,
                                gboolean
                                popup_single_match);
extern
                                void
gtk_entry_completion_set_text_column(GtkEntryCompletion *

```



```

                                completion, gint column);
extern gboolean gtk_entry_get_activates_default(GtkEntry * entry);
extern gfloat gtk_entry_get_alignment(GtkEntry * entry);
extern GtkEntryCompletion *gtk_entry_get_completion(GtkEntry *
entry);
extern gboolean gtk_entry_get_has_frame(GtkEntry * entry);
extern const GtkBorder *gtk_entry_get_inner_border(GtkEntry *
entry);
extern gunichar gtk_entry_get_invisible_char(GtkEntry * entry);
extern PangoLayout *gtk_entry_get_layout(GtkEntry * entry);
extern void gtk_entry_get_layout_offsets(GtkEntry * entry, gint *
x,
                                gint * y);
extern gint gtk_entry_get_max_length(GtkEntry * entry);
extern const gchar *gtk_entry_get_text(GtkEntry * entry);
extern GType gtk_entry_get_type(void);
extern gboolean gtk_entry_get_visibility(GtkEntry * entry);
extern gint gtk_entry_get_width_chars(GtkEntry * entry);
extern gint gtk_entry_layout_index_to_text_index(GtkEntry * entry,
                                gint layout_index);
extern GtkWidget *gtk_entry_new(void);
extern void gtk_entry_set_activates_default(GtkEntry * entry,
                                gboolean setting);
extern void gtk_entry_set_alignment(GtkEntry * entry, gfloat
xalign);
extern void gtk_entry_set_completion(GtkEntry * entry,
                                GtkEntryCompletion * completion);
extern void gtk_entry_set_has_frame(GtkEntry * entry, gboolean
setting);
extern void gtk_entry_set_inner_border(GtkEntry * entry,
                                const GtkBorder * border);
extern void gtk_entry_set_invisible_char(GtkEntry * entry, gunichar
ch);
extern void gtk_entry_set_max_length(GtkEntry * entry, gint max);
extern void gtk_entry_set_text(GtkEntry * entry, const gchar *
text);
extern void gtk_entry_set_visibility(GtkEntry * entry, gboolean
visible);
extern void gtk_entry_set_width_chars(GtkEntry * entry, gint
n_chars);
extern gint gtk_entry_text_index_to_layout_index(GtkEntry * entry,
                                gint text_index);
extern gboolean gtk_event_box_get_above_child(GtkEventBox *
event_box);
extern GType gtk_event_box_get_type(void);
extern gboolean gtk_event_box_get_visible_window(GtkEventBox *
event_box);
extern GtkWidget *gtk_event_box_new(void);
extern void gtk_event_box_set_above_child(GtkEventBox * event_box,
                                gboolean above_child);
extern void gtk_event_box_set_visible_window(GtkEventBox *
event_box,
                                gboolean visible_window);
extern gboolean gtk_events_pending(void);
extern gboolean gtk_expander_get_expanded(GtkExpander * expander);
extern const char *gtk_expander_get_label(GtkExpander * expander);
extern GtkWidget *gtk_expander_get_label_widget(GtkExpander *
expander);
extern gint gtk_expander_get_spacing(GtkExpander * expander);
extern GType gtk_expander_get_type(void);
extern gboolean gtk_expander_get_use_markup(GtkExpander *
expander);
extern gboolean gtk_expander_get_use_underline(GtkExpander *
expander);
extern GtkWidget *gtk_expander_new(const gchar * label);

```

```

extern GtkWidget *gtk_expander_new_with_mnemonic(const gchar *
label);
extern void gtk_expander_set_expanded(GtkExpander * expander,
gboolean expanded);
extern void gtk_expander_set_label(GtkExpander * expander,
const gchar * label);
extern void gtk_expander_set_label_widget(GtkExpander * expander,
GtkWidget * label_widget);
extern void gtk_expander_set_spacing(GtkExpander * expander, gint
spacing);
extern void gtk_expander_set_use_markup(GtkExpander * expander,
gboolean use_markup);
extern void gtk_expander_set_use_underline(GtkExpander * expander,
gboolean use_underline);
extern GType gtk_expander_style_get_type(void);
extern gboolean gtk_false(void);
extern GType gtk_file_chooser_action_get_type(void);
extern void gtk_file_chooser_add_filter(GtkFileChooser * chooser,
GtkFileFilter * filter);

extern
gtk_file_chooser_add_shortcut_folder(GtkFileChooser *
chooser,
const char *folder,
GError * *error);

extern
gtk_file_chooser_add_shortcut_folder_uri(GtkFileChooser *
chooser,
const char *uri,
GError * *error);

extern gboolean
gtk_file_chooser_button_get_focus_on_click(GtkFileChooserButton *
button);
extern
const gchar
*gtk_file_chooser_button_get_title(GtkFileChooserButton
* button);
extern GType gtk_file_chooser_button_get_type(void);
extern
gint
gtk_file_chooser_button_get_width_chars(GtkFileChooserButton *
button);
extern GtkWidget *gtk_file_chooser_button_new(const gchar * title,
GtkFileChooserAction
action);
extern GtkWidget *gtk_file_chooser_button_new_with_backend(const
gchar *
title,
GtkFileChooserAction
action,
const gchar *
backend);
extern
GtkWidget
*gtk_file_chooser_button_new_with_dialog(GtkWidget *
dialog);

extern
void
gtk_file_chooser_button_set_focus_on_click(GtkFileChooserButton
* button,
gboolean
focus_on_click);
extern void gtk_file_chooser_button_set_title(GtkFileChooserButton
*
button, const gchar * title);
extern
void
gtk_file_chooser_button_set_width_chars(GtkFileChooserButton *
button, gint n_chars);
extern GType gtk_file_chooser_dialog_get_type(void);
extern GtkWidget *gtk_file_chooser_dialog_new(const gchar * title,

```

```

        GtkWidget * parent,
        GtkFileChooserAction action,
        const gchar *
        first_button_text, ...);
extern GtkWidget *gtk_file_chooser_dialog_new_with_backend(const
gchar *

        title,
        GtkWidget *
        parent,

GtkFileChooserAction

        action,
        const gchar *
        backend,
        const gchar *

first_button_text,

        ...);
extern GType gtk_file_chooser_error_get_type(void);
extern GQuark gtk_file_chooser_error_quark(void);
extern          GtkFileChooserAction
gtk_file_chooser_get_action(GtkFileChooser *
        chooser);
extern gchar *gtk_file_chooser_get_current_folder(GtkFileChooser *
        chooser);
extern          gchar
*gtk_file_chooser_get_current_folder_uri(GtkFileChooser *
        chooser);
extern gboolean
gtk_file_chooser_get_do_overwrite_confirmation(GtkFileChooser *
        chooser);
extern GtkWidget *gtk_file_chooser_get_extra_widget(GtkFileChooser
*
        chooser);
extern gchar *gtk_file_chooser_get_filename(GtkFileChooser *
        chooser);
extern GSList *gtk_file_chooser_get_filenames(GtkFileChooser *
        chooser);
extern GtkFileFilter *gtk_file_chooser_get_filter(GtkFileChooser *
        chooser);
extern gboolean gtk_file_chooser_get_local_only(GtkFileChooser *
        chooser);
extern char *gtk_file_chooser_get_preview_filename(GtkFileChooser
*
        chooser);
extern char *gtk_file_chooser_get_preview_uri(GtkFileChooser *
        chooser);
extern          GtkWidget
*gtk_file_chooser_get_preview_widget(GtkFileChooser *
        chooser);
extern          gboolean
gtk_file_chooser_get_preview_widget_active(GtkFileChooser *
        chooser);
extern          gboolean
gtk_file_chooser_get_select_multiple(GtkFileChooser *
        chooser);
extern gboolean gtk_file_chooser_get_show_hidden(GtkFileChooser *
        chooser);
extern GType gtk_file_chooser_get_type(void);
extern gchar *gtk_file_chooser_get_uri(GtkFileChooser * chooser);
extern GSList *gtk_file_chooser_get_uris(GtkFileChooser * chooser);
extern          gboolean
gtk_file_chooser_get_use_preview_label(GtkFileChooser *
        chooser);
extern GSList *gtk_file_chooser_list_filters(GtkFileChooser *
        chooser);

```

```

extern                                                                    GSList
*gtk_file_chooser_list_shortcut_folder_uris(GtkFileChooser *
                                              chooser);

extern                                                                    GSList
*gtk_file_chooser_list_shortcut_folders(GtkFileChooser *
                                         chooser);

extern void    gtk_file_chooser_remove_filter(GtkFileChooser *
chooser,
                                              GtkFileFilter * filter);

extern                                                                    gboolean
gtk_file_chooser_remove_shortcut_folder(GtkFileChooser *
                                         chooser,
                                         const char *folder,
                                         GError * *error);

extern                                                                    gboolean
gtk_file_chooser_remove_shortcut_folder_uri(GtkFileChooser
                                             * chooser,
                                             const char
                                             *uri,
                                             GError *
                                             *error);

extern void gtk_file_chooser_select_all(GtkFileChooser * chooser);
extern gboolean gtk_file_chooser_select_filename(GtkFileChooser *
chooser,
                                                  const gchar * filename);
extern gboolean  gtk_file_chooser_select_uri(GtkFileChooser *
chooser,
                                              const char *uri);
extern void gtk_file_chooser_set_action(GtkFileChooser * chooser,
                                       GtkFileChooserAction action);
extern gboolean gtk_file_chooser_set_current_folder(GtkFileChooser
*
                                              chooser,
                                              const gchar *
                                              filename);

extern                                                                    gboolean
gtk_file_chooser_set_current_folder_uri(GtkFileChooser *
                                         chooser,
                                         const gchar * uri);
extern void  gtk_file_chooser_set_current_name(GtkFileChooser *
chooser,
                                              const gchar * name);
extern                                                                    void
gtk_file_chooser_set_do_overwrite_confirmation(GtkFileChooser *
                                              chooser,
                                              gboolean
                                              do_overwrite_confirmation);
extern void  gtk_file_chooser_set_extra_widget(GtkFileChooser *
chooser,
                                              GtkWidget * extra_widget);
extern gboolean  gtk_file_chooser_set_filename(GtkFileChooser *
chooser,
                                              const gchar * filename);
extern void  gtk_file_chooser_set_filter(GtkFileChooser * chooser,
                                       GtkFileFilter * filter);
extern void  gtk_file_chooser_set_local_only(GtkFileChooser *
chooser,
                                              gboolean local_only);
extern void  gtk_file_chooser_set_preview_widget(GtkFileChooser *
chooser,
                                              GtkWidget *
                                              preview_widget);
extern                                                                    void
gtk_file_chooser_set_preview_widget_active(GtkFileChooser *
                                           chooser,

```

```

                                gboolean active);
extern void gtk_file_chooser_set_select_multiple(GtkFileChooser *
chooser,
                                gboolean select_multiple);
extern void gtk_file_chooser_set_show_hidden(GtkFileChooser *
chooser,
                                gboolean show_hidden);
extern gboolean gtk_file_chooser_set_uri(GtkFileChooser * chooser,
                                const char *uri);
extern void gtk_file_chooser_set_use_preview_label(GtkFileChooser
*
                                chooser,
                                gboolean use_label);
extern void gtk_file_chooser_unselect_all(GtkFileChooser *
chooser);
extern void gtk_file_chooser_unselect_filename(GtkFileChooser *
chooser,
                                const char *filename);
extern void gtk_file_chooser_unselect_uri(GtkFileChooser * chooser,
                                const char *uri);
extern GType gtk_file_chooser_widget_get_type(void);
extern GtkWidget *gtk_file_chooser_widget_new(GtkFileChooserAction
action);
extern GtkWidget
*gtk_file_chooser_widget_new_with_backend(GtkFileChooserAction
action,
                                const gchar * backend);
extern void gtk_file_filter_add_custom(GtkFileFilter * filter,
                                GtkFileFilterFlags needed,
                                GtkFileFilterFunc func,
                                gpointer data,
                                GDestroyNotify notify);
extern void gtk_file_filter_add_mime_type(GtkFileFilter * filter,
                                const gchar * mime_type);
extern void gtk_file_filter_add_pattern(GtkFileFilter * filter,
                                const gchar * pattern);
extern void gtk_file_filter_add_pixbuf_formats(GtkFileFilter *
filter);
extern gboolean gtk_file_filter_filter(GtkFileFilter * filter,
                                const GtkFileFilterInfo *
filter_info);
extern GType gtk_file_filter_flags_get_type(void);
extern const gchar *gtk_file_filter_get_name(GtkFileFilter *
filter);
extern GtkFileFilterFlags gtk_file_filter_get_needed(GtkFileFilter
*
                                filter);
extern GType gtk_file_filter_get_type(void);
extern GtkFileFilter *gtk_file_filter_new(void);
extern void gtk_file_filter_set_name(GtkFileFilter * filter,
                                const gchar * name);
extern void gtk_file_selection_complete(GtkFileSelection * filesel,
                                const gchar * pattern);
extern
                                const
                                gchar
*gtk_file_selection_get_filename(GtkFileSelection *
                                filesel);
extern
                                gboolean
gtk_file_selection_get_select_multiple(GtkFileSelection *
                                filesel);
extern gchar **gtk_file_selection_get_selections(GtkFileSelection
*
                                filesel);
extern GType gtk_file_selection_get_type(void);
extern
                                void
gtk_file_selection_hide_fileop_buttons(GtkFileSelection *
                                filesel);

```

```

extern GtkWidget *gtk_file_selection_new(const gchar * title);
extern void gtk_file_selection_set_filename(GtkFileSelection *
filesel,
                                const gchar * filename);

extern void
gtk_file_selection_set_select_multiple(GtkFileSelection *
                                filesel,
                                gboolean
                                select_multiple);

extern void
gtk_file_selection_show_fileop_buttons(GtkFileSelection *
                                filesel);

extern gboolean gtk_fixed_get_has_window(GtkFixed * fixed);
extern GType gtk_fixed_get_type(void);
extern void gtk_fixed_move(GtkFixed * fixed, GtkWidget * widget,
gint x,
                                gint y);
extern GtkWidget *gtk_fixed_new(void);
extern void gtk_fixed_put(GtkFixed * fixed, GtkWidget * widget,
gint x,
                                gint y);
extern void gtk_fixed_set_has_window(GtkFixed * fixed,
                                gboolean has_window);
extern const gchar *gtk_font_button_get_font_name(GtkFontButton *
font_button);
extern gboolean gtk_font_button_get_show_size(GtkFontButton *
font_button);
extern gboolean gtk_font_button_get_show_style(GtkFontButton *
font_button);
extern const gchar *gtk_font_button_get_title(GtkFontButton *
font_button);
extern GType gtk_font_button_get_type(void);
extern gboolean gtk_font_button_get_use_font(GtkFontButton *
font_button);
extern gboolean gtk_font_button_get_use_size(GtkFontButton *
font_button);
extern GtkWidget *gtk_font_button_new(void);
extern GtkWidget *gtk_font_button_new_with_font(const gchar *
fontname);
extern gboolean gtk_font_button_set_font_name(GtkFontButton *
font_button,
                                const gchar * fontname);
extern void gtk_font_button_set_show_size(GtkFontButton *
font_button,
                                gboolean show_size);
extern void gtk_font_button_set_show_style(GtkFontButton *
font_button,
                                gboolean show_style);
extern void gtk_font_button_set_title(GtkFontButton * font_button,
                                const gchar * title);
extern void gtk_font_button_set_use_font(GtkFontButton *
font_button,
                                gboolean use_font);
extern void gtk_font_button_set_use_size(GtkFontButton *
font_button,
                                gboolean use_size);
extern gchar

*gtk_font_selection_dialog_get_font_name(GtkFontSelectionDialog *
fsd);
extern const gchar

*gtk_font_selection_dialog_get_preview_text(GtkFontSelectionDialog *
fsd);
extern GType gtk_font_selection_dialog_get_type(void);

```

```

extern GtkWidget *gtk_font_selection_dialog_new(const gchar *
title);
extern gboolean
gtk_font_selection_dialog_set_font_name(GtkFontSelectionDialog *
fsd,
                                     const gchar * fontname);

extern void
gtk_font_selection_dialog_set_preview_text(GtkFontSelectionDialog
* fsd,
                                     const gchar * text);

extern gchar *gtk_font_selection_get_font_name(GtkFontSelection *
fontsel);
extern
                                     const
                                     gchar
*gtk_font_selection_get_preview_text(GtkFontSelection *
                                     fontsel);

extern GType gtk_font_selection_get_type(void);
extern GtkWidget *gtk_font_selection_new(void);
extern gboolean gtk_font_selection_set_font_name(GtkFontSelection
*
                                     fontsel,
                                     const gchar * fontname);
extern void gtk_font_selection_set_preview_text(GtkFontSelection *
fontsel,
                                     const gchar * text);

extern const gchar *gtk_frame_get_label(GtkFrame * frame);
extern void gtk_frame_get_label_align(GtkFrame * frame, gfloat *
xalign,
                                     gfloat * yalign);
extern GtkWidget *gtk_frame_get_label_widget(GtkFrame * frame);
extern GtkShadowType gtk_frame_get_shadow_type(GtkFrame * frame);
extern GType gtk_frame_get_type(void);
extern GtkWidget *gtk_frame_new(const gchar * label);
extern void gtk_frame_set_label(GtkFrame * frame, const gchar *
label);
extern void gtk_frame_set_label_align(GtkFrame * frame, gfloat
xalign,
                                     gfloat yalign);
extern void gtk_frame_set_label_widget(GtkFrame * frame,
                                     GtkWidget * label_widget);
extern void gtk_frame_set_shadow_type(GtkFrame * frame,
                                     GtkShadowType type);
extern GType gtk_gamma_curve_get_type(void);
extern GtkWidget *gtk_gamma_curve_new(void);
extern GdkGC *gtk_gc_get(gint depth, GdkColormap * colormap,
                                     GdkGCValues * values,
                                     GdkGCValuesMask values_mask);
extern void gtk_gc_release(GdkGC * gc);
extern GdkEvent *gtk_get_current_event(void);
extern gboolean gtk_get_current_event_state(GdkModifierType *
state);
extern guint32 gtk_get_current_event_time(void);
extern PangoLanguage *gtk_get_default_language(void);
extern GtkWidget *gtk_get_event_widget(GdkEvent * event);
extern
OptionGroup
*gtk_get_option_group(gboolean
open_default_display);
extern void gtk_grab_add(GtkWidget * widget);
extern GtkWidget *gtk_grab_get_current(void);
extern void gtk_grab_remove(GtkWidget * widget);
extern
                                     GtkPositionType
gtk_handle_box_get_handle_position(GtkHandleBox *
                                     handle_box);
extern GtkShadowType gtk_handle_box_get_shadow_type(GtkHandleBox *
                                     handle_box);
extern GtkPositionType gtk_handle_box_get_snap_edge(GtkHandleBox *
                                     handle_box);
extern GType gtk_handle_box_get_type(void);

```

```

extern GtkWidget *gtk_handle_box_new(void);
extern void gtk_handle_box_set_handle_position(GtkHandleBox *
handle_box,
                                           GtkPositionType position);
extern void gtk_handle_box_set_shadow_type(GtkHandleBox *
handle_box,
                                           GtkShadowType type);
extern void gtk_handle_box_set_snap_edge(GtkHandleBox * handle_box,
                                           GtkPositionType edge);
extern GType gtk_hbox_get_type(void);
extern GtkWidget *gtk_hbox_new(gboolean homogeneous, gint spacing);
extern GType gtk_hbutton_box_get_type(void);
extern GtkWidget *gtk_hbutton_box_new(void);
extern GType gtk_hpaned_get_type(void);
extern GtkWidget *gtk_hpaned_new(void);
extern GType gtk_hruler_get_type(void);
extern GtkWidget *gtk_hruler_new(void);
extern GType gtk_hscale_get_type(void);
extern GtkWidget *gtk_hscale_new(GtkAdjustment * adjustment);
extern GtkWidget *gtk_hscale_new_with_range(gdouble min, gdouble
max,
                                           gdouble step);
extern GType gtk_hscrollbar_get_type(void);
extern GtkWidget *gtk_hscrollbar_new(GtkAdjustment * adjustment);
extern GType gtk_hseparator_get_type(void);
extern GtkWidget *gtk_hseparator_new(void);
extern void gtk_icon_factory_add(GtkIconFactory * factory,
                                const gchar * stock_id,
                                GtkIconSet * icon_set);
extern void gtk_icon_factory_add_default(GtkIconFactory * factory);
extern GType gtk_icon_factory_get_type(void);
extern GtkIconSet *gtk_icon_factory_lookup(GtkIconFactory *
factory,
                                           const gchar * stock_id);
extern GtkIconSet *gtk_icon_factory_lookup_default(const gchar *
stock_id);
extern GtkIconFactory *gtk_icon_factory_new(void);
extern void gtk_icon_factory_remove_default(GtkIconFactory *
factory);
extern GtkIconInfo *gtk_icon_info_copy(GtkIconInfo * icon_info);
extern void gtk_icon_info_free(GtkIconInfo * icon_info);
extern gboolean gtk_icon_info_get_attach_points(GtkIconInfo *
icon_info,
                                           GdkPoint * *points,
                                           gint * n_points);
extern gint gtk_icon_info_get_base_size(GtkIconInfo * icon_info);
extern GdkPixbuf *gtk_icon_info_get_builtin_pixbuf(GtkIconInfo *
icon_info);
extern const gchar *gtk_icon_info_get_display_name(GtkIconInfo *
icon_info);
extern gboolean gtk_icon_info_get_embedded_rect(GtkIconInfo *
icon_info,
                                           GdkRectangle * rectangle);
extern const gchar *gtk_icon_info_get_filename(GtkIconInfo *
icon_info);
extern GType gtk_icon_info_get_type(void);
extern GdkPixbuf *gtk_icon_info_load_icon(GtkIconInfo * icon_info,
                                           GError * *error);
extern void gtk_icon_info_set_raw_coordinates(GtkIconInfo *
icon_info,
                                           gboolean raw_coordinates);
extern GType gtk_icon_lookup_flags_get_type(void);
extern void gtk_icon_set_add_source(GtkIconSet * icon_set,
                                const GtkIconSource * source);
extern GtkIconSet *gtk_icon_set_copy(GtkIconSet * icon_set);
extern void gtk_icon_set_get_sizes(GtkIconSet * icon_set,

```



```

        GtkIconSize * *sizes, gint * n_sizes);
extern GType gtk_icon_set_get_type(void);
extern GtkIconSet *gtk_icon_set_new(void);
extern GtkIconSet *gtk_icon_set_new_from_pixbuf(GdkPixbuf *
pixbuf);
extern GtkIconSet *gtk_icon_set_ref(GtkIconSet * icon_set);
extern GdkPixbuf *gtk_icon_set_render_icon(GtkIconSet * icon_set,
        GtkStyle * style,
        GtkTextDirection direction,
        GtkStateType state,
        GtkIconSize size,
        GtkWidget * widget,
        const char *detail);
extern void gtk_icon_set_unref(GtkIconSet * icon_set);
extern GtkIconSize gtk_icon_size_from_name(const gchar * name);
extern const gchar *gtk_icon_size_get_name(GtkIconSize size);
extern GType gtk_icon_size_get_type(void);
extern gboolean gtk_icon_size_lookup(GtkIconSize size, gint * width,
        gint * height);
extern gboolean gtk_icon_size_lookup_for_settings(GtkSettings *
settings,
        GtkIconSize size,
        gint * width,
        gint * height);
extern GtkIconSize gtk_icon_size_register(const gchar * name, gint
width,
        gint height);
extern void gtk_icon_size_register_alias(const gchar * alias,
        GtkIconSize target);
extern GtkIconSource *gtk_icon_source_copy(const GtkIconSource *
source);
extern void gtk_icon_source_free(GtkIconSource * source);
extern GtkTextDirection gtk_icon_source_get_direction(const
GtkIconSource *
        source);
extern gboolean gtk_icon_source_get_direction_wildcarded(const
        GtkIconSource *
        source);
extern const gchar *gtk_icon_source_get_filename(const
GtkIconSource *
        source);
extern const gchar *gtk_icon_source_get_icon_name(const
GtkIconSource *
        source);
extern GdkPixbuf *gtk_icon_source_get_pixbuf(const GtkIconSource *
source);
extern GtkIconSize gtk_icon_source_get_size(const GtkIconSource *
source);
extern gboolean gtk_icon_source_get_size_wildcarded(const
GtkIconSource *
        source);
extern GtkStateType gtk_icon_source_get_state(const GtkIconSource
*
        source);
extern gboolean gtk_icon_source_get_state_wildcarded(const
GtkIconSource *
        source);
extern GType gtk_icon_source_get_type(void);
extern GtkIconSource *gtk_icon_source_new(void);
extern void gtk_icon_source_set_direction(GtkIconSource * source,
        GtkTextDirection direction);
extern void gtk_icon_source_set_direction_wildcarded(GtkIconSource
*
        source,
        gboolean setting);
extern void gtk_icon_source_set_filename(GtkIconSource * source,

```

```

        const gchar * filename);
extern void gtk_icon_source_set_icon_name(GtkIconSource * source,
        const gchar * icon_name);
extern void gtk_icon_source_set_pixbuf(GtkIconSource * source,
        GdkPixbuf * pixbuf);
extern void gtk_icon_source_set_size(GtkIconSource * source,
        GtkIconSize size);
extern void gtk_icon_source_set_size_wildcarded(GtkIconSource *
source,
        gboolean setting);
extern void gtk_icon_source_set_state(GtkIconSource * source,
        GtkStateType state);
extern void gtk_icon_source_set_state_wildcarded(GtkIconSource *
source,
        gboolean setting);
extern void gtk_icon_theme_add_builtin_icon(const gchar * icon_name,
        gint size, GdkPixbuf * pixbuf);
extern void gtk_icon_theme_append_search_path(GtkIconTheme *
icon_theme,
        const gchar * path);
extern GType gtk_icon_theme_error_get_type(void);
extern GQuark gtk_icon_theme_error_quark(void);
extern GtkIconTheme *gtk_icon_theme_get_default(void);
extern char *gtk_icon_theme_get_example_icon_name(GtkIconTheme *
icon_theme);
extern GtkIconTheme *gtk_icon_theme_get_for_screen(GdkScreen *
screen);
extern gint *gtk_icon_theme_get_icon_sizes(GtkIconTheme *
icon_theme,
        const char *icon_name);
extern void gtk_icon_theme_get_search_path(GtkIconTheme *
icon_theme,
        gchar * **path,
        gint * n_elements);
extern GType gtk_icon_theme_get_type(void);
extern gboolean gtk_icon_theme_has_icon(GtkIconTheme * icon_theme,
        const char *icon_name);
extern GList *gtk_icon_theme_list_icons(GtkIconTheme * icon_theme,
        const char *context);
extern GdkPixbuf *gtk_icon_theme_load_icon(GtkIconTheme *
icon_theme,
        const gchar * icon_name,
        gint size,
        GtkIconLookupFlags flags,
        GError * *error);
extern GtkIconInfo *gtk_icon_theme_lookup_icon(GtkIconTheme *
icon_theme,
        const gchar * icon_name,
        gint size,
        GtkIconLookupFlags flags);
extern GtkIconTheme *gtk_icon_theme_new(void);
extern void gtk_icon_theme_prepend_search_path(GtkIconTheme *
icon_theme,
        const gchar * path);
extern gboolean gtk_icon_theme_rescan_if_needed(GtkIconTheme *
icon_theme);
extern void gtk_icon_theme_set_custom_theme(GtkIconTheme *
icon_theme,
        const gchar * theme_name);
extern void gtk_icon_theme_set_screen(GtkIconTheme * icon_theme,
        GdkScreen * screen);
extern void gtk_icon_theme_set_search_path(GtkIconTheme *
icon_theme,
        const gchar * *path,
        gint n_elements);

```

```

extern GdkPixmap *gtk_icon_view_create_drag_icon(GtkIconView *
icon_view,
                                           GtkTreePath * path);
extern void gtk_icon_view_enable_model_drag_dest(GtkIconView *
icon_view,
                                           const GtkTargetEntry *
targets, gint n_targets,
                                           GdkDragAction actions);
extern void gtk_icon_view_enable_model_drag_source(GtkIconView *
icon_view,
                                           GdkModifierType
start_button_mask,
                                           const GtkTargetEntry *
targets, gint n_targets,
                                           GdkDragAction actions);
extern gint gtk_icon_view_get_column_spacing(GtkIconView *
icon_view);
extern gint gtk_icon_view_get_columns(GtkIconView * icon_view);
extern gboolean gtk_icon_view_get_cursor(GtkIconView * icon_view,
                                           GtkTreePath * *path,
                                           GtkCellRenderer * *cell);
extern gboolean gtk_icon_view_get_dest_item_at_pos(GtkIconView *
icon_view,
                                           gint drag_x,
                                           gint drag_y,
                                           GtkTreePath * *path,
                                           GtkIconViewDropPosition
* pos);
extern void gtk_icon_view_get_drag_dest_item(GtkIconView *
icon_view,
                                           GtkTreePath * *path,
                                           GtkIconViewDropPosition *
pos);
extern gboolean gtk_icon_view_get_item_at_pos(GtkIconView *
icon_view,
                                           gint x, gint y,
                                           GtkTreePath * *path,
                                           GtkCellRenderer * *cell);
extern gint gtk_icon_view_get_item_width(GtkIconView * icon_view);
extern gint gtk_icon_view_get_margin(GtkIconView * icon_view);
extern gint gtk_icon_view_get_markup_column(GtkIconView *
icon_view);
extern GtkTreeModel *gtk_icon_view_get_model(GtkIconView *
icon_view);
extern GtkOrientation gtk_icon_view_get_orientation(GtkIconView *
icon_view);
extern GtkTreePath *gtk_icon_view_get_path_at_pos(GtkIconView *
icon_view,
                                           gint x, gint y);
extern gint gtk_icon_view_get_pixbuf_column(GtkIconView *
icon_view);
extern gboolean gtk_icon_view_get_reorderable(GtkIconView *
icon_view);
extern gint gtk_icon_view_get_row_spacing(GtkIconView * icon_view);
extern GList *gtk_icon_view_get_selected_items(GtkIconView *
icon_view);
extern GtkSelectionMode
gtk_icon_view_get_selection_mode(GtkIconView *
icon_view);
extern gint gtk_icon_view_get_spacing(GtkIconView * icon_view);
extern gint gtk_icon_view_get_text_column(GtkIconView * icon_view);
extern GType gtk_icon_view_get_type(void);
extern gboolean gtk_icon_view_get_visible_range(GtkIconView *
icon_view,
                                           GtkTreePath * *start_path,
                                           GtkTreePath * *end_path);

```

```

extern void gtk_icon_view_item_activated(GtkIconView * icon_view,
                                         GtkTreePath * path);
extern GtkWidget *gtk_icon_view_new(void);
extern GtkWidget *gtk_icon_view_new_with_model(GtkTreeModel *
model);
extern gboolean gtk_icon_view_path_is_selected(GtkIconView *
icon_view,
                                         GtkTreePath * path);
extern void gtk_icon_view_scroll_to_path(GtkIconView * icon_view,
                                         GtkTreePath * path,
                                         gboolean use_align,
                                         gfloat row_align,
                                         gfloat col_align);
extern void gtk_icon_view_select_all(GtkIconView * icon_view);
extern void gtk_icon_view_select_path(GtkIconView * icon_view,
                                       GtkTreePath * path);
extern void gtk_icon_view_selected_foreach(GtkIconView * icon_view,
                                           GtkIconViewForeachFunc func,
                                           gpointer data);
extern void gtk_icon_view_set_column_spacing(GtkIconView *
icon_view,
                                           gint column_spacing);
extern void gtk_icon_view_set_columns(GtkIconView * icon_view,
                                       gint columns);
extern void gtk_icon_view_set_cursor(GtkIconView * icon_view,
                                       GtkTreePath * path,
                                       GtkCellRenderer * cell,
                                       gboolean start_editing);
extern void gtk_icon_view_set_drag_dest_item(GtkIconView *
icon_view,
                                       GtkTreePath * path,
                                       GtkIconViewDropPosition pos);
extern void gtk_icon_view_set_item_width(GtkIconView * icon_view,
                                          gint item_width);
extern void gtk_icon_view_set_margin(GtkIconView * icon_view, gint
margin);
extern void gtk_icon_view_set_markup_column(GtkIconView *
icon_view,
                                          gint column);
extern void gtk_icon_view_set_model(GtkIconView * icon_view,
                                     GtkTreeModel * model);
extern void gtk_icon_view_set_orientation(GtkIconView * icon_view,
                                           GtkOrientation orientation);
extern void gtk_icon_view_set_pixbuf_column(GtkIconView *
icon_view,
                                           gint column);
extern void gtk_icon_view_set_reorderable(GtkIconView * icon_view,
                                           gboolean reorderable);
extern void gtk_icon_view_set_row_spacing(GtkIconView * icon_view,
                                           gint row_spacing);
extern void gtk_icon_view_set_selection_mode(GtkIconView *
icon_view,
                                           GtkSelectionMode mode);
extern void gtk_icon_view_set_spacing(GtkIconView * icon_view,
                                       gint spacing);
extern void gtk_icon_view_set_text_column(GtkIconView * icon_view,
                                           gint column);
extern void gtk_icon_view_deselect_all(GtkIconView * icon_view);
extern void gtk_icon_view_deselect_path(GtkIconView * icon_view,
                                          GtkTreePath * path);
extern void gtk_icon_view_unset_model_drag_dest(GtkIconView *
icon_view);
extern void gtk_icon_view_unset_model_drag_source(GtkIconView *
icon_view);
extern GType gtk_icon_view_get_type(void);

```

```

extern gboolean gtk_im_context_delete_surrounding(GtkIMContext *
context,
                                                    gint offset,
                                                    gint n_chars);
extern gboolean gtk_im_context_filter_keypress(GtkIMContext *
context,
                                                    GdkEventKey * event);
extern void gtk_im_context_focus_in(GtkIMContext * context);
extern void gtk_im_context_focus_out(GtkIMContext * context);
extern void gtk_im_context_get_preedit_string(GtkIMContext *
context,
                                                    gchar * *str,
                                                    PangoAttrList * *attrs,
                                                    gint * cursor_pos);
extern gboolean gtk_im_context_get_surrounding(GtkIMContext *
context,
                                                    gchar * *text,
                                                    gint * cursor_index);
extern GType gtk_im_context_get_type(void);
extern void gtk_im_context_reset(GtkIMContext * context);
extern void gtk_im_context_set_client_window(GtkIMContext *
context,
                                                    GdkWindow * window);
extern void gtk_im_context_set_cursor_location(GtkIMContext *
context,
                                                    const GdkRectangle * area);
extern void gtk_im_context_set_surrounding(GtkIMContext * context,
                                                    const gchar * text, gint len,
                                                    gint cursor_index);
extern void gtk_im_context_set_use_preedit(GtkIMContext * context,
                                                    gboolean use_preedit);
extern void gtk_im_context_simple_add_table(GtkIMContextSimple *
context_simple, guint16 *
data,
                                                    gint max_seq_len, gint
n_seqs);
extern GType gtk_im_context_simple_get_type(void);
extern GtkIMContext *gtk_im_context_simple_new(void);
extern void gtk_im_multicontext_append_menuitems(GtkIMMulticontext *
context,
                                                    GtkMenuShell * menushell);
extern GType gtk_im_multicontext_get_type(void);
extern GtkIMContext *gtk_im_multicontext_new(void);
extern GType gtk_im_preedit_style_get_type(void);
extern GType gtk_im_status_style_get_type(void);
extern void gtk_image_clear(GtkImage * image);
extern GdkPixbufAnimation *gtk_image_get_animation(GtkImage *
image);
extern void gtk_image_get_icon_name(GtkImage * image,
                                                    const gchar * *icon_name,
                                                    GtkIconSize * size);
extern void gtk_image_get_icon_set(GtkImage * image,
                                                    GtkIconSet * *icon_set,
                                                    GtkIconSize * size);
extern void gtk_image_get_image(GtkImage * image, GdkImage *
*gdk_image,
                                                    GdkBitmap * *mask);
extern GdkPixbuf *gtk_image_get_pixbuf(GtkImage * image);
extern gint gtk_image_get_pixel_size(GtkImage * image);
extern void gtk_image_get_pixmap(GtkImage * image, GdkPixmap *
*pixmap,
                                                    GdkBitmap * *mask);
extern void gtk_image_get_stock(GtkImage * image, gchar * *stock_id,
                                                    GtkIconSize * size);
extern GtkImageType gtk_image_get_storage_type(GtkImage * image);

```

```

extern GType gtk_image_get_type(void);
extern GtkWidget *gtk_image_menu_item_get_image(GtkImageMenuItem *
                                                image_menu_item);
extern GType gtk_image_menu_item_get_type(void);
extern GtkWidget *gtk_image_menu_item_new(void);
extern GtkWidget *gtk_image_menu_item_new_from_stock(const gchar *
                                                stock_id,
                                                GtkAccelGroup *
                                                accel_group);
extern GtkWidget *gtk_image_menu_item_new_with_label(const gchar *
label);
extern GtkWidget *gtk_image_menu_item_new_with_mnemonic(const
gchar *
                                                label);
extern void gtk_image_menu_item_set_image(GtkImageMenuItem *
image_menu_item,
GtkWidget * image);
extern GtkWidget *gtk_image_new(void);
extern GtkWidget *gtk_image_new_from_animation(GdkPixbufAnimation
*
animation);
extern GtkWidget *gtk_image_new_from_file(const gchar * filename);
extern GtkWidget *gtk_image_new_from_icon_name(const gchar *
icon_name,
                                                GtkIconSize size);
extern GtkWidget *gtk_image_new_from_icon_set(GtkIconSet *
icon_set,
                                                GtkIconSize size);
extern GtkWidget *gtk_image_new_from_image(GdkImage * image,
GdkBitmap * mask);
extern GtkWidget *gtk_image_new_from_pixbuf(GdkPixbuf * pixbuf);
extern GtkWidget *gtk_image_new_from_pixmap(GdkPixmap * pixmap,
GdkBitmap * mask);
extern GtkWidget *gtk_image_new_from_stock(const gchar * stock_id,
GtkIconSize size);
extern void gtk_image_set_from_animation(GtkImage * image,
GdkPixbufAnimation * animation);
extern void gtk_image_set_from_file(GtkImage * image,
const gchar * filename);
extern void gtk_image_set_from_icon_name(GtkImage * image,
const gchar * icon_name,
GtkIconSize size);
extern void gtk_image_set_from_icon_set(GtkImage * image,
GtkIconSet * icon_set,
GtkIconSize size);
extern void gtk_image_set_from_image(GtkImage * image,
GdkImage * gdk_image,
GdkBitmap * mask);
extern void gtk_image_set_from_pixbuf(GtkImage * image,
GdkPixbuf * pixbuf);
extern void gtk_image_set_from_pixmap(GtkImage * image, GdkPixmap
* pixmap,
GdkBitmap * mask);
extern void gtk_image_set_from_stock(GtkImage * image,
const gchar * stock_id,
GtkIconSize size);
extern void gtk_image_set_pixel_size(GtkImage * image, gint
pixel_size);
extern GType gtk_image_type_get_type(void);
extern void gtk_init(int *argc, char ***argv);
extern void gtk_init_add(GtkFunction function, gpointer data);
extern gboolean gtk_init_check(int *argc, char ***argv);
extern gboolean gtk_init_with_args(int *argc, char ***argv,
const char *parameter_string,
GOptionEntry * entries,
const char *translation_domain,

```

```

        GError * *error);
extern GType gtk_input_dialog_get_type(void);
extern GtkWidget *gtk_input_dialog_new(void);
extern const guint gtk_interface_age;
extern GdkScreen *gtk_invisible_get_screen(GtkInvisible *
invisible);
extern GType gtk_invisible_get_type(void);
extern GtkWidget *gtk_invisible_new(void);
extern GtkWidget *gtk_invisible_new_for_screen(GdkScreen * screen);
extern void gtk_invisible_set_screen(GtkInvisible * invisible,
GdkScreen * screen);
extern void gtk_item_deselect(GtkItem * item);
extern GType gtk_item_get_type(void);
extern void gtk_item_select(GtkItem * item);
extern void gtk_item_toggle(GtkItem * item);
extern GType gtk_justification_get_type(void);
extern guint gtk_key_snooper_install(GtkKeySnoopFunc snooper,
gpointer func_data);
extern void gtk_key_snooper_remove(guint snooper_handler_id);
extern gdouble gtk_label_get_angle(GtkLabel * label);
extern PangoAttrList *gtk_label_get_attributes(GtkLabel * label);
extern PangoEllipsizeMode gtk_label_get_ellipsize(GtkLabel *
label);
extern GtkJustification gtk_label_get_justify(GtkLabel * label);
extern const gchar *gtk_label_get_label(GtkLabel * label);
extern PangoLayout *gtk_label_get_layout(GtkLabel * label);
extern void gtk_label_get_layout_offsets(GtkLabel * label, gint *
x,
gint * y);
extern gboolean gtk_label_get_line_wrap(GtkLabel * label);
extern PangoWrapMode gtk_label_get_line_wrap_mode(GtkLabel *
label);
extern gint gtk_label_get_max_width_chars(GtkLabel * label);
extern guint gtk_label_get_mnemonic_keyval(GtkLabel * label);
extern GtkWidget *gtk_label_get_mnemonic_widget(GtkLabel * label);
extern gboolean gtk_label_get_selectable(GtkLabel * label);
extern gboolean gtk_label_get_selection_bounds(GtkLabel * label,
gint * start, gint * end);
extern gboolean gtk_label_get_single_line_mode(GtkLabel * label);
extern const gchar *gtk_label_get_text(GtkLabel * label);
extern GType gtk_label_get_type(void);
extern gboolean gtk_label_get_use_markup(GtkLabel * label);
extern gboolean gtk_label_get_use_underline(GtkLabel * label);
extern gint gtk_label_get_width_chars(GtkLabel * label);
extern GtkWidget *gtk_label_new(const gchar * str);
extern GtkWidget *gtk_label_new_with_mnemonic(const gchar * str);
extern void gtk_label_select_region(GtkLabel * label, gint
start_offset,
gint end_offset);
extern void gtk_label_set_angle(GtkLabel * label, gdouble angle);
extern void gtk_label_set_attributes(GtkLabel * label,
PangoAttrList * attrs);
extern void gtk_label_set_ellipsize(GtkLabel * label,
PangoEllipsizeMode mode);
extern void gtk_label_set_justify(GtkLabel * label,
GtkJustification jtype);
extern void gtk_label_set_label(GtkLabel * label, const gchar *
str);
extern void gtk_label_set_line_wrap(GtkLabel * label, gboolean
wrap);
extern void gtk_label_set_line_wrap_mode(GtkLabel * label,
PangoWrapMode wrap_mode);
extern void gtk_label_set_markup(GtkLabel * label, const gchar *
str);
extern void gtk_label_set_markup_with_mnemonic(GtkLabel * label,
const gchar * str);

```

```

extern void gtk_label_set_max_width_chars(GtkLabel * label, gint
n_chars);
extern void gtk_label_set_mnemonic_widget(GtkLabel * label,
                                         GtkWidget * widget);
extern void gtk_label_set_pattern(GtkLabel * label, const gchar *
pattern);
extern void gtk_label_set_selectable(GtkLabel * label, gboolean
setting);
extern void gtk_label_set_single_line_mode(GtkLabel * label,
                                           gboolean single_line_mode);
extern void gtk_label_set_text(GtkLabel * label, const gchar * str);
extern void gtk_label_set_text_with_mnemonic(GtkLabel * label,
                                           const gchar * str);
extern void gtk_label_set_use_markup(GtkLabel * label, gboolean
setting);
extern void gtk_label_set_use_underline(GtkLabel * label,
                                         gboolean setting);
extern void gtk_label_set_width_chars(GtkLabel * label, gint
n_chars);
extern GtkWidget *gtk_layout_get_hadjustment(GtkLayout *
layout);
extern void gtk_layout_get_size(GtkLayout * layout, guint * width,
                                guint * height);
extern GType gtk_layout_get_type(void);
extern GtkWidget *gtk_layout_get_vadjustment(GtkLayout *
layout);
extern void gtk_layout_move(GtkLayout * layout, GtkWidget *
child_widget,
                           gint x, gint y);
extern GtkWidget *gtk_layout_new(GtkAdjustment * hadjustment,
                                 GtkAdjustment * vadjustment);
extern void gtk_layout_put(GtkLayout * layout, GtkWidget *
child_widget,
                           gint x, gint y);
extern void gtk_layout_set_hadjustment(GtkLayout * layout,
                                       GtkAdjustment * adjustment);
extern void gtk_layout_set_size(GtkLayout * layout, guint width,
                                guint height);
extern void gtk_layout_set_vadjustment(GtkLayout * layout,
                                       GtkAdjustment * adjustment);
extern GType gtk_link_button_get_type(void);
extern const gchar *gtk_link_button_get_uri(GtkLinkButton *
link_button);
extern GtkWidget *gtk_link_button_new(const char *uri);
extern GtkWidget *gtk_link_button_new_with_label(const char *uri,
                                                  const char *label);
extern void gtk_link_button_set_uri(GtkLinkButton * link_button,
                                    const char *uri);
extern GtkLinkButtonUriFunc
gtk_link_button_set_uri_hook(GtkLinkButtonUriFunc func, gpointer
data,
                             GDestroyNotify destroy);
extern void gtk_list_store_append(GtkListStore * list_store,
                                  GtkTreeIter * iter);
extern void gtk_list_store_clear(GtkListStore * list_store);
extern GType gtk_list_store_get_type(void);
extern void gtk_list_store_insert(GtkListStore * list_store,
                                  GtkTreeIter * iter, gint position);
extern void gtk_list_store_insert_after(GtkListStore * list_store,
                                         GtkTreeIter * iter,
                                         GtkTreeIter * sibling);
extern void gtk_list_store_insert_before(GtkListStore * list_store,
                                         GtkTreeIter * iter,
                                         GtkTreeIter * sibling);
extern void gtk_list_store_insert_with_values(GtkListStore *
list_store,

```



```

        GtkTreeIter * iter,
        gint position, ...);
extern void gtk_list_store_insert_with_valuesv(GtkListStore *
list_store,

        GtkTreeIter * iter,
        gint position,
        gint * columns,
        GValue * values,
        gint n_values);
extern gboolean gtk_list_store_iter_is_valid(GtkListStore *
list_store,

        GtkTreeIter * iter);
extern void gtk_list_store_move_after(GtkListStore * store,
        GtkTreeIter * iter,
        GtkTreeIter * position);
extern void gtk_list_store_move_before(GtkListStore * store,
        GtkTreeIter * iter,
        GtkTreeIter * position);
extern GtkListStore *gtk_list_store_new(gint n_columns, ...);
extern GtkListStore *gtk_list_store_newv(gint n_columns, GType *
types);
extern void gtk_list_store_prepend(GtkListStore * list_store,
        GtkTreeIter * iter);
extern gboolean gtk_list_store_remove(GtkListStore * list_store,
        GtkTreeIter * iter);
extern void gtk_list_store_reorder(GtkListStore * store, gint *
new_order);
extern void gtk_list_store_set(GtkListStore * list_store,
        GtkTreeIter * iter, ...);
extern void gtk_list_store_set_column_types(GtkListStore *
list_store,

        gint n_columns, GType * types);
extern void gtk_list_store_set_valist(GtkListStore * list_store,
        GtkTreeIter * iter,
        va_list var_args);
extern void gtk_list_store_set_value(GtkListStore * list_store,
        GtkTreeIter * iter, gint column,
        GValue * value);
extern void gtk_list_store_swap(GtkListStore * store, GtkTreeIter
* a,

        GtkTreeIter * b);
extern void gtk_main(void);
extern void gtk_main_do_event(GdkEvent * event);
extern gboolean gtk_main_iteration(void);
extern gboolean gtk_main_iteration_do(gboolean blocking);
extern guint gtk_main_level(void);
extern void gtk_main_quit(void);
extern const guint gtk_major_version;
extern GType gtk_match_type_get_type(void);
extern void gtk_menu_attach(GtkMenu * menu, GtkWidget * child,
        guint left_attach, guint right_attach,
        guint top_attach, guint bottom_attach);
extern void gtk_menu_attach_to_widget(GtkMenu * menu,
        GtkWidget * attach_widget,
        GtkMenuDetachFunc detach);
extern
        GtkPackDirection
gtk_menu_bar_get_child_pack_direction(GtkMenuBar *
        menubar);
extern GtkPackDirection gtk_menu_bar_get_pack_direction(GtkMenuBar
*
        menubar);
extern GType gtk_menu_bar_get_type(void);
extern GtkWidget *gtk_menu_bar_new(void);
extern void gtk_menu_bar_set_child_pack_direction(GtkMenuBar *
menubar,

        GtkPackDirection

```

```

                                child_pack_dir);
extern void gtk_menu_bar_set_pack_direction(GtkMenuBar * menubar,
                                GtkPackDirection pack_dir);
extern void gtk_menu_detach(GtkMenu * menu);
extern GType gtk_menu_direction_type_get_type(void);
extern GtkAccelGroup *gtk_menu_get_accel_group(GtkMenu * menu);
extern GtkWidget *gtk_menu_get_active(GtkMenu * menu);
extern GtkWidget *gtk_menu_get_attach_widget(GtkMenu * menu);
extern GList *gtk_menu_get_for_attach_widget(GtkWidget * widget);
extern gboolean gtk_menu_get_tearoff_state(GtkMenu * menu);
extern const gchar *gtk_menu_get_title(GtkMenu * menu);
extern GType gtk_menu_get_type(void);
extern void gtk_menu_item_activate(GtkMenuItem * menu_item);
extern void gtk_menu_item_deselect(GtkMenuItem * menu_item);
extern gboolean gtk_menu_item_get_right_justified(GtkMenuItem *
menu_item);
extern GtkWidget *gtk_menu_item_get_submenu(GtkMenuItem *
menu_item);
extern GType gtk_menu_item_get_type(void);
extern GtkWidget *gtk_menu_item_new(void);
extern GtkWidget *gtk_menu_item_new_with_label(const gchar * label);
extern GtkWidget *gtk_menu_item_new_with_mnemonic(const gchar *
label);
extern void gtk_menu_item_remove_submenu(GtkMenuItem * menu_item);
extern void gtk_menu_item_select(GtkMenuItem * menu_item);
extern void gtk_menu_item_set_accel_path(GtkMenuItem * menu_item,
                                const gchar * accel_path);
extern void gtk_menu_item_set_right_justified(GtkMenuItem *
menu_item,
                                gboolean right_justified);
extern void gtk_menu_item_set_submenu(GtkMenuItem * menu_item,
                                GtkWidget * submenu);
extern void gtk_menu_item_toggle_size_allocate(GtkMenuItem *
menu_item,
                                gint allocation);
extern void gtk_menu_item_toggle_size_request(GtkMenuItem *
menu_item,
                                gint * requisition);
extern GtkWidget *gtk_menu_new(void);
extern void gtk_menu_popdown(GtkMenu * menu);
extern void gtk_menu_popup(GtkMenu * menu, GtkWidget *
parent_menu_shell,
                                GtkWidget * parent_menu_item,
                                GtkMenuPositionFunc func, gpointer data,
                                guint button, guint32 activate_time);
extern void gtk_menu_reorder_child(GtkMenu * menu, GtkWidget *
child,
                                gint position);
extern void gtk_menu_reposition(GtkMenu * menu);
extern void gtk_menu_set_accel_group(GtkMenu * menu,
                                GtkAccelGroup * accel_group);
extern void gtk_menu_set_accel_path(GtkMenu * menu,
                                const gchar * accel_path);
extern void gtk_menu_set_active(GtkMenu * menu, guint index);
extern void gtk_menu_set_monitor(GtkMenu * menu, gint monitor_num);
extern void gtk_menu_set_screen(GtkMenu * menu, GdkScreen * screen);
extern void gtk_menu_set_tearoff_state(GtkMenu * menu, gboolean
torn_off);
extern void gtk_menu_set_title(GtkMenu * menu, const gchar * title);
extern void gtk_menu_shell_activate_item(GtkMenuShell * menu_shell,
                                GtkWidget * menu_item,
                                gboolean force_deactivate);
extern void gtk_menu_shell_append(GtkMenuShell * menu_shell,
                                GtkWidget * child);
extern void gtk_menu_shell_cancel(GtkMenuShell * menu_shell);
extern void gtk_menu_shell_deactivate(GtkMenuShell * menu_shell);

```

```

extern void gtk_menu_shell_deselect(GtkMenuShell * menu_shell);
extern gboolean gtk_menu_shell_get_take_focus(GtkMenuShell *
menu_shell);
extern GType gtk_menu_shell_get_type(void);
extern void gtk_menu_shell_insert(GtkMenuShell * menu_shell,
                                GtkWidget * child, gint position);
extern void gtk_menu_shell_prepend(GtkMenuShell * menu_shell,
                                GtkWidget * child);
extern void gtk_menu_shell_select_first(GtkMenuShell * menu_shell,
                                gboolean search_sensitive);
extern void gtk_menu_shell_select_item(GtkMenuShell * menu_shell,
                                GtkWidget * menu_item);
extern void gtk_menu_shell_set_take_focus(GtkMenuShell *
menu_shell,
                                gboolean take_focus);
extern GtkWidget *gtk_menu_tool_button_get_menu(GtkMenuToolButton *
button);
extern GType gtk_menu_tool_button_get_type(void);
extern GtkToolItem *gtk_menu_tool_button_new(GtkWidget *
icon_widget,
                                const gchar * label);
extern GtkToolItem *gtk_menu_tool_button_new_from_stock(const
gchar *
                                stock_id);
extern void gtk_menu_tool_button_set_arrow_tooltip(GtkMenuToolButton *
button,
                                GtkTooltips * tooltips,
                                const gchar * tip_text,
                                const gchar *
                                tip_private);
extern void gtk_menu_tool_button_set_menu(GtkMenuToolButton *
button,
                                GtkWidget * menu);
extern void gtk_message_dialog_format_secondary_markup(GtkMessageDialog *
message_dialog,
                                const gchar *
                                message_format,
                                ...);
extern void gtk_message_dialog_format_secondary_text(GtkMessageDialog *
message_dialog,
                                const gchar *
                                message_format, ...);
extern GType gtk_message_dialog_get_type(void);
extern GtkWidget *gtk_message_dialog_new(GtkWindow * parent,
                                GtkDialogFlags flags,
                                GtkMessageType type,
                                GtkButtonsType buttons,
                                const gchar * message_format,
                                ...);
extern GtkWidget *gtk_message_dialog_new_with_markup(GtkWindow *
parent,
                                GtkDialogFlags flags,
                                GtkMessageType type,
                                GtkButtonsType
                                buttons,
                                const gchar *
                                message_format, ...);
extern void gtk_message_dialog_set_image(GtkMessageDialog * dialog,
                                GtkWidget * image);
extern void gtk_message_dialog_set_markup(GtkMessageDialog *
message_dialog,
                                const gchar * str);

```

```

extern GType gtk_message_type_get_type(void);
extern GType gtk_metric_type_get_type(void);
extern const guint gtk_micro_version;
extern const guint gtk_minor_version;
extern void gtk_misc_get_alignment(GtkMisc * misc, gfloat * xalign,
                                   gfloat * yalign);
extern void gtk_misc_get_padding(GtkMisc * misc, gint * xpad, gint
* ypad);
extern GType gtk_misc_get_type(void);
extern void gtk_misc_set_alignment(GtkMisc * misc, gfloat xalign,
                                   gfloat yalign);
extern void gtk_misc_set_padding(GtkMisc * misc, gint xpad, gint
ypad);
extern GType gtk_movement_step_get_type(void);
extern gint gtk_notebook_append_page(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     GtkWidget * tab_label);
extern gint gtk_notebook_append_page_menu(GtkNotebook * notebook,
                                           GtkWidget * child,
                                           GtkWidget * tab_label,
                                           GtkWidget * menu_label);
extern gint gtk_notebook_get_current_page(GtkNotebook * notebook);
extern GtkWidget *gtk_notebook_get_menu_label(GtkNotebook *
notebook,
                                              GtkWidget * child);
extern const gchar *gtk_notebook_get_menu_label_text(GtkNotebook *
notebook,
                                                      GtkWidget * child);
extern gint gtk_notebook_get_n_pages(GtkNotebook * notebook);
extern GtkWidget *gtk_notebook_get_nth_page(GtkNotebook * notebook,
                                             gint page_num);
extern gboolean gtk_notebook_get_scrollable(GtkNotebook *
notebook);
extern gboolean gtk_notebook_get_show_border(GtkNotebook *
notebook);
extern gboolean gtk_notebook_get_show_tabs(GtkNotebook * notebook);
extern gboolean gtk_notebook_get_tab_detachable(GtkNotebook *
notebook,
                                                GtkWidget * child);
extern GtkWidget *gtk_notebook_get_tab_label(GtkNotebook *
notebook,
                                              GtkWidget * child);
extern const gchar *gtk_notebook_get_tab_label_text(GtkNotebook *
notebook,
                                                      GtkWidget * child);
extern GtkPositionType gtk_notebook_get_tab_pos(GtkNotebook *
notebook);
extern gboolean gtk_notebook_get_tab_reorderable(GtkNotebook *
notebook,
                                                  GtkWidget * child);
extern GType gtk_notebook_get_type(void);
extern gint gtk_notebook_insert_page(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     GtkWidget * tab_label, gint
position);
extern gint gtk_notebook_insert_page_menu(GtkNotebook * notebook,
                                           GtkWidget * child,
                                           GtkWidget * tab_label,
                                           GtkWidget * menu_label,
                                           gint position);
extern GtkWidget *gtk_notebook_new(void);
extern void gtk_notebook_next_page(GtkNotebook * notebook);
extern gint gtk_notebook_page_num(GtkNotebook * notebook,
                                   GtkWidget * child);
extern void gtk_notebook_popup_disable(GtkNotebook * notebook);
extern void gtk_notebook_popup_enable(GtkNotebook * notebook);

```

```

extern gint gtk_notebook_prepend_page(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     GtkWidget * tab_label);
extern gint gtk_notebook_prepend_page_menu(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     GtkWidget * tab_label,
                                     GtkWidget * menu_label);
extern void gtk_notebook_prev_page(GtkNotebook * notebook);
extern void gtk_notebook_query_tab_label_packing(GtkNotebook *
notebook,
                                     GtkWidget * child,
                                     gboolean * expand,
                                     gboolean * fill,
                                     GtkPackType * pack_type);
extern void gtk_notebook_remove_page(GtkNotebook * notebook,
                                     gint page_num);
extern void gtk_notebook_reorder_child(GtkNotebook * notebook,
                                     GtkWidget * child, gint position);
extern void gtk_notebook_set_current_page(GtkNotebook * notebook,
                                     gint page_num);
extern void gtk_notebook_set_menu_label(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     GtkWidget * menu_label);
extern void gtk_notebook_set_menu_label_text(GtkNotebook *
notebook,
                                     GtkWidget * child,
                                     const gchar * menu_text);
extern void gtk_notebook_set_scrollable(GtkNotebook * notebook,
                                     gboolean scrollable);
extern void gtk_notebook_set_show_border(GtkNotebook * notebook,
                                     gboolean show_border);
extern void gtk_notebook_set_show_tabs(GtkNotebook * notebook,
                                     gboolean show_tabs);
extern void gtk_notebook_set_tab_detachable(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     gboolean detachable);
extern void gtk_notebook_set_tab_label(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     GtkWidget * tab_label);
extern void gtk_notebook_set_tab_label_packing(GtkNotebook *
notebook,
                                     GtkWidget * child,
                                     gboolean expand,
                                     gboolean fill,
                                     GtkPackType pack_type);
extern void gtk_notebook_set_tab_label_text(GtkNotebook * notebook,
                                     GtkWidget * child,
                                     const gchar * tab_text);
extern void gtk_notebook_set_tab_pos(GtkNotebook * notebook,
                                     GtkPositionType pos);
extern void gtk_notebook_set_tab_reorderable(GtkNotebook *
notebook,
                                     GtkWidget * child,
                                     gboolean reorderable);

extern void
gtk_notebook_set_window_creation_hook(GtkNotebookWindowCreationFu
nc func,
                                     gpointer data,
                                     GDestroyNotify destroy);

extern GType gtk_notebook_tab_get_type(void);
extern void gtk_object_destroy(GtkObject * object);
extern GType gtk_object_flags_get_type(void);
extern long unsigned int gtk_object_get_type(void);
extern void gtk_object_sink(GtkObject * object);
extern GType gtk_orientation_get_type(void);
extern GType gtk_pack_direction_get_type(void);

```

```

extern GType gtk_pack_type_get_type(void);
extern GType gtk_page_orientation_get_type(void);
extern GType gtk_page_set_get_type(void);
extern GtkWidget *gtk_page_setup_copy(GtkPageSetup * other);
extern gdouble gtk_page_setup_get_bottom_margin(GtkPageSetup *
setup,
                                         GtkUnit unit);
extern gdouble gtk_page_setup_get_left_margin(GtkPageSetup * setup,
                                         GtkUnit unit);
extern
                                         GtkPageOrientation
gtk_page_setup_get_orientation(GtkPageSetup *
                                         setup);
extern gdouble gtk_page_setup_get_page_height(GtkPageSetup * setup,
                                         GtkUnit unit);
extern gdouble gtk_page_setup_get_page_width(GtkPageSetup * setup,
                                         GtkUnit unit);
extern gdouble gtk_page_setup_get_paper_height(GtkPageSetup *
setup,
                                         GtkUnit unit);
extern GtkWidget *gtk_page_setup_get_paper_size(GtkPageSetup *
setup);
extern gdouble gtk_page_setup_get_paper_width(GtkPageSetup * setup,
                                         GtkUnit unit);
extern gdouble gtk_page_setup_get_right_margin(GtkPageSetup *
setup,
                                         GtkUnit unit);
extern gdouble gtk_page_setup_get_top_margin(GtkPageSetup * setup,
                                         GtkUnit unit);
extern GType gtk_page_setup_get_type(void);
extern GtkWidget *gtk_page_setup_new(void);
extern void gtk_page_setup_set_bottom_margin(GtkPageSetup * setup,
                                         gdouble margin, GtkUnit unit);
extern void gtk_page_setup_set_left_margin(GtkPageSetup * setup,
                                         gdouble margin, GtkUnit unit);
extern void gtk_page_setup_set_orientation(GtkPageSetup * setup,
                                         GtkPageOrientation
orientation);
extern void gtk_page_setup_set_paper_size(GtkPageSetup * setup,
                                         GtkWidget * size);
extern
                                         void
gtk_page_setup_set_paper_size_and_default_margins(GtkPageSetup
                                         * setup,
                                         GtkWidget
                                         * size);
extern void gtk_page_setup_set_right_margin(GtkPageSetup * setup,
                                         gdouble margin, GtkUnit unit);
extern void gtk_page_setup_set_top_margin(GtkPageSetup * setup,
                                         gdouble margin, GtkUnit unit);
extern void gtk_paint_arrow(GtkStyle * style, GdkWindow * window,
                                         GtkStateType state_type,
                                         GtkShadowType shadow_type,
                                         const GdkRectangle * area, GtkWidget *
widget,
                                         const gchar * detail, GtkArrowType
arrow_type,
                                         gboolean fill, gint x, gint y, gint width,
                                         gint height);
extern void gtk_paint_box(GtkStyle * style, GdkWindow * window,
                                         GtkStateType state_type,
                                         GtkShadowType shadow_type,
                                         const GdkRectangle * area, GtkWidget * widget,
                                         const gchar * detail, gint x, gint y, gint
width,
                                         gint height);
extern void gtk_paint_box_gap(GtkStyle * style, GdkWindow * window,
                                         GtkStateType state_type,

```

```

        GtkShadowType shadow_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar * detail,
        gint x, gint y, gint width, gint height,
        GtkPositionType gap_side, gint gap_x,
        gint gap_width);
extern void gtk_paint_check(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        gint width, gint height);
extern void gtk_paint_diamond(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar * detail,
        gint x, gint y, gint width, gint height);
extern void gtk_paint_expander(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar * detail,
        gint x, gint y,
        GtkExpanderStyle expander_style);
extern void gtk_paint_extension(GtkStyle * style, GdkWindow *
window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar * detail,
        gint x, gint y, gint width, gint height,
        GtkPositionType gap_side);
extern void gtk_paint_flat_box(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar * detail,
        gint x, gint y, gint width, gint height);
extern void gtk_paint_focus(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        gint width, gint height);
extern void gtk_paint_handle(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        gint width, gint height,
        GtkOrientation orientation);
extern void gtk_paint_hline(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x1, gint x2,
        gint y);
extern void gtk_paint_layout(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type, gboolean use_text,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        PangoLayout * layout);
extern void gtk_paint_option(GtkStyle * style, GdkWindow * window,

```

```

        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        gint width, gint height);
extern void gtk_paint_polygon(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar * detail,
        const GdkPoint * points, gint npoints,
        gboolean fill);
extern void gtk_paint_resize_grip(GtkStyle * style, GdkWindow *
window,
        GtkStateType state_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar *
detail,
        GdkWindowEdge edge, gint x, gint y,
        gint width, gint height);
extern void gtk_paint_shadow(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        gint width, gint height);
extern void gtk_paint_shadow_gap(GtkStyle * style, GdkWindow *
window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area,
        GtkWidget * widget, const gchar *
detail,
        gint x, gint y, gint width, gint height,
        GtkPositionType gap_side, gint gap_x,
        gint gap_width);
extern void gtk_paint_slider(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint x, gint y,
        gint width, gint height,
        GtkOrientation orientation);
extern void gtk_paint_tab(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        GtkShadowType shadow_type,
        const GdkRectangle * area, GtkWidget * widget,
        const gchar * detail, gint x, gint y, gint
width,
        gint height);
extern void gtk_paint_vline(GtkStyle * style, GdkWindow * window,
        GtkStateType state_type,
        const GdkRectangle * area, GtkWidget *
widget,
        const gchar * detail, gint y1_, gint y2_,
        gint x);
extern void gtk_paned_add1(GtkPaned * paned, GtkWidget * child);
extern void gtk_paned_add2(GtkPaned * paned, GtkWidget * child);
extern GtkWidget *gtk_paned_get_child1(GtkPaned * paned);
extern GtkWidget *gtk_paned_get_child2(GtkPaned * paned);
extern gint gtk_paned_get_position(GtkPaned * paned);
extern GType gtk_paned_get_type(void);
extern void gtk_paned_pack1(GtkPaned * paned, GtkWidget * child,

```



```

        gboolean resize, gboolean shrink);
extern void gtk_paned_pack2(GtkPaned * paned, GtkWidget * child,
        gboolean resize, gboolean shrink);
extern void gtk_paned_set_position(GtkPaned * paned, gint position);
extern GtkPaperSize *gtk_paper_size_copy(GtkPaperSize * other);
extern void gtk_paper_size_free(GtkPaperSize * size);
extern const char *gtk_paper_size_get_default(void);
extern
        gdouble
gtk_paper_size_get_default_bottom_margin(GtkPaperSize *
        size,
        GtkUnit unit);
extern gdouble gtk_paper_size_get_default_left_margin(GtkPaperSize
* size,
        GtkUnit unit);
extern
        gdouble
gtk_paper_size_get_default_right_margin(GtkPaperSize * size,
        GtkUnit unit);
extern gdouble gtk_paper_size_get_default_top_margin(GtkPaperSize
* size,
        GtkUnit unit);
extern const char *gtk_paper_size_get_display_name(GtkPaperSize *
size);
extern gdouble gtk_paper_size_get_height(GtkPaperSize * size,
        GtkUnit unit);
extern const char *gtk_paper_size_get_name(GtkPaperSize * size);
extern const char *gtk_paper_size_get_ppd_name(GtkPaperSize * size);
extern GType gtk_paper_size_get_type(void);
extern gdouble gtk_paper_size_get_width(GtkPaperSize * size,
        GtkUnit unit);
extern gboolean gtk_paper_size_is_custom(GtkPaperSize * size);
extern gboolean gtk_paper_size_is_equal(GtkPaperSize * size1,
        GtkPaperSize * size2);
extern GtkPaperSize *gtk_paper_size_new(const char *name);
extern GtkPaperSize *gtk_paper_size_new_custom(const char *name,
        const char *display_name,
        gdouble width,
        gdouble height,
        GtkUnit unit);
extern GtkPaperSize *gtk_paper_size_new_from_ppd(const char
*ppd_name,
        const char
        *ppd_display_name,
        gdouble width,
        gdouble height);
extern void gtk_paper_size_set_size(GtkPaperSize * size, gdouble
width,
        gdouble height, GtkUnit unit);
extern gboolean gtk_parse_args(int *argc, char ***argv);
extern GType gtk_path_priority_type_get_type(void);
extern GType gtk_path_type_get_type(void);
extern void gtk_plugin_construct(GtkPlug * plug, GdkNativeWindow
socket_id);
extern void gtk_plugin_construct_for_display(GtkPlug * plug,
        GdkDisplay * display,
        GdkNativeWindow socket_id);
extern GdkNativeWindow gtk_plugin_get_id(GtkPlug * plug);
extern GType gtk_plugin_get_type(void);
extern GtkWidget *gtk_plugin_new(GdkNativeWindow socket_id);
extern GtkWidget *gtk_plugin_new_for_display(GdkDisplay * display,
        GdkNativeWindow socket_id);
extern GType gtk_policy_type_get_type(void);
extern GType gtk_position_type_get_type(void);
extern
        PangoContext
*gtk_print_context_create_pango_context(GtkPrintContext
        * context);

```

```

extern PangoLayout
*gtk_print_context_create_pango_layout(GtkPrintContext *
context);

extern cairo_t
*gtk_print_context_get_cairo_context(GtkPrintContext *
context);

extern gdouble gtk_print_context_get_dpi_x(GtkPrintContext *
context);
extern gdouble gtk_print_context_get_dpi_y(GtkPrintContext *
context);
extern gdouble gtk_print_context_get_height(GtkPrintContext *
context);

extern GtkPageSetup
*gtk_print_context_get_page_setup(GtkPrintContext *
context);

extern PangoFontMap
*gtk_print_context_get_pango_fontmap(GtkPrintContext *
context);

extern GType gtk_print_context_get_type(void);
extern gdouble gtk_print_context_get_width(GtkPrintContext *
context);
extern void gtk_print_context_set_cairo_context(GtkPrintContext *
context,
cairo_t *cr, double dpi_x,
double dpi_y);

extern GType gtk_print_duplex_get_type(void);
extern GType gtk_print_error_get_type(void);
extern GQuark gtk_print_error_quark(void);
extern GType gtk_print_operation_action_get_type(void);
extern void gtk_print_operation_cancel(GtkPrintOperation * op);
extern GtkPageSetup
*gtk_print_operation_get_default_page_setup(GtkPrintOperation
* op);
extern void gtk_print_operation_get_error(GtkPrintOperation * op,
GError * *error);

extern GtkPrintSettings
*gtk_print_operation_get_print_settings(GtkPrintOperation *
op);

extern GtkPrintStatus
gtk_print_operation_get_status(GtkPrintOperation *
op);

extern const char
*gtk_print_operation_get_status_string(GtkPrintOperation
* op);

extern GType gtk_print_operation_get_type(void);
extern gboolean gtk_print_operation_is_finished(GtkPrintOperation
* op);
extern GtkPrintOperation *gtk_print_operation_new(void);
extern GType gtk_print_operation_result_get_type(void);
extern GtkPrintOperationResult
gtk_print_operation_run(GtkPrintOperation *
op,

GtkPrintOperationAction
action,
GtkWindow * parent,
GError * *error);

extern void gtk_print_operation_set_allow_async(GtkPrintOperation
* op,
gboolean allow_async);
extern void gtk_print_operation_set_current_page(GtkPrintOperation
* op,
gint current_page);
extern void
gtk_print_operation_set_custom_tab_label(GtkPrintOperation *
op,

```

```

        const char *label);
extern void
gtk_print_operation_set_default_page_setup(GtkPrintOperation *
        op,
        GtkPageSetup *
        default_page_setup);
extern void
gtk_print_operation_set_export_filename(GtkPrintOperation * op,
        const char *filename);
extern void gtk_print_operation_set_job_name(GtkPrintOperation *
op,
        const char *job_name);
extern void gtk_print_operation_set_n_pages(GtkPrintOperation * op,
        gint n_pages);
extern void
gtk_print_operation_set_print_settings(GtkPrintOperation * op,
        GtkPrintSettings *
        print_settings);
extern void
gtk_print_operation_set_show_progress(GtkPrintOperation * op,
        gboolean show_progress);
extern void
gtk_print_operation_set_track_print_status(GtkPrintOperation *
        op,
        gboolean
        track_status);
extern void gtk_print_operation_set_unit(GtkPrintOperation * op,
        GtkUnit unit);
extern void
gtk_print_operation_set_use_full_page(GtkPrintOperation * op,
        gboolean full_page);
extern GType gtk_print_pages_get_type(void);
extern GType gtk_print_quality_get_type(void);
extern GtkPageSetup *gtk_print_run_page_setup_dialog(GtkWindow *
parent,
        GtkPageSetup *
        page_setup,
        GtkPrintSettings *
        settings);
extern void gtk_print_run_page_setup_dialog_async(GtkWindow *
parent,
        GtkPageSetup *
        page_setup,
        GtkPrintSettings *
        settings,
        GtkPageSetupDoneFunc
        done_cb, gpointer data);
extern GtkPrintSettings *gtk_print_settings_copy(GtkPrintSettings
* other);
extern void gtk_print_settings_foreach(GtkPrintSettings * settings,
        GtkPrintSettingsFunc func,
        gpointer user_data);
extern const char *gtk_print_settings_get(GtkPrintSettings *
settings,
        const char *key);
extern gboolean gtk_print_settings_get_bool(GtkPrintSettings *
settings,
        const char *key);
extern gboolean gtk_print_settings_get_collate(GtkPrintSettings *
settings);
extern const char *
*gtk_print_settings_get_default_source(GtkPrintSettings *
        settings);
extern const char *gtk_print_settings_get_dither(GtkPrintSettings
*
        settings);

```

```

extern gdouble gtk_print_settings_get_double(GtkPrintSettings *
settings,
                                const char *key);
extern
                                gdouble
gtk_print_settings_get_double_with_default(GtkPrintSettings
                                * settings,
                                const char *key,
                                gdouble def);
extern
                                GtkPrintDuplex
gtk_print_settings_get_duplex(GtkPrintSettings *
                                settings);
extern
                                const
                                char
*gtk_print_settings_get_finishings(GtkPrintSettings *
                                settings);
extern gint gtk_print_settings_get_int(GtkPrintSettings * settings,
                                const char *key);
extern
                                gint
gtk_print_settings_get_int_with_default(GtkPrintSettings *
                                settings,
                                const char *key,
                                gint def);
extern gdouble gtk_print_settings_get_length(GtkPrintSettings *
settings,
                                const char *key,
                                GtkUnit unit);
extern
                                const
                                char
*gtk_print_settings_get_media_type(GtkPrintSettings *
                                settings);
extern gint gtk_print_settings_get_n_copies(GtkPrintSettings *
settings);
extern gint gtk_print_settings_get_number_up(GtkPrintSettings *
settings);
extern GtkPageOrientation
gtk_print_settings_get_orientation(GtkPrintSettings * settings);
extern
                                const
                                char
*gtk_print_settings_get_output_bin(GtkPrintSettings *
                                settings);
extern
                                GtkPageRange
*gtk_print_settings_get_page_ranges(GtkPrintSettings *
                                settings,
                                gint * num_ranges);
extern GtkPageSet gtk_print_settings_get_page_set(GtkPrintSettings
*
                                settings);
extern
                                gdouble
gtk_print_settings_get_paper_height(GtkPrintSettings *
                                settings, GtkUnit unit);
extern
                                GtkPaperSize
*gtk_print_settings_get_paper_size(GtkPrintSettings *
                                settings);
extern gdouble gtk_print_settings_get_paper_width(GtkPrintSettings
*
                                settings, GtkUnit unit);
extern
                                GtkPrintPages
gtk_print_settings_get_print_pages(GtkPrintSettings *
                                settings);
extern const char *gtk_print_settings_get_printer(GtkPrintSettings
*
                                settings);
extern
                                GtkPrintQuality
gtk_print_settings_get_quality(GtkPrintSettings *
                                settings);
extern gint gtk_print_settings_get_resolution(GtkPrintSettings *
settings);
extern gboolean gtk_print_settings_get_reverse(GtkPrintSettings *
                                settings);

```

```

extern gdouble gtk_print_settings_get_scale(GtkPrintSettings *
settings);
extern GType gtk_print_settings_get_type(void);
extern gboolean gtk_print_settings_get_use_color(GtkPrintSettings
*
settings);
extern gboolean gtk_print_settings_has_key(GtkPrintSettings *
settings,
const char *key);
extern GtkPrintSettings *gtk_print_settings_new(void);
extern void gtk_print_settings_set(GtkPrintSettings * settings,
const char *key, const char *value);
extern void gtk_print_settings_set_bool(GtkPrintSettings *
settings,
const char *key, gboolean value);
extern void gtk_print_settings_set_collate(GtkPrintSettings *
settings,
gboolean collate);
extern void gtk_print_settings_set_default_source(GtkPrintSettings
*
settings,
const char
*default_source);
extern void gtk_print_settings_set_dither(GtkPrintSettings *
settings,
const char *dither);
extern void gtk_print_settings_set_double(GtkPrintSettings *
settings,
const char *key, gdouble value);
extern void gtk_print_settings_set_duplex(GtkPrintSettings *
settings,
GtkPrintDuplex duplex);
extern void gtk_print_settings_set_finishings(GtkPrintSettings *
settings,
const char *finishings);
extern void gtk_print_settings_set_int(GtkPrintSettings * settings,
const char *key, gint value);
extern void gtk_print_settings_set_length(GtkPrintSettings *
settings,
const char *key, gdouble value,
GtkUnit unit);
extern void gtk_print_settings_set_media_type(GtkPrintSettings *
settings,
const char *media_type);
extern void gtk_print_settings_set_n_copies(GtkPrintSettings *
settings,
gint num_copies);
extern void gtk_print_settings_set_number_up(GtkPrintSettings *
settings,
gint number_up);
extern void gtk_print_settings_set_orientation(GtkPrintSettings *
settings,
GtkPageOrientation
orientation);
extern void gtk_print_settings_set_output_bin(GtkPrintSettings *
settings,
const char *output_bin);
extern void gtk_print_settings_set_page_ranges(GtkPrintSettings *
settings,
GtkPageRange * page_ranges,
gint num_ranges);
extern void gtk_print_settings_set_page_set(GtkPrintSettings *
settings,
GtkPageSet page_set);
extern void gtk_print_settings_set_paper_height(GtkPrintSettings *
settings, gdouble height,

```

```

                                GtkUnit unit);
extern void gtk_print_settings_set_paper_size(GtkPrintSettings *
settings,
                                GtkPaperSize * paper_size);
extern void gtk_print_settings_set_paper_width(GtkPrintSettings *
settings,
                                gdouble width,
                                GtkUnit unit);
extern void gtk_print_settings_set_print_pages(GtkPrintSettings *
settings,
                                GtkPrintPages pages);
extern void gtk_print_settings_set_printer(GtkPrintSettings *
settings,
                                const char *printer);
extern void gtk_print_settings_set_quality(GtkPrintSettings *
settings,
                                GtkPrintQuality quality);
extern void gtk_print_settings_set_resolution(GtkPrintSettings *
settings,
                                gint resolution);
extern void gtk_print_settings_set_reverse(GtkPrintSettings *
settings,
                                gboolean reverse);
extern void gtk_print_settings_set_scale(GtkPrintSettings *
settings,
                                gdouble scale);
extern void gtk_print_settings_set_use_color(GtkPrintSettings *
settings,
                                gboolean use_color);
extern void gtk_print_settings_unset(GtkPrintSettings * settings,
                                const char *key);
extern GType gtk_print_status_get_type(void);
extern PangoEllipsizeMode
gtk_progress_bar_get_ellipsize(GtkProgressBar *
                                pbar);
extern gdouble gtk_progress_bar_get_fraction(GtkProgressBar *
pbar);
extern GtkProgressBarOrientation
gtk_progress_bar_get_orientation(GtkProgressBar * pbar);
extern gdouble gtk_progress_bar_get_pulse_step(GtkProgressBar *
pbar);
extern const gchar *gtk_progress_bar_get_text(GtkProgressBar *
pbar);
extern GType gtk_progress_bar_get_type(void);
extern GtkWidget *gtk_progress_bar_new(void);
extern GType gtk_progress_bar_orientation_get_type(void);
extern void gtk_progress_bar_pulse(GtkProgressBar * pbar);
extern void gtk_progress_bar_set_ellipsize(GtkProgressBar * pbar,
                                PangoEllipsizeMode mode);
extern void gtk_progress_bar_set_fraction(GtkProgressBar * pbar,
                                gdouble fraction);
extern void gtk_progress_bar_set_orientation(GtkProgressBar * pbar,
                                GtkProgressBarOrientation
                                orientation);
extern void gtk_progress_bar_set_pulse_step(GtkProgressBar * pbar,
                                gdouble fraction);
extern void gtk_progress_bar_set_text(GtkProgressBar * pbar,
                                const gchar * text);
extern GType gtk_progress_bar_style_get_type(void);
extern void gtk_propagate_event(GtkWidget * widget, GdkEvent *
event);
extern guint gtk_quit_add(guint main_level, GtkFunction function,
                                gpointer data);
extern void gtk_quit_add_destroy(guint main_level, GObject *
object);

```

```

extern guint gtk_quit_add_full(guint main_level, GtkFunction
function,
                                GtkCallbackMarshal marshal, gpointer
data,
                                GtkDestroyNotify destroy);
extern void gtk_quit_remove(guint quit_handler_id);
extern void gtk_quit_remove_by_data(gpointer data);
extern gint gtk_radio_action_get_current_value(GtkRadioAction *
action);
extern GSList *gtk_radio_action_get_group(GtkRadioAction * action);
extern GType gtk_radio_action_get_type(void);
extern GtkRadioAction *gtk_radio_action_new(const gchar * name,
                                             const gchar * label,
                                             const gchar * tooltip,
                                             const gchar * stock_id,
                                             gint value);
extern void gtk_radio_action_set_current_value(GtkRadioAction *
action,
                                             gint current_value);
extern void gtk_radio_action_set_group(GtkRadioAction * action,
                                       GSList * group);
extern GSList *gtk_radio_button_get_group(GtkRadioButton *
radio_button);
extern GType gtk_radio_button_get_type(void);
extern GtkWidget *gtk_radio_button_new(GSList * group);
extern GtkWidget *gtk_radio_button_new_from_widget(GtkRadioButton
*
radio_group_member);
extern GtkWidget *gtk_radio_button_new_with_label(GSList * group,
                                                  const gchar * label);
extern GtkWidget
*gtk_radio_button_new_with_label_from_widget(GtkRadioButton *
radio_group_member,
const gchar * label);
extern GtkWidget *gtk_radio_button_new_with_mnemonic(GSList *
group,
                                                  const gchar * label);
extern GtkWidget
*gtk_radio_button_new_with_mnemonic_from_widget(GtkRadioButton
*
radio_group_member,
const gchar * label);
extern void gtk_radio_button_set_group(GtkRadioButton *
radio_button,
                                       GSList * group);
extern GSList *gtk_radio_menu_item_get_group(GtkRadioMenuItem *
radio_menu_item);
extern GType gtk_radio_menu_item_get_type(void);
extern GtkWidget *gtk_radio_menu_item_new(GSList * group);
extern GtkWidget
*gtk_radio_menu_item_new_from_widget(GtkRadioMenuItem *
group);
extern GtkWidget *gtk_radio_menu_item_new_with_label(GSList *
group,
                                                  const gchar * label);
extern GtkWidget
*gtk_radio_menu_item_new_with_label_from_widget(GtkRadioMenuItem *
group,
const gchar * label);
extern GtkWidget *gtk_radio_menu_item_new_with_mnemonic(GSList *
group,
                                                  const gchar *
label);
extern GtkWidget

```

```

*gtk_radio_menu_item_new_with_mnemonic_from_widget(GtkRadioMenuItem *
em *
                                group,
                                const gchar *
                                label);
extern void gtk_radio_menu_item_set_group(GtkRadioMenuItem *
                                radio_menu_item, GSList *
                                group);
extern GSList *gtk_radio_tool_button_get_group(GtkRadioToolButton *
button);
extern GType gtk_radio_tool_button_get_type(void);
extern GtkWidget *gtk_radio_tool_button_new(GSList * group);
extern GtkWidget *gtk_radio_tool_button_new_from_stock(GSList *
group,
                                const gchar *
                                stock_id);
extern GtkWidget *
    *gtk_radio_tool_button_new_from_widget(GtkRadioToolButton *
group);
extern GtkWidget *

*gtk_radio_tool_button_new_with_stock_from_widget(GtkRadioToolBut
ton *
                                group,
                                const gchar *
                                stock_id);
extern void gtk_radio_tool_button_set_group(GtkRadioToolButton *
button,
                                GSList * group);
extern GtkWidget *gtk_range_get_adjustment(GtkRange * range);
extern gboolean gtk_range_get_inverted(GtkRange * range);
extern GtkWidget *gtk_range_get_lower_stepper_sensitivity(GtkRange
* range);
extern GType gtk_range_get_type(void);
extern GtkWidget *gtk_range_get_update_policy(GtkRange * range);
extern GtkWidget *
    *gtk_range_get_upper_stepper_sensitivity(GtkRange
* range);
extern gdouble gtk_range_get_value(GtkRange * range);
extern void gtk_range_set_adjustment(GtkRange * range,
    GtkWidget * adjustment);
extern void gtk_range_set_increments(GtkRange * range, gdouble step,
    gdouble page);
extern void gtk_range_set_inverted(GtkRange * range, gboolean
setting);
extern void gtk_range_set_lower_stepper_sensitivity(GtkRange *
range,
    GtkWidget *
    GtkWidget *
    sensitivity);
extern void gtk_range_set_range(GtkRange * range, gdouble min,
    gdouble max);
extern void gtk_range_set_update_policy(GtkRange * range,
    GtkWidget *
    policy);
extern void gtk_range_set_upper_stepper_sensitivity(GtkRange *
range,
    GtkWidget *
    GtkWidget *
    sensitivity);
extern void gtk_range_set_value(GtkRange * range, gdouble value);
extern void gtk_rc_add_default_file(const gchar * filename);
extern gchar *gtk_rc_find_module_in_path(const gchar * module_file);
extern gchar *gtk_rc_find_pixmap_in_path(GtkSettings * settings,
    GtkWidget * scanner,
    const gchar * pixmap_file);

```



```

extern GType gtk_rc_flags_get_type(void);
extern gchar **gtk_rc_get_default_files(void);
extern gchar *gtk_rc_get_im_module_file(void);
extern gchar *gtk_rc_get_im_module_path(void);
extern gchar *gtk_rc_get_module_dir(void);
extern GtkStyle *gtk_rc_get_style(GtkWidget * widget);
extern GtkStyle *gtk_rc_get_style_by_paths(GtkSettings * settings,
                                           const char *widget_path,
                                           const char *class_path,
                                           GType type);

extern gchar *gtk_rc_get_theme_dir(void);
extern void gtk_rc_parse(const gchar * filename);
extern guint gtk_rc_parse_color(GScanner * scanner, GdkColor *
color);
extern guint gtk_rc_parse_priority(GScanner * scanner,
                                   GtkPathPriorityType * priority);
extern guint gtk_rc_parse_state(GScanner * scanner, GtkStateType *
state);
extern void gtk_rc_parse_string(const gchar * rc_string);
extern gboolean gtk_rc_property_parse_border(const GParamSpec *
pspec,
                                           const GString * gstring,
                                           GValue * property_value);
extern gboolean gtk_rc_property_parse_color(const GParamSpec *
pspec,
                                           const GString * gstring,
                                           GValue * property_value);
extern gboolean gtk_rc_property_parse_enum(const GParamSpec * pspec,
                                           const GString * gstring,
                                           GValue * property_value);
extern gboolean gtk_rc_property_parse_flags(const GParamSpec *
pspec,
                                           const GString * gstring,
                                           GValue * property_value);
extern gboolean gtk_rc_property_parse_requisition(const GParamSpec
* pspec,
                                           const GString * gstring,
                                           GValue * property_value);
extern gboolean gtk_rc_reparse_all(void);
extern gboolean gtk_rc_reparse_all_for_settings(GtkSettings *
settings,
                                           gboolean force_load);
extern void gtk_rc_reset_styles(GtkSettings * settings);
extern GScanner *gtk_rc_scanner_new(void);
extern void gtk_rc_set_default_files(gchar * *filenames);
extern GtkRcStyle *gtk_rc_style_copy(GtkRcStyle * orig);
extern GType gtk_rc_style_get_type(void);
extern GtkRcStyle *gtk_rc_style_new(void);
extern void gtk_rc_style_ref(GtkRcStyle * rc_style);
extern void gtk_rc_style_unref(GtkRcStyle * rc_style);
extern GType gtk_rc_token_type_get_type(void);
extern void gtk_recent_chooser_add_filter(GtkRecentChooser *
chooser,
                                           GtkRecentFilter * filter);
extern GType gtk_recent_chooser_dialog_get_type(void);
extern GtkWidget *gtk_recent_chooser_dialog_new(const char *title,
                                           GtkWindow * parent,
                                           const char
                                           *first_button_text, ...);
extern GtkWidget *gtk_recent_chooser_dialog_new_for_manager(const
char
                                           *title,
                                           GtkWindow *
                                           parent,
                                           GtkRecentManager

```

```

        * manager,
        const char

*first_button_text,
        ...);
extern GType gtk_recent_chooser_error_get_type(void);
extern GQuark gtk_recent_chooser_error_quark(void);
extern          GtkRecentInfo
*gtk_recent_chooser_get_current_item(GtkRecentChooser
        * chooser);
extern gchar *gtk_recent_chooser_get_current_uri(GtkRecentChooser
*
        chooser);
extern          GtkRecentFilter
*gtk_recent_chooser_get_filter(GtkRecentChooser *
        chooser);
extern GList *gtk_recent_chooser_get_items(GtkRecentChooser *
chooser);
extern gint gtk_recent_chooser_get_limit(GtkRecentChooser *
chooser);
extern gboolean gtk_recent_chooser_get_local_only(GtkRecentChooser
*
        chooser);
extern          gboolean
gtk_recent_chooser_get_select_multiple(GtkRecentChooser *
        chooser);
extern gboolean gtk_recent_chooser_get_show_icons(GtkRecentChooser
*
        chooser);
extern          gboolean
gtk_recent_chooser_get_show_not_found(GtkRecentChooser *
        chooser);
extern          gboolean
gtk_recent_chooser_get_show_private(GtkRecentChooser *
        chooser);
extern gboolean gtk_recent_chooser_get_show_tips(GtkRecentChooser
*
        chooser);
extern          GtkRecentSortType
gtk_recent_chooser_get_sort_type(GtkRecentChooser
        * chooser);
extern GType gtk_recent_chooser_get_type(void);
extern gchar **gtk_recent_chooser_get_uris(GtkRecentChooser *
chooser,
        gsize * length);
extern GSList *gtk_recent_chooser_list_filters(GtkRecentChooser *
chooser);
extern gboolean
gtk_recent_chooser_menu_get_show_numbers(GtkRecentChooserMenu *
menu);
extern GType gtk_recent_chooser_menu_get_type(void);
extern GtkWidget *gtk_recent_chooser_menu_new(void);
extern          GtkWidget
*gtk_recent_chooser_menu_new_for_manager(GtkRecentManager
        * manager);
extern          void
gtk_recent_chooser_menu_set_show_numbers(GtkRecentChooserMenu *
        menu,
        gboolean
        show_numbers);
extern void gtk_recent_chooser_remove_filter(GtkRecentChooser *
chooser,
        GtkRecentFilter * filter);
extern void gtk_recent_chooser_select_all(GtkRecentChooser *
chooser);

```

```

extern gboolean gtk_recent_chooser_select_uri(GtkRecentChooser *
chooser,
const char *uri,
GError * *error);

extern gboolean
gtk_recent_chooser_set_current_uri(GtkRecentChooser *
chooser,
const char *uri,
GError * *error);

extern void gtk_recent_chooser_set_filter(GtkRecentChooser *
chooser,
GtkRecentFilter * filter);

extern void gtk_recent_chooser_set_limit(GtkRecentChooser *
chooser,
gint limit);

extern void gtk_recent_chooser_set_local_only(GtkRecentChooser *
chooser,
gboolean local_only);

extern void
gtk_recent_chooser_set_select_multiple(GtkRecentChooser *
chooser,
gboolean
select_multiple);

extern void gtk_recent_chooser_set_show_icons(GtkRecentChooser *
chooser,
gboolean show_icons);

extern void gtk_recent_chooser_set_show_not_found(GtkRecentChooser *
chooser,
gboolean show_not_found);

extern void gtk_recent_chooser_set_show_private(GtkRecentChooser *
chooser,
gboolean show_private);

extern void gtk_recent_chooser_set_show_tips(GtkRecentChooser *
chooser,
gboolean show_tips);

extern void gtk_recent_chooser_set_sort_func(GtkRecentChooser *
chooser,
GtkRecentSortFunc sort_func,
gpointer sort_data,
GDestroyNotify data_destroy);

extern void gtk_recent_chooser_set_sort_type(GtkRecentChooser *
chooser,
GtkRecentSortType sort_type);

extern void gtk_recent_chooser_unselect_all(GtkRecentChooser *
chooser);

extern void gtk_recent_chooser_unselect_uri(GtkRecentChooser *
chooser,
const char *uri);

extern GType gtk_recent_chooser_widget_get_type(void);
extern GtkWidget *gtk_recent_chooser_widget_new(void);
extern GtkWidget
*gtk_recent_chooser_widget_new_for_manager(GtkRecentManager *
manager);

extern GType gtk_recent_filter_flags_get_type(void);
extern GType gtk_recent_manager_error_get_type(void);
extern GType gtk_recent_sort_type_get_type(void);
extern GType gtk_relief_style_get_type(void);
extern GtkRequisition *gtk_requisition_copy(const GtkRequisition *
requisition);

extern void gtk_requisition_free(GtkRequisition * requisition);
extern GType gtk_requisition_get_type(void);
extern GType gtk_resize_mode_get_type(void);
extern GType gtk_response_type_get_type(void);
extern void gtk_ruler_draw_pos(GtkRuler * ruler);
extern void gtk_ruler_draw_ticks(GtkRuler * ruler);

```

```

extern GtkMetricType gtk_ruler_get_metric(GtkRuler * ruler);
extern void gtk_ruler_get_range(GtkRuler * ruler, gdouble * lower,
                                gdouble * upper, gdouble * position,
                                gdouble * max_size);
extern GType gtk_ruler_get_type(void);
extern void gtk_ruler_set_metric(GtkRuler * ruler, GtkMetricType
metric);
extern void gtk_ruler_set_range(GtkRuler * ruler, gdouble lower,
                                gdouble upper, gdouble position,
                                gdouble max_size);
extern gint gtk_scale_get_digits(GtkScale * scale);
extern gboolean gtk_scale_get_draw_value(GtkScale * scale);
extern PangoLayout *gtk_scale_get_layout(GtkScale * scale);
extern void gtk_scale_get_layout_offsets(GtkScale * scale, gint *
x,
                                gint * y);
extern GType gtk_scale_get_type(void);
extern GtkPositionType gtk_scale_get_value_pos(GtkScale * scale);
extern void gtk_scale_set_digits(GtkScale * scale, gint digits);
extern void gtk_scale_set_draw_value(GtkScale * scale,
                                gboolean draw_value);
extern void gtk_scale_set_value_pos(GtkScale * scale,
GtkPositionType pos);
extern GType gtk_scroll_step_get_type(void);
extern GType gtk_scroll_type_get_type(void);
extern GType gtk_scrollbar_get_type(void);
extern void gtk_scrolled_window_add_with_viewport(GtkScrolledWindow *
scrolled_window,
                                GtkWidget * child);
extern GtkAdjustment *gtk_scrolled_window_get_hadjustment(GtkScrolledWindow
*
                                scrolled_window);
extern GtkWidget *gtk_scrolled_window_get_hscrollbar(GtkScrolledWindow *
scrolled_window);
extern GtkCornerType gtk_scrolled_window_get_placement(GtkScrolledWindow *
scrolled_window);
extern void gtk_scrolled_window_get_policy(GtkScrolledWindow *
scrolled_window,
                                GtkPolicyType *
                                hscrollbar_policy,
                                GtkPolicyType *
                                vscrollbar_policy);
extern GtkShadowType gtk_scrolled_window_get_shadow_type(GtkScrolledWindow
*
                                scrolled_window);
extern GType gtk_scrolled_window_get_type(void);
extern GtkAdjustment *gtk_scrolled_window_get_vadjustment(GtkScrolledWindow
*
                                scrolled_window);
extern GtkWidget *gtk_scrolled_window_get_vscrollbar(GtkScrolledWindow *
scrolled_window);
extern GtkWidget *gtk_scrolled_window_new(GtkAdjustment *
hadjustment,
                                GtkAdjustment * vadjustment);
extern void gtk_scrolled_window_set_hadjustment(GtkScrolledWindow
*
                                scrolled_window,
                                GtkAdjustment *
                                hadjustment);

```

```

extern void gtk_scrolled_window_set_placement(GtkScrolledWindow *
                                             scrolled_window,
                                             GtkCornerType
                                             window_placement);
extern void gtk_scrolled_window_set_policy(GtkScrolledWindow *
                                             scrolled_window,
                                             GtkPolicyType
                                             hscrollbar_policy,
                                             GtkPolicyType
                                             vscrollbar_policy);
extern void gtk_scrolled_window_set_shadow_type(GtkScrolledWindow
*
                                             scrolled_window,
                                             GtkShadowType type);
extern void gtk_scrolled_window_set_vadjustment(GtkScrolledWindow
*
                                             scrolled_window,
                                             GtkAdjustment *
                                             vadjustment);
extern void gtk_scrolled_window_unset_placement(GtkScrolledWindow
*
                                             scrolled_window);
extern void gtk_selection_add_target(GtkWidget * widget, GdkAtom
selection,
                                             GdkAtom target, quint info);
extern void gtk_selection_add_targets(GtkWidget * widget,
                                     GdkAtom selection,
                                     const GtkTargetEntry * targets,
                                     quint ntargets);
extern void gtk_selection_clear_targets(GtkWidget * widget,
                                       GdkAtom selection);
extern gboolean gtk_selection_convert(GtkWidget * widget,
                                     GdkAtom selection, GdkAtom target,
                                     quint32 time_);
extern GtkSelectionData *gtk_selection_data_copy(GtkSelectionData
* data);
extern void gtk_selection_data_free(GtkSelectionData * data);
extern GdkPixbuf *gtk_selection_data_get_pixbuf(GtkSelectionData *
selection_data);
extern gboolean gtk_selection_data_get_targets(GtkSelectionData *
selection_data,
                                             GdkAtom **targets,
                                             gint * n_atoms);
extern gchar *gtk_selection_data_get_text(GtkSelectionData *
selection_data);
extern GType gtk_selection_data_get_type(void);
extern gchar **gtk_selection_data_get_uris(GtkSelectionData *
selection_data);
extern void gtk_selection_data_set(GtkSelectionData *
selection_data,
                                   GdkAtom type, gint format,
                                   const gchar * data, gint length);
extern gboolean gtk_selection_data_set_pixbuf(GtkSelectionData *
selection_data,
                                             GdkPixbuf * pixbuf);
extern gboolean gtk_selection_data_set_text(GtkSelectionData *
selection_data,
                                             const gchar * str, gint len);
extern gboolean gtk_selection_data_set_uris(GtkSelectionData *
selection_data,
                                             gchar *
*uris);
extern gboolean
gtk_selection_data_targets_include_image(GtkSelectionData *
selection_data,
gboolean
writable);

```

```

extern gboolean
gtk_selection_data_targets_include_rich_text(GtkSelectionData *
                                             selection_data,
                                             GtkTextBuffer * buffer);

extern gboolean
gtk_selection_data_targets_include_text(GtkSelectionData *
                                         selection_data);

extern gboolean
gtk_selection_data_targets_include_uri(GtkSelectionData *
                                       selection_data);

extern GType gtk_selection_mode_get_type(void);
extern gboolean gtk_selection_owner_set(GtkWidget * widget,
                                         GdkAtom selection, guint32 time_);
extern gboolean gtk_selection_owner_set_for_display(GdkDisplay *
                                                    display,
                                                    GtkWidget * widget,
                                                    GdkAtom selection,
                                                    guint32 time_);

extern void gtk_selection_remove_all(GtkWidget * widget);
extern GType gtk_sensitivity_type_get_type(void);
extern GType gtk_separator_get_type(void);
extern GType gtk_separator_menu_item_get_type(void);
extern GtkWidget *gtk_separator_menu_item_new(void);
extern gboolean
gtk_separator_tool_item_get_draw(GtkSeparatorToolItem *
                                  item);

extern GType gtk_separator_tool_item_get_type(void);
extern GtkToolItem *gtk_separator_tool_item_new(void);
extern void gtk_separator_tool_item_set_draw(GtkSeparatorToolItem
* item,
                                             gboolean draw);

extern gchar *gtk_set_locale(void);
extern GtkSettings *gtk_settings_get_default(void);
extern GtkSettings *gtk_settings_get_for_screen(GdkScreen *
screen);
extern GType gtk_settings_get_type(void);
extern void gtk_settings_install_property(GParamSpec * pspec);
extern void gtk_settings_install_property_parser(GParamSpec *
pspec,
                                                  GtkRcPropertyParser
                                                  parser);

extern void gtk_settings_set_double_property(GtkSettings *
settings,
                                             const gchar * name,
                                             gdouble v_double,
                                             const gchar * origin);
extern void gtk_settings_set_long_property(GtkSettings * settings,
                                           const gchar * name,
                                           glong v_long,
                                           const gchar * origin);
extern void gtk_settings_set_property_value(GtkSettings * settings,
                                             const gchar * name,
                                             const GtkSettingsValue *
svalue);
extern void gtk_settings_set_string_property(GtkSettings *
settings,
                                             const gchar * name,
                                             const gchar * v_string,
                                             const gchar * origin);

extern GType gtk_shadow_type_get_type(void);
extern void gtk_show_about_dialog(GtkWindow * parent,
                                  const gchar *
                                  first_property_name, ...);
extern GType gtk_side_type_get_type(void);
extern GType gtk_signal_run_type_get_type(void);
extern void gtk_size_group_add_widget(GtkSizeGroup * size_group,

```

```

        GtkWidget * widget);
extern gboolean gtk_size_group_get_ignore_hidden(GtkSizeGroup *
        size_group);
extern GtkSizeGroupMode gtk_size_group_get_mode(GtkSizeGroup *
size_group);
extern GType gtk_size_group_get_type(void);
extern GSList *gtk_size_group_get_widgets(GtkSizeGroup *
size_group);
extern GType gtk_size_group_mode_get_type(void);
extern GtkSizeGroup *gtk_size_group_new(GtkSizeGroupMode mode);
extern void gtk_size_group_remove_widget(GtkSizeGroup * size_group,
        GtkWidget * widget);
extern void gtk_size_group_set_ignore_hidden(GtkSizeGroup *
size_group,
        gboolean ignore_hidden);
extern void gtk_size_group_set_mode(GtkSizeGroup * size_group,
        GtkSizeGroupMode mode);
extern void gtk_socket_add_id(GtkSocket * socket_,
        GdkNativeWindow window_id);
extern GdkNativeWindow gtk_socket_get_id(GtkSocket * socket_);
extern GType gtk_socket_get_type(void);
extern GtkWidget *gtk_socket_new(void);
extern GType gtk_sort_type_get_type(void);
extern void gtk_spin_button_configure(GtkSpinButton * spin_button,
        GtkAdjustment * adjustment,
        gdouble climb_rate, guint digits);
extern GtkAdjustment *gtk_spin_button_get_adjustment(GtkSpinButton
*
        spin_button);
extern guint gtk_spin_button_get_digits(GtkSpinButton *
spin_button);
extern void gtk_spin_button_get_increments(GtkSpinButton *
spin_button,
        gdouble * step, gdouble * page);
extern gboolean gtk_spin_button_get_numeric(GtkSpinButton *
spin_button);
extern void gtk_spin_button_get_range(GtkSpinButton * spin_button,
        gdouble * min, gdouble * max);
extern gboolean gtk_spin_button_get_snap_to_ticks(GtkSpinButton *
spin_button);
extern GType gtk_spin_button_get_type(void);
extern GtkSpinButtonUpdatePolicy
gtk_spin_button_get_update_policy(GtkSpinButton * spin_button);
extern gdouble gtk_spin_button_get_value(GtkSpinButton *
spin_button);
extern gint gtk_spin_button_get_value_as_int(GtkSpinButton *
spin_button);
extern gboolean gtk_spin_button_get_wrap(GtkSpinButton *
spin_button);
extern GtkWidget *gtk_spin_button_new(GtkAdjustment * adjustment,
        gdouble climb_rate, guint digits);
extern GtkWidget *gtk_spin_button_new_with_range(gdouble min,
        gdouble max,
        gdouble step);
extern void gtk_spin_button_set_adjustment(GtkSpinButton *
spin_button,
        GtkAdjustment * adjustment);
extern void gtk_spin_button_set_digits(GtkSpinButton * spin_button,
        guint digits);
extern void gtk_spin_button_set_increments(GtkSpinButton *
spin_button,
        gdouble step, gdouble page);
extern void gtk_spin_button_set_numeric(GtkSpinButton *
spin_button,
        gboolean numeric);
extern void gtk_spin_button_set_range(GtkSpinButton * spin_button,

```

```

                                gdouble min, gdouble max);
extern void gtk_spin_button_set_snap_to_ticks(GtkSpinButton *
spin_button,
                                gboolean snap_to_ticks);
extern void gtk_spin_button_set_update_policy(GtkSpinButton *
spin_button,
                                GtkSpinButtonUpdatePolicy
                                policy);
extern void gtk_spin_button_set_value(GtkSpinButton * spin_button,
                                gdouble value);
extern void gtk_spin_button_set_wrap(GtkSpinButton * spin_button,
                                gboolean wrap);
extern void gtk_spin_button_spin(GtkSpinButton * spin_button,
                                GtkSpinType direction, gdouble
increment);
extern void gtk_spin_button_update(GtkSpinButton * spin_button);
extern GType gtk_spin_button_update_policy_get_type(void);
extern GType gtk_spin_type_get_type(void);
extern GType gtk_state_type_get_type(void);
extern guint gtk_statusbar_get_context_id(GtkStatusbar * statusbar,
                                const gchar *
                                context_description);
extern gboolean gtk_statusbar_get_has_resize_grip(GtkStatusbar *
statusbar,
                                statusbar);
extern GType gtk_statusbar_get_type(void);
extern GtkWidget *gtk_statusbar_new(void);
extern void gtk_statusbar_pop(GtkStatusbar * statusbar, guint
context_id);
extern guint gtk_statusbar_push(GtkStatusbar * statusbar, guint
context_id,
                                const gchar * text);
extern void gtk_statusbar_remove(GtkStatusbar * statusbar,
                                guint context_id, guint message_id);
extern void gtk_statusbar_set_has_resize_grip(GtkStatusbar *
statusbar,
                                gboolean setting);
extern void gtk_stock_add(const GtkStockItem * items, guint
n_items);
extern void gtk_stock_add_static(const GtkStockItem * items,
                                guint n_items);
extern GtkStockItem *gtk_stock_item_copy(const GtkStockItem * item);
extern void gtk_stock_item_free(GtkStockItem * item);
extern GSList *gtk_stock_list_ids(void);
extern gboolean gtk_stock_lookup(const gchar * stock_id,
                                GtkStockItem * item);
extern void gtk_stock_set_translate_func(const gchar * domain,
                                GtkTranslateFunc func,
                                gpointer data,
                                GtkDestroyNotify notify);
extern void gtk_style_apply_default_background(GtkStyle * style,
                                GdkWindow * window,
                                gboolean set_bg,
                                GtkStateType state_type,
                                const GdkRectangle * area,
                                gint x, gint y, gint width,
                                gint height);
extern GtkStyle *gtk_style_attach(GtkStyle * style, GdkWindow *
window);
extern GtkStyle *gtk_style_copy(GtkStyle * style);
extern void gtk_style_detach(GtkStyle * style);
extern GType gtk_style_get_type(void);
extern gboolean gtk_style_lookup_color(GtkStyle * style,
                                const gchar * color_name,
                                GdkColor * color);
extern GtkIconSet *gtk_style_lookup_icon_set(GtkStyle * style,
                                const char *stock_id);

```



```

extern GtkStyle *gtk_style_new(void);
extern GdkPixbuf *gtk_style_render_icon(GtkStyle * style,
                                         const GtkIconSource * source,
                                         GtkTextDirection direction,
                                         GtkStateType state,
                                         GtkIconSize size,
                                         GtkWidget * widget,
                                         const gchar * detail);
extern void gtk_style_set_background(GtkStyle * style, GdkWindow *
window,
                                   GtkStateType state_type);
extern GType gtk_submenu_direction_get_type(void);
extern GType gtk_submenu_placement_get_type(void);
extern void gtk_table_attach(GtkTable * table, GtkWidget * child,
                             guint left_attach, guint right_attach,
                             guint top_attach, guint bottom_attach,
                             GtkAttachOptions xoptions,
                             GtkAttachOptions yoptions, guint xpadding,
                             guint ypadding);
extern void gtk_table_attach_defaults(GtkTable * table, GtkWidget
* widget,
                                     guint left_attach,
                                     guint right_attach,
                                     guint
top_attach,
                                     guint bottom_attach);
extern guint gtk_table_get_col_spacing(GtkTable * table, guint
column);
extern guint gtk_table_get_default_col_spacing(GtkTable * table);
extern guint gtk_table_get_default_row_spacing(GtkTable * table);
extern gboolean gtk_table_get_homogeneous(GtkTable * table);
extern guint gtk_table_get_row_spacing(GtkTable * table, guint row);
extern GType gtk_table_get_type(void);
extern GtkWidget *gtk_table_new(guint rows, guint columns,
                                gboolean homogeneous);
extern void gtk_table_resize(GtkTable * table, guint rows, guint
columns);
extern void gtk_table_set_col_spacing(GtkTable * table, guint
column,
                                     guint spacing);
extern void gtk_table_set_col_spacings(GtkTable * table, guint
spacing);
extern void gtk_table_set_homogeneous(GtkTable * table,
                                       gboolean homogeneous);
extern void gtk_table_set_row_spacing(GtkTable * table, guint row,
                                       guint spacing);
extern void gtk_table_set_row_spacings(GtkTable * table, guint
spacing);
extern GType gtk_target_flags_get_type(void);
extern void gtk_target_list_add(GtkTargetList * list, GdkAtom
target,
                                guint flags, guint info);
extern void gtk_target_list_add_image_targets(GtkTargetList * list,
                                              guint info,
                                              gboolean writable);
extern void gtk_target_list_add_rich_text_targets(GtkTargetList *
list,
                                                  guint info,
                                                  gboolean deserializable,
                                                  GtkTextBuffer * buffer);
extern void gtk_target_list_add_table(GtkTargetList * list,
                                       const GtkTargetEntry * targets,
                                       guint ntargets);
extern void gtk_target_list_add_text_targets(GtkTargetList * list,
                                              guint info);
extern void gtk_target_list_add_uri_targets(GtkTargetList * list,
                                             guint info);

```

```

extern gboolean gtk_target_list_find(GtkTargetList * list, GdkAtom
target,
                                guint * info);
extern GType gtk_target_list_get_type(void);
extern GtkTargetList *gtk_target_list_new(const GtkTargetEntry *
targets,
                                guint ntargets);
extern GtkTargetList gtk_target_list_ref(GtkTargetList * list);
extern void gtk_target_list_remove(GtkTargetList * list, GdkAtom
target);
extern void gtk_target_list_unref(GtkTargetList * list);
extern void gtk_target_table_free(GtkTargetEntry * targets,
                                gint n_targets);
extern
                                GtkTargetEntry
*gtk_target_table_new_from_list(GtkTargetList * list,
                                gint * n_targets);
extern gboolean gtk_targets_include_image(GdkAtom * targets,
                                gint n_targets,
                                gboolean writable);
extern gboolean gtk_targets_include_rich_text(GdkAtom * targets,
                                gint n_targets,
                                GtkTextBuffer * buffer);
extern gboolean gtk_targets_include_text(GdkAtom * targets,
                                gint n_targets);
extern gboolean gtk_targets_include_uri(GdkAtom * targets, gint
n_targets);
extern GType gtk_tearoff_menu_item_get_type(void);
extern GtkWidget *gtk_tearoff_menu_item_new(void);
extern
                                GtkTextAttributes
*gtk_text_attributes_copy(GtkTextAttributes *
                                src);
extern void gtk_text_attributes_copy_values(GtkTextAttributes *
src,
                                GtkTextAttributes * dest);
extern GType gtk_text_attributes_get_type(void);
extern GtkTextAttributes *gtk_text_attributes_new(void);
extern
                                GtkTextAttributes
*gtk_text_attributes_ref(GtkTextAttributes *
                                values);
extern void gtk_text_attributes_unref(GtkTextAttributes * values);
extern void gtk_text_buffer_add_selection_clipboard(GtkTextBuffer
* buffer,
                                GtkClipboard *
                                clipboard);
extern void gtk_text_buffer_apply_tag(GtkTextBuffer * buffer,
                                GtkTextTag * tag,
                                const GtkTextIter * start,
                                const GtkTextIter * end);
extern void gtk_text_buffer_apply_tag_by_name(GtkTextBuffer *
buffer,
                                const gchar * name,
                                const GtkTextIter * start,
                                const GtkTextIter * end);
extern gboolean gtk_text_buffer_backspace(GtkTextBuffer * buffer,
                                GtkTextIter * iter,
                                gboolean interactive,
                                gboolean default_editable);
extern void gtk_text_buffer_begin_user_action(GtkTextBuffer *
buffer);
extern void gtk_text_buffer_copy_clipboard(GtkTextBuffer * buffer,
                                GtkClipboard * clipboard);
extern GtkTextChildAnchor
*gtk_text_buffer_create_child_anchor(GtkTextBuffer * buffer,
                                GtkTextIter * iter);
extern GtkTextMark *gtk_text_buffer_create_mark(GtkTextBuffer *
buffer,

```

```

        const gchar * mark_name,
        const GtkTextIter * where,
        gboolean left_gravity);
extern GtkTextTag *gtk_text_buffer_create_tag(GtkTextBuffer *
buffer,
        const gchar * tag_name,
        const gchar *
        first_property_name, ...);
extern void gtk_text_buffer_cut_clipboard(GtkTextBuffer * buffer,
        GtkClipboard * clipboard,
        gboolean default_editable);
extern void gtk_text_buffer_delete(GtkTextBuffer * buffer,
        GtkTextIter * start, GtkTextIter *
end);
extern gboolean gtk_text_buffer_delete_interactive(GtkTextBuffer *
buffer,
        GtkTextIter *
        start_iter,
        GtkTextIter * end_iter,
        gboolean
        default_editable);
extern void gtk_text_buffer_delete_mark(GtkTextBuffer * buffer,
        GtkTextMark * mark);
extern void gtk_text_buffer_delete_mark_by_name(GtkTextBuffer *
buffer,
        const gchar * name);
extern gboolean gtk_text_buffer_delete_selection(GtkTextBuffer *
buffer,
        gboolean interactive,
        gboolean
        default_editable);
extern void gtk_text_buffer_end_user_action(GtkTextBuffer *
buffer);
extern void gtk_text_buffer_get_bounds(GtkTextBuffer * buffer,
        GtkTextIter * start,
        GtkTextIter * end);
extern gint gtk_text_buffer_get_char_count(GtkTextBuffer * buffer);
extern GtkTargetList
*gtk_text_buffer_get_copy_target_list(GtkTextBuffer *
buffer);
extern void gtk_text_buffer_get_end_iter(GtkTextBuffer * buffer,
        GtkTextIter * iter);
extern gboolean gtk_text_buffer_get_has_selection(GtkTextBuffer *
buffer);
extern GtkTextMark *gtk_text_buffer_get_insert(GtkTextBuffer *
buffer);
extern void gtk_text_buffer_get_iter_at_child_anchor(GtkTextBuffer
*
        buffer,
        GtkTextIter * iter,
        GtkTextChildAnchor *
        anchor);
extern void gtk_text_buffer_get_iter_at_line(GtkTextBuffer *
buffer,
        GtkTextIter * iter,
        gint line_number);
extern void gtk_text_buffer_get_iter_at_line_index(GtkTextBuffer *
buffer,
        GtkTextIter * iter,
        gint line_number,
        gint byte_index);
extern void gtk_text_buffer_get_iter_at_line_offset(GtkTextBuffer
*
buffer,
        GtkTextIter * iter,
        gint line_number,
        gint char_offset);

```

```

extern void gtk_text_buffer_get_iter_at_mark(GtkTextBuffer *
buffer,
                                           GtkTextIter * iter,
                                           GtkTextMark * mark);
extern void gtk_text_buffer_get_iter_at_offset(GtkTextBuffer *
buffer,
                                           GtkTextIter * iter,
                                           gint char_offset);
extern gint gtk_text_buffer_get_line_count(GtkTextBuffer * buffer);
extern GtkTextMark *gtk_text_buffer_get_mark(GtkTextBuffer *
buffer,
                                           const gchar * name);
extern gboolean gtk_text_buffer_get_modified(GtkTextBuffer *
buffer);
extern GtkTargetList
*gtk_text_buffer_get_paste_target_list(GtkTextBuffer *
buffer);
extern GtkTextMark
*gtk_text_buffer_get_selection_bound(GtkTextBuffer *
buffer);
extern gboolean gtk_text_buffer_get_selection_bounds(GtkTextBuffer
*
buffer,
                                           GtkTextIter * start,
                                           GtkTextIter * end);
extern gchar *gtk_text_buffer_get_slice(GtkTextBuffer * buffer,
                                           const GtkTextIter * start,
                                           const GtkTextIter * end,
                                           gboolean include_hidden_chars);
extern void gtk_text_buffer_get_start_iter(GtkTextBuffer * buffer,
                                           GtkTextIter * iter);
extern GtkTextTagTable
*gtk_text_buffer_get_tag_table(GtkTextBuffer *
buffer);
extern gchar *gtk_text_buffer_get_text(GtkTextBuffer * buffer,
                                           const GtkTextIter * start,
                                           const GtkTextIter * end,
                                           gboolean include_hidden_chars);
extern GType gtk_text_buffer_get_type(void);
extern void gtk_text_buffer_insert(GtkTextBuffer * buffer,
                                           GtkTextIter * iter, const gchar * text,
                                           gint len);
extern void gtk_text_buffer_insert_at_cursor(GtkTextBuffer *
buffer,
                                           const gchar * text, gint len);
extern void gtk_text_buffer_insert_child_anchor(GtkTextBuffer *
buffer,
                                           GtkTextIter * iter,
                                           GtkTextChildAnchor *
anchor);
extern gboolean gtk_text_buffer_insert_interactive(GtkTextBuffer *
buffer,
                                           GtkTextIter * iter,
                                           const gchar * text,
                                           gint len,
                                           gboolean
                                           default_editable);
extern gboolean
gtk_text_buffer_insert_interactive_at_cursor(GtkTextBuffer
* buffer,
                                           const gchar *
                                           text,
                                           gint len,
                                           gboolean
                                           default_editable);

```

```

extern void gtk_text_buffer_insert_pixbuf(GtkTextBuffer * buffer,
                                           GtkTextIter * iter,
                                           GdkPixbuf * pixbuf);
extern void gtk_text_buffer_insert_range(GtkTextBuffer * buffer,
                                           GtkTextIter * iter,
                                           const GtkTextIter * start,
                                           const GtkTextIter * end);
extern
gtk_text_buffer_insert_range_interactive(GtkTextBuffer *
                                           buffer,
                                           GtkTextIter *
                                           iter,
                                           const GtkTextIter
                                           * start,
                                           const GtkTextIter
                                           * end,
                                           gboolean
                                           default_editable);
extern void gtk_text_buffer_insert_with_tags(GtkTextBuffer *
buffer,
                                           GtkTextIter * iter,
                                           const gchar * text, gint len,
                                           GtkTextTag * first_tag, ...);
extern void gtk_text_buffer_insert_with_tags_by_name(GtkTextBuffer
*
                                           buffer,
                                           GtkTextIter * iter,
                                           const gchar * text,
                                           gint len,
                                           const gchar *
                                           first_tag_name, ...);
extern void gtk_text_buffer_move_mark(GtkTextBuffer * buffer,
                                       GtkTextMark * mark,
                                       const GtkTextIter * where);
extern void gtk_text_buffer_move_mark_by_name(GtkTextBuffer *
buffer,
                                           const gchar * name,
                                           const GtkTextIter * where);
extern GtkTextBuffer *gtk_text_buffer_new(GtkTextTagTable * table);
extern void gtk_text_buffer_paste_clipboard(GtkTextBuffer * buffer,
                                           GtkClipboard * clipboard,
                                           GtkTextIter *
                                           override_location,
                                           gboolean default_editable);
extern void gtk_text_buffer_place_cursor(GtkTextBuffer * buffer,
                                           const GtkTextIter * where);
extern void gtk_text_buffer_remove_all_tags(GtkTextBuffer * buffer,
                                           const GtkTextIter * start,
                                           const GtkTextIter * end);
extern
void
gtk_text_buffer_remove_selection_clipboard(GtkTextBuffer *
                                           buffer,
                                           GtkClipboard *
                                           clipboard);
extern void gtk_text_buffer_remove_tag(GtkTextBuffer * buffer,
                                       GtkTextTag * tag,
                                       const GtkTextIter * start,
                                       const GtkTextIter * end);
extern void gtk_text_buffer_remove_tag_by_name(GtkTextBuffer *
buffer,
                                           const gchar * name,
                                           const GtkTextIter * start,
                                           const GtkTextIter * end);
extern void gtk_text_buffer_select_range(GtkTextBuffer * buffer,
                                           const GtkTextIter * ins,
                                           const GtkTextIter * bound);

```

```

extern void gtk_text_buffer_set_modified(GtkTextBuffer * buffer,
                                         gboolean setting);
extern void gtk_text_buffer_set_text(GtkTextBuffer * buffer,
                                     const gchar * text, gint len);
extern GType gtk_text_buffer_target_info_get_type(void);
extern
                                         gboolean
gtk_text_child_anchor_get_deleted(GtkTextChildAnchor *
                                  anchor);
extern GType gtk_text_child_anchor_get_type(void);
extern GList *gtk_text_child_anchor_get_widgets(GtkTextChildAnchor
*
                                         anchor);
extern GtkTextChildAnchor *gtk_text_child_anchor_new(void);
extern GType gtk_text_direction_get_type(void);
extern gboolean gtk_text_iter_backward_char(GtkTextIter * iter);
extern gboolean gtk_text_iter_backward_chars(GtkTextIter * iter,
                                             gint count);
extern gboolean gtk_text_iter_backward_cursor_position(GtkTextIter
* iter);
extern
                                         gboolean
gtk_text_iter_backward_cursor_positions(GtkTextIter * iter,
                                         gint count);
extern gboolean gtk_text_iter_backward_find_char(GtkTextIter *
iter,
                                                GtkTextCharPredicate pred,
                                                gpointer user_data,
                                                const GtkTextIter *
limit);
extern gboolean gtk_text_iter_backward_line(GtkTextIter * iter);
extern gboolean gtk_text_iter_backward_lines(GtkTextIter * iter,
                                             gint count);
extern gboolean gtk_text_iter_backward_search(const GtkTextIter *
iter,
                                             const gchar * str,
                                             GtkTextSearchFlags flags,
                                             GtkTextIter * match_start,
                                             GtkTextIter * match_end,
                                             const GtkTextIter * limit);
extern gboolean gtk_text_iter_backward_sentence_start(GtkTextIter
* iter);
extern gboolean gtk_text_iter_backward_sentence_starts(GtkTextIter
* iter,
                                                       gint count);
extern gboolean gtk_text_iter_backward_to_tag_toggle(GtkTextIter *
iter,
                                                     GtkTextTag * tag);
extern
                                         gboolean
gtk_text_iter_backward_visible_cursor_position(GtkTextIter
* iter);
extern
                                         gboolean
gtk_text_iter_backward_visible_cursor_positions(GtkTextIter
* iter,
                                                gint
count);
extern gboolean gtk_text_iter_backward_visible_line(GtkTextIter *
iter);
extern gboolean gtk_text_iter_backward_visible_lines(GtkTextIter *
iter,
                                                     gint count);
extern
                                         gboolean
gtk_text_iter_backward_visible_word_start(GtkTextIter *
iter);
extern
                                         gboolean
gtk_text_iter_backward_visible_word_starts(GtkTextIter *
iter,
                                           gint count);

```

```

extern gboolean gtk_text_iter_backward_word_start(GtkTextIter *
iter);
extern gboolean gtk_text_iter_backward_word_starts(GtkTextIter *
iter,
gint count);
extern gboolean gtk_text_iter_begins_tag(const GtkTextIter * iter,
GtkTextTag * tag);
extern gboolean gtk_text_iter_can_insert(const GtkTextIter * iter,
gboolean default_editability);
extern gint gtk_text_iter_compare(const GtkTextIter * lhs,
const GtkTextIter * rhs);
extern GtkTextIter *gtk_text_iter_copy(const GtkTextIter * iter);
extern gboolean gtk_text_iter_editable(const GtkTextIter * iter,
gboolean default_setting);
extern gboolean gtk_text_iter_ends_line(const GtkTextIter * iter);
extern gboolean gtk_text_iter_ends_sentence(const GtkTextIter *
iter);
extern gboolean gtk_text_iter_ends_tag(const GtkTextIter * iter,
GtkTextTag * tag);
extern gboolean gtk_text_iter_ends_word(const GtkTextIter * iter);
extern gboolean gtk_text_iter_equal(const GtkTextIter * lhs,
const GtkTextIter * rhs);
extern gboolean gtk_text_iter_forward_char(GtkTextIter * iter);
extern gboolean gtk_text_iter_forward_chars(GtkTextIter * iter,
gint count);
extern gboolean gtk_text_iter_forward_cursor_position(GtkTextIter
* iter);
extern gboolean gtk_text_iter_forward_cursor_positions(GtkTextIter
* iter,
gint count);
extern gboolean gtk_text_iter_forward_find_char(GtkTextIter * iter,
GtkTextCharPredicate pred,
gpointer user_data,
const GtkTextIter * limit);
extern gboolean gtk_text_iter_forward_line(GtkTextIter * iter);
extern gboolean gtk_text_iter_forward_lines(GtkTextIter * iter,
gint count);
extern gboolean gtk_text_iter_forward_search(const GtkTextIter *
iter,
const gchar * str,
GtkTextSearchFlags flags,
GtkTextIter * match_start,
GtkTextIter * match_end,
const GtkTextIter * limit);
extern gboolean gtk_text_iter_forward_sentence_end(GtkTextIter *
iter);
extern gboolean gtk_text_iter_forward_sentence_ends(GtkTextIter *
iter,
gint count);
extern void gtk_text_iter_forward_to_end(GtkTextIter * iter);
extern gboolean gtk_text_iter_forward_to_line_end(GtkTextIter *
iter);
extern gboolean gtk_text_iter_forward_to_tag_toggle(GtkTextIter *
iter,
GtkTextTag * tag);
extern gboolean
gtk_text_iter_forward_visible_cursor_position(GtkTextIter *
iter);
extern gboolean
gtk_text_iter_forward_visible_cursor_positions(GtkTextIter
* iter,
gint count);
extern gboolean gtk_text_iter_forward_visible_line(GtkTextIter *
iter);
extern gboolean gtk_text_iter_forward_visible_lines(GtkTextIter *
iter,

```

```

                                gint count);
extern gboolean gtk_text_iter_forward_visible_word_end(GtkTextIter
* iter);
extern                                gboolean
gtk_text_iter_forward_visible_word_ends(GtkTextIter * iter,
                                gint count);
extern gboolean gtk_text_iter_forward_word_end(GtkTextIter * iter);
extern gboolean gtk_text_iter_forward_word_ends(GtkTextIter * iter,
                                gint count);
extern void gtk_text_iter_free(GtkTextIter * iter);
extern gboolean gtk_text_iter_get_attributes(const GtkTextIter *
iter,
                                GtkTextAttributes * values);
extern GtkTextBuffer *gtk_text_iter_get_buffer(const GtkTextIter *
iter);
extern gint gtk_text_iter_get_bytes_in_line(const GtkTextIter *
iter);
extern gunichar gtk_text_iter_get_char(const GtkTextIter * iter);
extern gint gtk_text_iter_get_chars_in_line(const GtkTextIter *
iter);
extern GtkTextChildAnchor *gtk_text_iter_get_child_anchor(const
GtkTextIter
                                * iter);
extern PangoLanguage *gtk_text_iter_get_language(const GtkTextIter
* iter);
extern gint gtk_text_iter_get_line(const GtkTextIter * iter);
extern gint gtk_text_iter_get_line_index(const GtkTextIter * iter);
extern gint gtk_text_iter_get_line_offset(const GtkTextIter * iter);
extern GSList *gtk_text_iter_get_marks(const GtkTextIter * iter);
extern gint gtk_text_iter_get_offset(const GtkTextIter * iter);
extern GdkPixbuf *gtk_text_iter_get_pixbuf(const GtkTextIter *
iter);
extern gchar *gtk_text_iter_get_slice(const GtkTextIter * start,
                                const GtkTextIter * end);
extern GSList *gtk_text_iter_get_tags(const GtkTextIter * iter);
extern gchar *gtk_text_iter_get_text(const GtkTextIter * start,
                                const GtkTextIter * end);
extern GSList *gtk_text_iter_get_toggled_tags(const GtkTextIter *
iter,
                                gboolean toggled_on);
extern GType gtk_text_iter_get_type(void);
extern gint gtk_text_iter_get_visible_line_index(const GtkTextIter
* iter);
extern gint gtk_text_iter_get_visible_line_offset(const
GtkTextIter *
                                iter);
extern gchar *gtk_text_iter_get_visible_slice(const GtkTextIter *
start,
                                const GtkTextIter * end);
extern gchar *gtk_text_iter_get_visible_text(const GtkTextIter *
start,
                                const GtkTextIter * end);
extern gboolean gtk_text_iter_has_tag(const GtkTextIter * iter,
                                GtkTextTag * tag);
extern gboolean gtk_text_iter_in_range(const GtkTextIter * iter,
                                const GtkTextIter * start,
                                const GtkTextIter * end);
extern gboolean gtk_text_iter_inside_sentence(const GtkTextIter *
iter);
extern gboolean gtk_text_iter_inside_word(const GtkTextIter * iter);
extern gboolean gtk_text_iter_is_cursor_position(const GtkTextIter
* iter);
extern gboolean gtk_text_iter_is_end(const GtkTextIter * iter);
extern gboolean gtk_text_iter_is_start(const GtkTextIter * iter);
extern void gtk_text_iter_order(GtkTextIter * first, GtkTextIter *
second);

```



```

extern void gtk_text_iter_set_line(GtkTextIter * iter, gint
line_number);
extern void gtk_text_iter_set_line_index(GtkTextIter * iter,
gint byte_on_line);
extern void gtk_text_iter_set_line_offset(GtkTextIter * iter,
gint char_on_line);
extern void gtk_text_iter_set_offset(GtkTextIter * iter, gint
char_offset);
extern void gtk_text_iter_set_visible_line_index(GtkTextIter *
iter,
gint byte_on_line);
extern void gtk_text_iter_set_visible_line_offset(GtkTextIter *
iter,
gint char_on_line);
extern gboolean gtk_text_iter_starts_line(const GtkTextIter * iter);
extern gboolean gtk_text_iter_starts_sentence(const GtkTextIter *
iter);
extern gboolean gtk_text_iter_starts_word(const GtkTextIter * iter);
extern gboolean gtk_text_iter_toggles_tag(const GtkTextIter * iter,
GtkTextTag * tag);
extern GtkTextBuffer *gtk_text_mark_get_buffer(GtkTextMark *);
extern gboolean gtk_text_mark_get_deleted(GtkTextMark *);
extern gboolean gtk_text_mark_get_left_gravity(GtkTextMark *);
extern const char *gtk_text_mark_get_name(GtkTextMark *);
extern GType gtk_text_mark_get_type(void);
extern gboolean gtk_text_mark_get_visible(GtkTextMark *);
extern void gtk_text_mark_set_visible(GtkTextMark *, gboolean);
extern GType gtk_text_search_flags_get_type(void);
extern gboolean gtk_text_tag_event(GtkTextTag * tag,
GObject * event_object,
GdkEvent * event,
const GtkTextIter * iter);
extern gint gtk_text_tag_get_priority(GtkTextTag * tag);
extern GType gtk_text_tag_get_type(void);
extern GtkTextTag *gtk_text_tag_new(const gchar * name);
extern void gtk_text_tag_set_priority(GtkTextTag * tag, gint
priority);
extern void gtk_text_tag_table_add(GtkTextTagTable * table,
GtkTextTag * tag);
extern void gtk_text_tag_table_foreach(GtkTextTagTable * table,
GtkTextTagTableForeach func,
gpointer data);
extern gint gtk_text_tag_table_get_size(GtkTextTagTable * table);
extern GType gtk_text_tag_table_get_type(void);
extern GtkTextTag *gtk_text_tag_table_lookup(GtkTextTagTable *
table,
const gchar * name);
extern GtkTextTagTable *gtk_text_tag_table_new(void);
extern void gtk_text_tag_table_remove(GtkTextTagTable * table,
GtkTextTag * tag);
extern void gtk_text_view_add_child_at_anchor(GtkTextView *
text_view,
GtkWidget * child,
GtkTextChildAnchor *
anchor);
extern void gtk_text_view_add_child_in_window(GtkTextView *
text_view,
GtkWidget * child,
GtkTextWindowType
which_window, gint xpos,
gint ypos);
extern gboolean gtk_text_view_backward_display_line(GtkTextView *
text_view,
GtkTextIter * iter);
extern
gboolean
gtk_text_view_backward_display_line_start(GtkTextView *

```

```

        text_view,
        GtkTextIter *
        iter);

extern void gtk_text_view_buffer_to_window_coords(GtkTextView *
text_view,

        GtkTextWindowType win,
        gint buffer_x,
        gint buffer_y,
        gint * window_x,
        gint * window_y);

extern gboolean gtk_text_view_forward_display_line(GtkTextView *
text_view,

        GtkTextIter * iter);

extern gboolean gtk_text_view_forward_display_line_end(GtkTextView *
*
        text_view,
        GtkTextIter * iter);

extern gboolean gtk_text_view_get_accepts_tab(GtkTextView *
text_view);

extern gint gtk_text_view_get_border_window_size(GtkTextView *
text_view,

        GtkTextWindowType type);

extern GtkTextBuffer *gtk_text_view_get_buffer(GtkTextView *
text_view);

extern gboolean gtk_text_view_get_cursor_visible(GtkTextView *
text_view);

extern
        GtkTextAttributes
*gtk_text_view_get_default_attributes(GtkTextView

        *
        text_view);

extern gboolean gtk_text_view_get_editable(GtkTextView *
text_view);

extern gint gtk_text_view_get_indent(GtkTextView * text_view);

extern void gtk_text_view_get_iter_at_location(GtkTextView *
text_view,

        GtkTextIter * iter, gint x,
        gint y);

extern void gtk_text_view_get_iter_at_position(GtkTextView *
text_view,

        GtkTextIter * iter,
        gint * trailing, gint x,
        gint y);

extern void gtk_text_view_get_iter_location(GtkTextView *
text_view,

        const GtkTextIter * iter,
        GdkRectangle * location);

extern
        GtkJustification
gtk_text_view_get_justification(GtkTextView *

        text_view);

extern gint gtk_text_view_get_left_margin(GtkTextView * text_view);

extern void gtk_text_view_get_line_at_y(GtkTextView * text_view,
        GtkTextIter * target_iter, gint y,
        gint * line_top);

extern void gtk_text_view_get_line_yrange(GtkTextView * text_view,
        const GtkTextIter * iter,
        gint * y, gint * height);

extern gboolean gtk_text_view_get_overwrite(GtkTextView *
text_view);

extern gint gtk_text_view_get_pixels_above_lines(GtkTextView *
text_view);

extern gint gtk_text_view_get_pixels_below_lines(GtkTextView *
text_view);

extern gint gtk_text_view_get_pixels_inside_wrap(GtkTextView *
text_view);

extern gint gtk_text_view_get_right_margin(GtkTextView *
text_view);

```

```

extern PangoTabArray *gtk_text_view_get_tabs(GtkTextView *
text_view);
extern GType gtk_text_view_get_type(void);
extern void gtk_text_view_get_visible_rect(GtkTextView * text_view,
GdkRectangle * visible_rect);
extern GdkWindow *gtk_text_view_get_window(GtkTextView * text_view,
GtkTextWindowType win);
extern GtkTextWindowType gtk_text_view_get_window_type(GtkTextView
*
text_view,
GdkWindow * window);
extern GtkWrapMode gtk_text_view_get_wrap_mode(GtkTextView *
text_view);
extern void gtk_text_view_move_child(GtkTextView * text_view,
GtkWidget * child, gint xpos,
gint ypos);
extern gboolean gtk_text_view_move_mark_onscreen(GtkTextView *
text_view,
GtkTextMark * mark);
extern gboolean gtk_text_view_move_visually(GtkTextView *
text_view,
GtkTextIter * iter,
gint count);
extern GtkWidget *gtk_text_view_new(void);
extern GtkWidget *gtk_text_view_new_with_buffer(GtkTextBuffer *
buffer);
extern gboolean gtk_text_view_place_cursor_onscreen(GtkTextView *
text_view);
extern void gtk_text_view_scroll_mark_onscreen(GtkTextView *
text_view,
GtkTextMark * mark);
extern gboolean gtk_text_view_scroll_to_iter(GtkTextView *
text_view,
GtkTextIter * iter,
gdouble within_margin,
gboolean use_align,
gdouble xalign,
gdouble yalign);
extern void gtk_text_view_scroll_to_mark(GtkTextView * text_view,
GtkTextMark * mark,
gdouble within_margin,
gboolean use_align,
gdouble xalign, gdouble yalign);
extern void gtk_text_view_set_accepts_tab(GtkTextView * text_view,
gboolean accepts_tab);
extern void gtk_text_view_set_border_window_size(GtkTextView *
text_view,
GtkTextWindowType type,
gint size);
extern void gtk_text_view_set_buffer(GtkTextView * text_view,
GtkTextBuffer * buffer);
extern void gtk_text_view_set_cursor_visible(GtkTextView *
text_view,
gboolean setting);
extern void gtk_text_view_set_editable(GtkTextView * text_view,
gboolean setting);
extern void gtk_text_view_set_indent(GtkTextView * text_view, gint
indent);
extern void gtk_text_view_set_justification(GtkTextView *
text_view,
GtkJustification
justification);
extern void gtk_text_view_set_left_margin(GtkTextView * text_view,
gint left_margin);
extern void gtk_text_view_set_overwrite(GtkTextView * text_view,
gboolean overwrite);

```

```

extern void gtk_text_view_set_pixels_above_lines(GtkTextView *
text_view,
                                           gint pixels_above_lines);
extern void gtk_text_view_set_pixels_below_lines(GtkTextView *
text_view,
                                           gint pixels_below_lines);
extern void gtk_text_view_set_pixels_inside_wrap(GtkTextView *
text_view,
                                           gint pixels_inside_wrap);
extern void gtk_text_view_set_right_margin(GtkTextView * text_view,
                                           gint right_margin);
extern void gtk_text_view_set_tabs(GtkTextView * text_view,
                                   PangoTabArray * tabs);
extern void gtk_text_view_set_wrap_mode(GtkTextView * text_view,
                                         GtkWrapMode wrap_mode);
extern gboolean gtk_text_view_starts_display_line(GtkTextView *
text_view,
                                           const GtkTextIter *
                                           iter);
extern void gtk_text_view_window_to_buffer_coords(GtkTextView *
text_view,
                                           GtkTextWindowType win,
                                           gint window_x,
                                           gint window_y,
                                           gint * buffer_x,
                                           gint * buffer_y);
extern GType gtk_text_window_type_get_type(void);
extern gboolean gtk_toggle_action_get_active(GtkToggleAction *
action);
extern
                                           gboolean
gtk_toggle_action_get_draw_as_radio(GtkToggleAction *
action);
extern GType gtk_toggle_action_get_type(void);
extern GtkToggleAction *gtk_toggle_action_new(const gchar * name,
                                           const gchar * label,
                                           const gchar * tooltip,
                                           const gchar * stock_id);
extern void gtk_toggle_action_set_active(GtkToggleAction * action,
                                           gboolean is_active);
extern void gtk_toggle_action_set_draw_as_radio(GtkToggleAction *
action,
                                           gboolean draw_as_radio);
extern void gtk_toggle_action_toggled(GtkToggleAction * action);
extern gboolean gtk_toggle_button_get_active(GtkToggleButton *
toggle_button);
extern gboolean gtk_toggle_button_get_inconsistent(GtkToggleButton *
*
toggle_button);
extern gboolean gtk_toggle_button_get_mode(GtkToggleButton *
toggle_button);
extern GType gtk_toggle_button_get_type(void);
extern GtkWidget *gtk_toggle_button_new(void);
extern GtkWidget *gtk_toggle_button_new_with_label(const gchar *
label);
extern GtkWidget *gtk_toggle_button_new_with_mnemonic(const gchar
* label);
extern void gtk_toggle_button_set_active(GtkToggleButton *
toggle_button,
                                           gboolean is_active);
extern void gtk_toggle_button_set_inconsistent(GtkToggleButton *
toggle_button,
                                           gboolean setting);
extern void gtk_toggle_button_set_mode(GtkToggleButton *
toggle_button,
                                           gboolean draw_indicator);

```

```

extern void gtk_toggle_button_toggled(GtkToggleButton *
toggle_button);
extern gboolean
gtk_toggle_tool_button_get_active(GtkToggleToolButton *
button);
extern GType gtk_toggle_tool_button_get_type(void);
extern GtkWidget *gtk_toggle_tool_button_new(void);
extern GtkWidget *gtk_toggle_tool_button_new_from_stock(const
gchar *
stock_id);
extern void gtk_toggle_tool_button_set_active(GtkToggleToolButton
* button,
gboolean is_active);
extern const gchar *gtk_tool_button_get_icon_name(GtkToolButton *
button);
extern GtkWidget *gtk_tool_button_get_icon_widget(GtkToolButton *
button);
extern const gchar *gtk_tool_button_get_label(GtkToolButton *
button);
extern GtkWidget *gtk_tool_button_get_label_widget(GtkToolButton *
button);
extern const gchar *gtk_tool_button_get_stock_id(GtkToolButton *
button);
extern GType gtk_tool_button_get_type(void);
extern gboolean gtk_tool_button_get_use_underline(GtkToolButton *
button);
extern GtkWidget *gtk_tool_button_new(GtkWidget * icon_widget,
const gchar * label);
extern GtkWidget *gtk_tool_button_new_from_stock(const gchar *
stock_id);
extern void gtk_tool_button_set_icon_name(GtkToolButton * button,
const gchar * icon_name);
extern void gtk_tool_button_set_icon_widget(GtkToolButton * button,
GtkWidget * icon_widget);
extern void gtk_tool_button_set_label(GtkToolButton * button,
const gchar * label);
extern void gtk_tool_button_set_label_widget(GtkToolButton *
button,
GtkWidget * label_widget);
extern void gtk_tool_button_set_stock_id(GtkToolButton * button,
const gchar * stock_id);
extern void gtk_tool_button_set_use_underline(GtkToolButton *
button,
gboolean use_underline);
extern gboolean gtk_tool_item_get_expand(GtkToolItem * tool_item);
extern gboolean gtk_tool_item_get_homogeneous(GtkToolItem *
tool_item);
extern GtkIconSize gtk_tool_item_get_icon_size(GtkToolItem *
tool_item);
extern gboolean gtk_tool_item_get_is_important(GtkToolItem *
tool_item);
extern GtkOrientation gtk_tool_item_get_orientation(GtkToolItem *
tool_item);
extern GtkWidget *gtk_tool_item_get_proxy_menu_item(GtkToolItem *
tool_item,
const gchar *
menu_item_id);
extern GtkReliefStyle gtk_tool_item_get_relief_style(GtkToolItem *
tool_item);
extern GtkToolbarStyle gtk_tool_item_get_toolbar_style(GtkToolItem
*
tool_item);
extern GType gtk_tool_item_get_type(void);
extern gboolean gtk_tool_item_get_use_drag_window(GtkToolItem *
tool_item);
extern gboolean gtk_tool_item_get_visible_horizontal(GtkToolItem *

```

```

                                tool_item);
extern gboolean gtk_tool_item_get_visible_vertical(GtkToolItem *
                                tool_item);
extern GtkWidget *gtk_tool_item_new(void);
extern void gtk_tool_item_rebuild_menu(GtkToolItem * tool_item);
extern void gtk_tool_item_retrieve_proxy_menu_item(GtkToolItem *
                                tool_item);
extern void gtk_tool_item_set_expand(GtkToolItem * tool_item,
                                gboolean expand);
extern void gtk_tool_item_set_homogeneous(GtkToolItem * tool_item,
                                gboolean homogeneous);
extern void gtk_tool_item_set_is_important(GtkToolItem * tool_item,
                                gboolean is_important);
extern void gtk_tool_item_set_proxy_menu_item(GtkToolItem *
tool_item,
                                const gchar * menu_item_id,
                                GtkWidget * menu_item);
extern void gtk_tool_item_set_tooltip(GtkToolItem * tool_item,
                                GtkTooltips * tooltips,
                                const gchar * tip_text,
                                const gchar * tip_private);
extern void gtk_tool_item_set_use_drag_window(GtkToolItem *
tool_item,
                                gboolean use_drag_window);
extern void gtk_tool_item_set_visible_horizontal(GtkToolItem *
tool_item,
                                gboolean
                                visible_horizontal);
extern void gtk_tool_item_set_visible_vertical(GtkToolItem *
tool_item,
                                gboolean visible_vertical);
extern GType gtk_toolbar_child_type_get_type(void);
extern gint gtk_toolbar_get_drop_index(GtkToolbar * toolbar, gint
x,
                                gint y);
extern GtkWidget *gtk_toolbar_get_icon_size(GtkToolbar * toolbar);
extern gint gtk_toolbar_get_item_index(GtkToolbar * toolbar,
                                GtkWidget * item);
extern gint gtk_toolbar_get_n_items(GtkToolbar * toolbar);
extern GtkWidget *gtk_toolbar_get_nth_item(GtkToolbar * toolbar,
gint n);
extern GtkOrientation gtk_toolbar_get_orientation(GtkToolbar *
toolbar);
extern GtkReliefStyle gtk_toolbar_get_relief_style(GtkToolbar *
toolbar);
extern gboolean gtk_toolbar_get_show_arrow(GtkToolbar * toolbar);
extern GtkWidget *gtk_toolbar_get_style(GtkToolbar * toolbar);
extern gboolean gtk_toolbar_get_tooltips(GtkToolbar * toolbar);
extern GType gtk_toolbar_get_type(void);
extern void gtk_toolbar_insert(GtkToolbar * toolbar, GtkWidget *
item,
                                gint pos);
extern GtkWidget *gtk_toolbar_new(void);
extern void gtk_toolbar_set_drop_highlight_item(GtkToolbar *
toolbar,
                                GtkWidget * tool_item,
                                gint index_);
extern void gtk_toolbar_set_icon_size(GtkToolbar * toolbar,
                                GtkWidget * icon_size);
extern void gtk_toolbar_set_orientation(GtkToolbar * toolbar,
                                GtkOrientation orientation);
extern void gtk_toolbar_set_show_arrow(GtkToolbar * toolbar,
                                gboolean show_arrow);
extern void gtk_toolbar_set_style(GtkToolbar * toolbar,
                                GtkWidget * style);

```

```

extern void gtk_toolbar_set_tooltips(GtkToolbar * toolbar,
                                     gboolean enable);
extern GType gtk_toolbar_space_style_get_type(void);
extern GType gtk_toolbar_style_get_type(void);
extern void gtk_toolbar_unset_style(GtkToolbar * toolbar);
extern GtkTooltipsData *gtk_tooltips_data_get(GtkWidget * widget);
extern void gtk_tooltips_disable(GtkTooltips * tooltips);
extern void gtk_tooltips_enable(GtkTooltips * tooltips);
extern void gtk_tooltips_force_window(GtkTooltips * tooltips);
extern gboolean gtk_tooltips_get_info_from_tip_window(GtkWindow *
                                                       tip_window,
                                                       GtkTooltips *
                                                       *tooltips,
                                                       GtkWidget *
                                                       *current_widget);

extern GType gtk_tooltips_get_type(void);
extern GtkTooltips *gtk_tooltips_new(void);
extern void gtk_tooltips_set_tip(GtkTooltips * tooltips,
                                  GtkWidget * widget,
                                  const gchar * tip_text,
                                  const gchar * tip_private);

extern
gtk_tree_drag_dest_drag_data_received(GtkTreeDragDest *
                                       drag_dest,
                                       GtkTreePath * dest,
                                       GtkSelectionData *
                                       selection_data);

extern GType gtk_tree_drag_dest_get_type(void);
extern
gtk_tree_drag_dest_row_drop_possible(GtkTreeDragDest *
                                       drag_dest,
                                       GtkTreePath *
                                       dest_path,
                                       GtkSelectionData *
                                       selection_data);

extern
gtk_tree_drag_source_drag_data_delete(GtkTreeDragSource *
                                       drag_source,
                                       GtkTreePath * path);

extern
gtk_tree_drag_source_drag_data_get(GtkTreeDragSource *
                                    drag_source,
                                    GtkTreePath * path,
                                    GtkSelectionData *
                                    selection_data);

extern GType gtk_tree_drag_source_get_type(void);
extern
gtk_tree_drag_source_row_draggable(GtkTreeDragSource *
                                    drag_source,
                                    GtkTreePath * path);
extern gboolean gtk_tree_get_row_drag_data(GtkSelectionData *
                                             selection_data,
                                             GtkTreeModel * *tree_model,
                                             GtkTreePath * *path);

extern GtkTreeIter *gtk_tree_iter_copy(GtkTreeIter * iter);
extern void gtk_tree_iter_free(GtkTreeIter * iter);
extern GType gtk_tree_iter_get_type(void);
extern void gtk_tree_model_filter_clear_cache(GtkTreeModelFilter *
                                              filter);
extern gboolean
gtk_tree_model_filter_convert_child_iter_to_iter(GtkTreeModelFilter *
                                                  filter,
                                                  GtkTreeIter * filter_iter,
                                                  GtkTreeIter * child_iter);

extern GtkTreePath

```

```

*gtk_tree_model_filter_convert_child_path_to_path(GtkTreeModelFilter *
                                                    filter,
                                                    GtkTreePath *
                                                    child_path);

extern void
gtk_tree_model_filter_convert_iter_to_child_iter(GtkTreeModelFilter *
                                                    filter,
                                                    GtkTreeIter * child_iter,
                                                    GtkTreeIter *
                                                    filter_iter);

extern GtkTreePath
gtk_tree_model_filter_convert_path_to_child_path(GtkTreeModelFilter *
                                                    filter,
                                                    GtkTreePath *
                                                    filter_path);

extern GtkTreeModel
*gtk_tree_model_filter_get_model(GtkTreeModelFilter *
                                  filter);

extern GType gtk_tree_model_filter_get_type(void);
extern GtkTreeModel *gtk_tree_model_filter_new(GtkTreeModel *
child_model,
                                                GtkTreePath * root);

extern void gtk_tree_model_filter_refilter(GtkTreeModelFilter *
filter);
extern void
gtk_tree_model_filter_set_modify_func(GtkTreeModelFilter *
                                       filter, gint n_columns,
                                       GType * types,

                                       GtkTreeModelFilterModifyFunc
                                       func, gpointer data,
                                       GtkDestroyNotify
                                       destroy);

extern void
gtk_tree_model_filter_set_visible_column(GtkTreeModelFilter *
                                         filter, gint column);

extern void
gtk_tree_model_filter_set_visible_func(GtkTreeModelFilter *
                                       filter,

                                       GtkTreeModelFilterVisibleFunc
                                       func, gpointer data,
                                       GtkDestroyNotify
                                       destroy);

extern GType gtk_tree_model_flags_get_type(void);
extern void gtk_tree_model_foreach(GtkTreeModel * model,
                                   GtkTreeModelForeachFunc func,
                                   gpointer user_data);
extern void gtk_tree_model_get(GtkTreeModel * tree_model,
                               GtkTreeIter * iter, ...);
extern GType gtk_tree_model_get_column_type(GtkTreeModel *
tree_model,
                                             gint index);
extern GtkTreeModelFlags gtk_tree_model_get_flags(GtkTreeModel *
tree_model);
extern gboolean gtk_tree_model_get_iter(GtkTreeModel * tree_model,
                                         GtkTreeIter * iter,
                                         GtkTreePath * path);
extern gboolean gtk_tree_model_get_iter_first(GtkTreeModel *
tree_model,
                                              GtkTreeIter * iter);

```



```

extern gboolean gtk_tree_model_get_iter_from_string(GtkTreeModel *
                                                    tree_model,
                                                    GtkTreeIter * iter,
                                                    const gchar *
                                                    path_string);
extern gint      gtk_tree_model_get_n_columns(GtkTreeModel *
tree_model);
extern GtkTreePath *gtk_tree_model_get_path(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter);
extern gchar *gtk_tree_model_get_string_from_iter(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter);
extern GType gtk_tree_model_get_type(void);
extern void gtk_tree_model_get_valist(GtkTreeModel * tree_model,
                                     GtkTreeIter * iter,
                                     va_list var_args);
extern void gtk_tree_model_get_value(GtkTreeModel * tree_model,
                                     GtkTreeIter * iter, gint column,
                                     GValue * value);
extern gboolean  gtk_tree_model_iter_children(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter,
                                           GtkTreeIter * parent);
extern gboolean  gtk_tree_model_iter_has_child(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter);
extern gint      gtk_tree_model_iter_n_children(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter);
extern gboolean  gtk_tree_model_iter_next(GtkTreeModel * tree_model,
                                           GtkTreeIter * iter);
extern gboolean  gtk_tree_model_iter_nth_child(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter,
                                           GtkTreeIter * parent,
                                           gint n);
extern gboolean  gtk_tree_model_iter_parent(GtkTreeModel *
tree_model,
                                           GtkTreeIter * iter,
                                           GtkTreeIter * child);
extern void gtk_tree_model_ref_node(GtkTreeModel * tree_model,
                                    GtkTreeIter * iter);
extern void gtk_tree_model_row_changed(GtkTreeModel * tree_model,
                                       GtkTreePath * path,
                                       GtkTreeIter * iter);
extern void gtk_tree_model_row_deleted(GtkTreeModel * tree_model,
                                       GtkTreePath * path);
extern void gtk_tree_model_row_has_child_toggled(GtkTreeModel *
tree_model,
                                       GtkTreePath * path,
                                       GtkTreeIter * iter);
extern void gtk_tree_model_row_inserted(GtkTreeModel * tree_model,
                                       GtkTreePath * path,
                                       GtkTreeIter * iter);
extern void      gtk_tree_model_rows_reordered(GtkTreeModel *
tree_model,
                                       GtkTreePath * path,
                                       GtkTreeIter * iter,
                                       gint * new_order);
extern void gtk_tree_model_sort_clear_cache(GtkTreeModelSort *
tree_model_sort);
extern gboolean
gtk_tree_model_sort_convert_child_iter_to_iter(GtkTreeModelSort *
tree_model_sort,
                                           GtkTreeIter * sort_iter,

```

```

                                GtkTreeIter * child_iter);

extern GtkTreePath

*gtk_tree_model_sort_convert_child_path_to_path(GtkTreeModelSort *
                                                tree_model_sort,
                                                GtkTreePath *
                                                child_path);

extern void
gtk_tree_model_sort_convert_iter_to_child_iter(GtkTreeModelSort
                                                *
                                                tree_model_sort,
                                                GtkTreeIter *
                                                child_iter,
                                                GtkTreeIter *
                                                sorted_iter);

extern GtkTreePath

*gtk_tree_model_sort_convert_path_to_child_path(GtkTreeModelSort *
                                                tree_model_sort,
                                                GtkTreePath *
                                                sorted_path);

extern GtkTreeModel
*gtk_tree_model_sort_get_model(GtkTreeModelSort *
                                tree_model);

extern GType gtk_tree_model_sort_get_type(void);
extern gboolean gtk_tree_model_sort_iter_is_valid(GtkTreeModelSort
*
                                                tree_model_sort,
                                                GtkTreeIter * iter);

extern GtkTreeModel
*gtk_tree_model_sort_new_with_model(GtkTreeModel *
                                    child_model);

extern void
gtk_tree_model_sort_reset_default_sort_func(GtkTreeModelSort *
                                             tree_model_sort);

extern void gtk_tree_model_unref_node(GtkTreeModel * tree_model,
                                       GtkTreeIter * iter);

extern void gtk_tree_path_append_index(GtkTreePath * path, gint
index_);
extern gint gtk_tree_path_compare(const GtkTreePath * a,
                                 const GtkTreePath * b);
extern GtkTreePath *gtk_tree_path_copy(const GtkTreePath * path);
extern void gtk_tree_path_down(GtkTreePath * path);
extern void gtk_tree_path_free(GtkTreePath * path);
extern gint gtk_tree_path_get_depth(GtkTreePath * path);
extern gint *gtk_tree_path_get_indices(GtkTreePath * path);
extern GType gtk_tree_path_get_type(void);
extern gboolean gtk_tree_path_is_ancestor(GtkTreePath * path,
                                         GtkTreePath * descendant);
extern gboolean gtk_tree_path_is_descendant(GtkTreePath * path,
                                           GtkTreePath * ancestor);
extern GtkTreePath *gtk_tree_path_new(void);
extern GtkTreePath *gtk_tree_path_new_first(void);
extern GtkTreePath *gtk_tree_path_new_from_indices(gint
first_index, ...);
extern GtkTreePath *gtk_tree_path_new_from_string(const gchar *
path);
extern void gtk_tree_path_next(GtkTreePath * path);
extern void gtk_tree_path_prepend_index(GtkTreePath * path, gint
index_);
extern gboolean gtk_tree_path_prev(GtkTreePath * path);
extern gchar *gtk_tree_path_to_string(GtkTreePath * path);
extern gboolean gtk_tree_path_up(GtkTreePath * path);
extern GtkTreeRowReference
*gtk_tree_row_reference_copy(GtkTreeRowReference
                             * reference);

```

```

extern void gtk_tree_row_reference_deleted(GObject * proxy,
                                           GtkTreePath * path);
extern void gtk_tree_row_reference_free(GtkTreeRowReference *
reference);
extern                                     GtkTreeModel
*gtk_tree_row_reference_get_model(GtkTreeRowReference *
reference);
extern                                     GtkTreePath
*gtk_tree_row_reference_get_path(GtkTreeRowReference *
reference);
extern GType gtk_tree_row_reference_get_type(void);
extern void gtk_tree_row_reference_inserted(GObject * proxy,
                                           GtkTreePath * path);
extern                                     GtkTreeRowReference
*gtk_tree_row_reference_new(GtkTreeModel *
model,
                           GtkTreePath * path);
extern                                     GtkTreeRowReference
*gtk_tree_row_reference_new_proxy(GObject *
proxy,
                                  GtkTreeModel *
model,
                                  GtkTreePath *
path);
extern void gtk_tree_row_reference_reordered(GObject * proxy,
                                           GtkTreePath * path,
                                           GtkTreeIter * iter,
                                           gint * new_order);
extern gboolean gtk_tree_row_reference_valid(GtkTreeRowReference *
reference);
extern                                     gint
gtk_tree_selection_count_selected_rows(GtkTreeSelection *
selection);
extern                                     GtkSelectionMode
gtk_tree_selection_get_mode(GtkTreeSelection *
selection);
extern gboolean gtk_tree_selection_get_selected(GtkTreeSelection *
selection,
                                                GtkTreeModel * *model,
                                                GtkTreeIter * iter);
extern                                     GList
*gtk_tree_selection_get_selected_rows(GtkTreeSelection *
selection,
                                      GtkTreeModel * *model);
extern                                     GtkTreeView
*gtk_tree_selection_get_tree_view(GtkTreeSelection *
selection);
extern GType gtk_tree_selection_get_type(void);
extern gpointer gtk_tree_selection_get_user_data(GtkTreeSelection
*
selection);
extern                                     gboolean
gtk_tree_selection_iter_is_selected(GtkTreeSelection *
selection,
                                    GtkTreeIter * iter);
extern                                     gboolean
gtk_tree_selection_path_is_selected(GtkTreeSelection *
selection,
                                    GtkTreePath * path);
extern void gtk_tree_selection_select_all(GtkTreeSelection *
selection);
extern void gtk_tree_selection_select_iter(GtkTreeSelection *
selection,
                                           GtkTreeIter * iter);
extern void gtk_tree_selection_select_path(GtkTreeSelection *
selection,

```

```

                                GtkTreePath * path);
extern void gtk_tree_selection_select_range(GtkTreeSelection *
selection,
                                GtkTreePath * start_path,
                                GtkTreePath * end_path);
extern void gtk_tree_selection_selected_foreach(GtkTreeSelection *
selection,

GtkTreeSelectionForeachFunc
                                func, gpointer data);
extern void gtk_tree_selection_set_mode(GtkTreeSelection *
selection,
                                GtkSelectionMode type);
extern
                                void
gtk_tree_selection_set_select_function(GtkTreeSelection *
selection,
                                GtkTreeSelectionFunc
                                func, gpointer data,
                                GtkDestroyNotify
                                destroy);
extern void gtk_tree_selection_unselect_all(GtkTreeSelection *
selection);
extern void gtk_tree_selection_unselect_iter(GtkTreeSelection *
selection,
                                GtkTreeIter * iter);
extern void gtk_tree_selection_unselect_path(GtkTreeSelection *
selection,
                                GtkTreePath * path);
extern void gtk_tree_selection_unselect_range(GtkTreeSelection *
selection,
                                GtkTreePath * start_path,
                                GtkTreePath * end_path);
extern gboolean gtk_tree_set_row_drag_data(GtkSelectionData *
selection_data,
                                GtkTreeModel * tree_model,
                                GtkTreePath * path);
extern
                                gboolean
gtk_tree_sortable_get_sort_column_id(GtkTreeSortable *
sortable,
                                gint * sort_column_id,
                                GtkSortType * order);
extern GType gtk_tree_sortable_get_type(void);
extern
                                gboolean
gtk_tree_sortable_has_default_sort_func(GtkTreeSortable *
sortable);
extern
                                void
gtk_tree_sortable_set_default_sort_func(GtkTreeSortable *
sortable,
                                GtkTreeIterCompareFunc
                                sort_func,
                                gpointer user_data,
                                GtkDestroyNotify
                                destroy);
extern void gtk_tree_sortable_set_sort_column_id(GtkTreeSortable *
sortable,
                                gint sort_column_id,
                                GtkSortType order);
extern void gtk_tree_sortable_set_sort_func(GtkTreeSortable *
sortable,
                                gint sort_column_id,
                                GtkTreeIterCompareFunc
                                sort_func, gpointer user_data,
                                GtkDestroyNotify destroy);
extern void gtk_tree_sortable_sort_column_changed(GtkTreeSortable
*
                                sortable);

```

```

extern void gtk_tree_store_append(GtkTreeStore * tree_store,
                                GtkTreeIter * iter,
                                GtkTreeIter * parent);
extern void gtk_tree_store_clear(GtkTreeStore * tree_store);
extern GType gtk_tree_store_get_type(void);
extern void gtk_tree_store_insert(GtkTreeStore * tree_store,
                                GtkTreeIter * iter, GtkTreeIter *
parent,
                                gint position);
extern void gtk_tree_store_insert_after(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * parent,
                                       GtkTreeIter * sibling);
extern void gtk_tree_store_insert_before(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * parent,
                                       GtkTreeIter * sibling);
extern void gtk_tree_store_insert_with_values(GtkTreeStore *
tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * parent,
                                       gint position, ...);
extern void gtk_tree_store_insert_with_valuesv(GtkTreeStore *
tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * parent,
                                       gint position,
                                       gint * columns,
                                       GValue * values,
                                       gint n_values);
extern gboolean gtk_tree_store_is_ancestor(GtkTreeStore *
tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * descendant);
extern gint gtk_tree_store_iter_depth(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter);
extern gboolean gtk_tree_store_iter_is_valid(GtkTreeStore *
tree_store,
                                       GtkTreeIter * iter);
extern void gtk_tree_store_move_after(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * position);
extern void gtk_tree_store_move_before(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter,
                                       GtkTreeIter * position);
extern GtkTreeStore *gtk_tree_store_new(gint n_columns, ...);
extern GtkTreeStore *gtk_tree_store_newv(gint n_columns, GType *
types);
extern void gtk_tree_store_prepend(GtkTreeStore * tree_store,
                                   GtkTreeIter * iter,
                                   GtkTreeIter * parent);
extern gboolean gtk_tree_store_remove(GtkTreeStore * tree_store,
                                      GtkTreeIter * iter);
extern void gtk_tree_store_reorder(GtkTreeStore * tree_store,
                                   GtkTreeIter * parent, gint *
new_order);
extern void gtk_tree_store_set(GtkTreeStore * tree_store,
                              GtkTreeIter * iter, ...);
extern void gtk_tree_store_set_column_types(GtkTreeStore *
tree_store,
                              gint n_columns, GType * types);
extern void gtk_tree_store_set_valist(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter,
                                       va_list var_args);
extern void gtk_tree_store_set_value(GtkTreeStore * tree_store,
                                       GtkTreeIter * iter, gint column,

```

```

                                GValue * value);
extern void gtk_tree_store_swap(GtkTreeStore * tree_store,
GtkTreeIter * a,
                                GtkTreeIter * b);
extern gint gtk_tree_view_append_column(GtkTreeView * tree_view,
                                GtkTreeViewColumn * column);
extern void gtk_tree_view_collapse_all(GtkTreeView * tree_view);
extern gboolean gtk_tree_view_collapse_row(GtkTreeView * tree_view,
                                GtkTreePath * path);
extern void gtk_tree_view_column_add_attribute(GtkTreeViewColumn *
                                tree_column,
                                GtkCellRenderer *
                                cell_renderer,
                                const gchar * attribute,
                                gint column);

extern
                                gboolean
gtk_tree_view_column_cell_get_position(GtkTreeViewColumn *
                                tree_column,
                                GtkCellRenderer *
                                cell_renderer,
                                gint * start_pos,
                                gint * width);

extern void gtk_tree_view_column_cell_get_size(GtkTreeViewColumn *
                                tree_column,
                                const GdkRectangle *
                                cell_area, gint * x_offset,
                                gint * y_offset,
                                gint * width,
                                gint * height);

extern
                                gboolean
gtk_tree_view_column_cell_is_visible(GtkTreeViewColumn *
                                tree_column);

extern
                                void
gtk_tree_view_column_cell_set_cell_data(GtkTreeViewColumn *
                                tree_column,
                                GtkTreeModel *
                                tree_model,
                                GtkTreeIter * iter,
                                gboolean is_expander,
                                gboolean is_expanded);

extern void gtk_tree_view_column_clear(GtkTreeViewColumn *
tree_column);
extern
                                void
gtk_tree_view_column_clear_attributes(GtkTreeViewColumn *
                                tree_column,
                                GtkCellRenderer *
                                cell_renderer);

extern void gtk_tree_view_column_clicked(GtkTreeViewColumn *
tree_column);
extern void gtk_tree_view_column_focus_cell(GtkTreeViewColumn *
                                tree_column,
                                GtkCellRenderer * cell);
extern gfloat gtk_tree_view_column_get_alignment(GtkTreeViewColumn
*
                                tree_column);

extern
                                GList
*gtk_tree_view_column_get_cell_renderers(GtkTreeViewColumn *
                                tree_column);

extern
                                gboolean
gtk_tree_view_column_get_clickable(GtkTreeViewColumn *
                                tree_column);

extern gboolean gtk_tree_view_column_get_expand(GtkTreeViewColumn
*
                                tree_column);

extern gint gtk_tree_view_column_get_fixed_width(GtkTreeViewColumn
*

```

```

        tree_column);
extern gint gtk_tree_view_column_get_max_width(GtkTreeViewColumn *
        tree_column);
extern gint gtk_tree_view_column_get_min_width(GtkTreeViewColumn *
        tree_column);

extern                                gboolean
gtk_tree_view_column_get_reorderable(GtkTreeViewColumn *
        tree_column);

extern                                gboolean
gtk_tree_view_column_get_resizable(GtkTreeViewColumn *
        tree_column);

extern GtkTreeViewColumnSizing
gtk_tree_view_column_get_sizing(GtkTreeViewColumn * tree_column);
extern                                gint
gtk_tree_view_column_get_sort_column_id(GtkTreeViewColumn *
        tree_column);

extern                                gboolean
gtk_tree_view_column_get_sort_indicator(GtkTreeViewColumn *
        tree_column);

extern                                GtkSortType
gtk_tree_view_column_get_sort_order(GtkTreeViewColumn *
        tree_column);
extern gint gtk_tree_view_column_get_spacing(GtkTreeViewColumn *
        tree_column);

extern                                const          gchar
*gtk_tree_view_column_get_title(GtkTreeViewColumn *
        tree_column);

extern GType gtk_tree_view_column_get_type(void);
extern gboolean gtk_tree_view_column_get_visible(GtkTreeViewColumn
*
        tree_column);

extern                                GtkWidget
*gtk_tree_view_column_get_widget(GtkTreeViewColumn *
        tree_column);
extern gint gtk_tree_view_column_get_width(GtkTreeViewColumn *
        tree_column);
extern GtkTreeViewColumn *gtk_tree_view_column_new(void);
extern                                GtkTreeViewColumn
*gtk_tree_view_column_new_with_attributes(const
        gchar *
        title,

        GtkCellRenderer
        * cell,
        ...);
extern void gtk_tree_view_column_pack_end(GtkTreeViewColumn *
tree_column,
        GtkCellRenderer * cell,
        gboolean expand);
extern void gtk_tree_view_column_pack_start(GtkTreeViewColumn *
        tree_column,
        GtkCellRenderer * cell,
        gboolean expand);
extern void gtk_tree_view_column_queue_resize(GtkTreeViewColumn *
        tree_column);
extern void gtk_tree_view_column_set_alignment(GtkTreeViewColumn *
        tree_column, gfloat xalign);
extern void gtk_tree_view_column_set_attributes(GtkTreeViewColumn
*
        tree_column,
        GtkCellRenderer *
        cell_renderer, ...);

extern                                void
gtk_tree_view_column_set_cell_data_func(GtkTreeViewColumn *
        tree_column,
        GtkCellRenderer *

```

```

        cell_renderer,
        GtkTreeCellDataFunc
        func,
        gpointer func_data,
        GtkDestroyNotify
        destroy);
extern void gtk_tree_view_column_set_clickable(GtkTreeViewColumn *
        tree_column,
        gboolean clickable);
extern void gtk_tree_view_column_set_expand(GtkTreeViewColumn *
        tree_column, gboolean expand);
extern void gtk_tree_view_column_set_fixed_width(GtkTreeViewColumn
*
        tree_column,
        gint fixed_width);
extern void gtk_tree_view_column_set_max_width(GtkTreeViewColumn *
        tree_column,
        gint max_width);
extern void gtk_tree_view_column_set_min_width(GtkTreeViewColumn *
        tree_column,
        gint min_width);
extern void gtk_tree_view_column_set_reorderable(GtkTreeViewColumn
*
        tree_column,
        gboolean reorderable);
extern void gtk_tree_view_column_set_resizable(GtkTreeViewColumn *
        tree_column,
        gboolean resizable);
extern void gtk_tree_view_column_set_sizing(GtkTreeViewColumn *
        tree_column,
        GtkTreeViewColumnSizing type);
extern
        void
        gtk_tree_view_column_set_sort_column_id(GtkTreeViewColumn *
        tree_column,
        gint sort_column_id);
extern
        void
        gtk_tree_view_column_set_sort_indicator(GtkTreeViewColumn *
        tree_column,
        gboolean setting);
extern void gtk_tree_view_column_set_sort_order(GtkTreeViewColumn
*
        tree_column,
        GtkSortType order);
extern void gtk_tree_view_column_set_spacing(GtkTreeViewColumn *
        tree_column, gint spacing);
extern void gtk_tree_view_column_set_title(GtkTreeViewColumn *
        tree_column,
        const gchar * title);
extern void gtk_tree_view_column_set_visible(GtkTreeViewColumn *
        tree_column,
        gboolean visible);
extern void gtk_tree_view_column_set_widget(GtkTreeViewColumn *
        tree_column,
        GtkWidget * widget);
extern GType gtk_tree_view_column_sizing_get_type(void);
extern void gtk_tree_view_columns_autosize(GtkTreeView *
        tree_view);
extern GdkPixmap *gtk_tree_view_create_row_drag_icon(GtkTreeView *
        tree_view,
        GtkTreePath * path);
extern GType gtk_tree_view_drop_position_get_type(void);
extern void gtk_tree_view_enable_model_drag_dest(GtkTreeView *
        tree_view,
        const GtkTargetEntry *
        targets, gint n_targets,
        GdkDragAction actions);

```



```

extern void gtk_tree_view_enable_model_drag_source(GtkTreeView *
tree_view,

GdkModifierType
start_button_mask,
const GtkTargetEntry *
targets, gint n_targets,
GdkDragAction actions);

extern void gtk_tree_view_expand_all(GtkTreeView * tree_view);
extern gboolean gtk_tree_view_expand_row(GtkTreeView * tree_view,
GtkTreePath * path,
gboolean open_all);
extern void gtk_tree_view_expand_to_path(GtkTreeView * tree_view,
GtkTreePath * path);
extern void gtk_tree_view_get_background_area(GtkTreeView *
tree_view,

GtkTreePath * path,
GtkTreeViewColumn * column,
GdkRectangle * rect);
extern GdkWindow *gtk_tree_view_get_bin_window(GtkTreeView *
tree_view);
extern void gtk_tree_view_get_cell_area(GtkTreeView * tree_view,
GtkTreePath * path,
GtkTreeViewColumn * column,
GdkRectangle * rect);
extern GtkTreeViewColumn *gtk_tree_view_get_column(GtkTreeView *
tree_view,

gint n);
extern GList *gtk_tree_view_get_columns(GtkTreeView * tree_view);
extern void gtk_tree_view_get_cursor(GtkTreeView * tree_view,
GtkTreePath * *path,
GtkTreeViewColumn * *focus_column);
extern gboolean gtk_tree_view_get_dest_row_at_pos(GtkTreeView *
tree_view,

gint drag_x, gint drag_y,
GtkTreePath * *path,
GtkTreeViewDropPosition

*
pos);
extern void gtk_tree_view_get_drag_dest_row(GtkTreeView *
tree_view,

GtkTreePath * *path,
GtkTreeViewDropPosition *
pos);
extern gboolean gtk_tree_view_get_enable_search(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_enable_tree_lines(GtkTreeView *
tree_view);
extern GtkTreeViewColumn
*gtk_tree_view_get_expander_column(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_fixed_height_mode(GtkTreeView *
tree_view);
extern GtkTreeViewGridLines
gtk_tree_view_get_grid_lines(GtkTreeView *
tree_view);
extern GtkAdjustment *gtk_tree_view_get_hadjustment(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_headers_clickable(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_headers_visible(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_hover_expand(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_hover_selection(GtkTreeView *
tree_view);

```

```

extern GtkTreeModel *gtk_tree_view_get_model(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_path_at_pos(GtkTreeView *
tree_view,
                                gint x, gint y,
                                GtkTreePath * *path,
                                GtkTreeViewColumn * *column,
                                gint * cell_x,
                                gint * cell_y);
extern gboolean gtk_tree_view_get_reorderable(GtkTreeView *
tree_view);
extern GtkTreeViewRowSeparatorFunc
gtk_tree_view_get_row_separator_func(GtkTreeView * tree_view);
extern gboolean gtk_tree_view_get_rubber_banding(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_rules_hint(GtkTreeView *
tree_view);
extern gint gtk_tree_view_get_search_column(GtkTreeView *
tree_view);
extern GtkEntry *gtk_tree_view_get_search_entry(GtkTreeView *
tree_view);
extern GtkTreeViewSearchEqualFunc
gtk_tree_view_get_search_equal_func(GtkTreeView * tree_view);
extern GtkTreeViewSearchPositionFunc
gtk_tree_view_get_search_position_func(GtkTreeView * tree_view);
extern GtkTreeSelection *gtk_tree_view_get_selection(GtkTreeView *
tree_view);

extern GType gtk_tree_view_get_type(void);
extern GtkAdjustment *gtk_tree_view_get_vadjustment(GtkTreeView *
tree_view);
extern gboolean gtk_tree_view_get_visible_range(GtkTreeView *
tree_view,
                                GtkTreePath * *start_path,
                                GtkTreePath * *end_path);
extern void gtk_tree_view_get_visible_rect(GtkTreeView * tree_view,
                                GdkRectangle * visible_rect);
extern GType gtk_tree_view_grid_lines_get_type(void);
extern gint gtk_tree_view_insert_column(GtkTreeView * tree_view,
                                GtkTreeViewColumn * column,
                                gint position);
extern
                                gint
gtk_tree_view_insert_column_with_attributes(GtkTreeView *
tree_view,
                                gint position,
                                const gchar *
title,
                                GtkCellRenderer *
cell, ...);
extern gint gtk_tree_view_insert_column_with_data_func(GtkTreeView
tree_view,
                                gint position,
                                const gchar * title,
                                GtkCellRenderer *
cell,
                                GtkTreeCellDataFunc
func, gpointer data,
                                GDestroyNotify
dnotify);
extern void gtk_tree_view_map_expanded_rows(GtkTreeView *
tree_view,
                                GtkTreeViewMappingFunc func,
                                gpointer data);
extern GType gtk_tree_view_mode_get_type(void);
extern void gtk_tree_view_move_column_after(GtkTreeView *
tree_view,

```

```

        GtkTreeViewColumn * column,
        GtkTreeViewColumn *
        base_column);
extern GtkWidget *gtk_tree_view_new(void);
extern GtkWidget *gtk_tree_view_new_with_model(GtkTreeModel *
model);
extern gint gtk_tree_view_remove_column(GtkTreeView * tree_view,
        GtkTreeViewColumn * column);
extern void gtk_tree_view_row_activated(GtkTreeView * tree_view,
        GtkTreePath * path,
        GtkTreeViewColumn * column);
extern gboolean gtk_tree_view_row_expanded(GtkTreeView * tree_view,
        GtkTreePath * path);
extern void gtk_tree_view_scroll_to_cell(GtkTreeView * tree_view,
        GtkTreePath * path,
        GtkTreeViewColumn * column,
        gboolean use_align,
        gfloat row_align,
        gfloat col_align);
extern void gtk_tree_view_scroll_to_point(GtkTreeView * tree_view,
        gint tree_x, gint tree_y);
extern void gtk_tree_view_set_column_drag_function(GtkTreeView *
tree_view,

GtkTreeViewColumnDropFunc
        func,
        gpointer user_data,
        GtkDestroyNotify
        destroy);
extern void gtk_tree_view_set_cursor(GtkTreeView * tree_view,
        GtkTreePath * path,
        GtkTreeViewColumn * focus_column,
        gboolean start_editing);
extern void gtk_tree_view_set_cursor_on_cell(GtkTreeView *
tree_view,

        GtkTreePath * path,
        GtkTreeViewColumn *
        focus_column,
        GtkCellRenderer * focus_cell,
        gboolean start_editing);
extern void gtk_tree_view_set_destroy_count_func(GtkTreeView *
tree_view,

        GtkTreeDestroyCountFunc
        func, gpointer data,
        GtkDestroyNotify destroy);
extern void gtk_tree_view_set_drag_dest_row(GtkTreeView *
tree_view,

        GtkTreePath * path,
        GtkTreeViewDropPosition pos);
extern void gtk_tree_view_set_enable_search(GtkTreeView *
tree_view,

        gboolean enable_search);
extern void gtk_tree_view_set_enable_tree_lines(GtkTreeView *
tree_view,

        gboolean enabled);
extern void gtk_tree_view_set_expander_column(GtkTreeView *
tree_view,

        GtkTreeViewColumn * column);
extern void gtk_tree_view_set_fixed_height_mode(GtkTreeView *
tree_view,

        gboolean enable);
extern void gtk_tree_view_set_grid_lines(GtkTreeView * tree_view,
        GtkTreeViewGridLines
        grid_lines);
extern void gtk_tree_view_set_hadjustment(GtkTreeView * tree_view,
        GtkAdjustment * adjustment);

```

```

extern void gtk_tree_view_set_headers_clickable(GtkTreeView *
tree_view,
                                           gboolean setting);
extern void gtk_tree_view_set_headers_visible(GtkTreeView *
tree_view,
                                           gboolean headers_visible);
extern void gtk_tree_view_set_hover_expand(GtkTreeView * tree_view,
                                           gboolean expand);
extern void gtk_tree_view_set_hover_selection(GtkTreeView *
tree_view,
                                           gboolean hover);
extern void gtk_tree_view_set_model(GtkTreeView * tree_view,
                                   GtkTreeModel * model);
extern void gtk_tree_view_set_reorderable(GtkTreeView * tree_view,
                                           gboolean reorderable);
extern void gtk_tree_view_set_row_separator_func(GtkTreeView *
tree_view,

GtkTreeViewRowSeparatorFunc
                                           func, gpointer data,
                                           GtkDestroyNotify destroy);
extern void gtk_tree_view_set_rubber_banding(GtkTreeView *
tree_view,
                                           gboolean enable);
extern void gtk_tree_view_set_rules_hint(GtkTreeView * tree_view,
                                           gboolean setting);
extern void gtk_tree_view_set_search_column(GtkTreeView *
tree_view,
                                           gint column);
extern void gtk_tree_view_set_search_entry(GtkTreeView * tree_view,
                                           GtkEntry * entry);
extern void gtk_tree_view_set_search_equal_func(GtkTreeView *
tree_view,

GtkTreeViewSearchEqualFunc
                                           search_equal_func,
                                           gpointer search_user_data,
                                           GtkDestroyNotify
                                           search_destroy);
extern void gtk_tree_view_set_search_position_func(GtkTreeView *
tree_view,

GtkTreeViewSearchPositionFunc
                                           func, gpointer data,
                                           GDestroyNotify destroy);
extern void gtk_tree_view_set_vadjustment(GtkTreeView * tree_view,
                                           GtkAdjustment * adjustment);
extern void gtk_tree_view_tree_to_widget_coords(GtkTreeView *
tree_view,
                                           gint tx, gint ty,
                                           gint * wx, gint * wy);
extern void gtk_tree_view_unset_rows_drag_dest(GtkTreeView *
tree_view);
extern void gtk_tree_view_unset_rows_drag_source(GtkTreeView *
tree_view);
extern void gtk_tree_view_widget_to_tree_coords(GtkTreeView *
tree_view,
                                           gint wx, gint wy,
                                           gint * tx, gint * ty);
extern gboolean gtk_true(void);
extern gpointer gtk_type_class(GtkType type);
extern void gtk_ui_manager_add_ui(GtkUIManager * self, guint
merge_id,
                                           const gchar * path, const gchar * name,
                                           const gchar * action,

```

```

                                GtkUIManagerItemType type, gboolean
top);
extern guint gtk_ui_manager_add_ui_from_file(GtkUIManager * self,
                                              const gchar * filename,
                                              GError * *error);
extern guint gtk_ui_manager_add_ui_from_string(GtkUIManager * self,
                                              const gchar * buffer,
                                              gssize length,
                                              GError * *error);
extern void gtk_ui_manager_ensure_update(GtkUIManager * self);
extern GtkAccelGroup *gtk_ui_manager_get_accel_group(GtkUIManager
* self);
extern GtkAction *gtk_ui_manager_get_action(GtkUIManager * self,
                                              const gchar * path);
extern GList *gtk_ui_manager_get_action_groups(GtkUIManager *
self);
extern gboolean gtk_ui_manager_get_add_tearoffs(GtkUIManager *
self);
extern GSList *gtk_ui_manager_get_toplevels(GtkUIManager * self,
                                              GtkUIManagerItemType types);
extern GType gtk_ui_manager_get_type(void);
extern gchar *gtk_ui_manager_get_ui(GtkUIManager * self);
extern GtkWidget *gtk_ui_manager_get_widget(GtkUIManager * self,
                                              const gchar * path);
extern void gtk_ui_manager_insert_action_group(GtkUIManager * self,
                                              GtkActionGroup *
                                              action_group, gint pos);
extern GType gtk_ui_manager_item_type_get_type(void);
extern GtkUIManager *gtk_ui_manager_new(void);
extern guint gtk_ui_manager_new_merge_id(GtkUIManager * self);
extern void gtk_ui_manager_remove_action_group(GtkUIManager * self,
                                              GtkActionGroup *
                                              action_group);
extern void gtk_ui_manager_remove_ui(GtkUIManager * self, guint
merge_id);
extern void gtk_ui_manager_set_add_tearoffs(GtkUIManager * self,
                                              gboolean add_tearoffs);
extern GType gtk_unit_get_type(void);
extern GType gtk_update_type_get_type(void);
extern GType gtk_vbox_get_type(void);
extern GtkWidget *gtk_vbox_new(gboolean homogeneous, gint spacing);
extern GType gtk_vbutton_box_get_type(void);
extern GtkWidget *gtk_vbutton_box_new(void);
extern GtkAdjustment *gtk_viewport_get_hadjustment(GtkViewport *
viewport);
extern GtkShadowType gtk_viewport_get_shadow_type(GtkViewport *
viewport);
extern GType gtk_viewport_get_type(void);
extern GtkAdjustment *gtk_viewport_get_vadjustment(GtkViewport *
viewport);
extern GtkWidget *gtk_viewport_new(GtkAdjustment * hadjustment,
                                    GtkAdjustment * vadjustment);
extern void gtk_viewport_set_hadjustment(GtkViewport * viewport,
                                          GtkAdjustment * adjustment);
extern void gtk_viewport_set_shadow_type(GtkViewport * viewport,
                                          GtkShadowType type);
extern void gtk_viewport_set_vadjustment(GtkViewport * viewport,
                                          GtkAdjustment * adjustment);
extern GType gtk_visibility_get_type(void);
extern GType gtk_vpaned_get_type(void);
extern GtkWidget *gtk_vpaned_new(void);
extern GType gtk_vruler_get_type(void);
extern GtkWidget *gtk_vruler_new(void);
extern GType gtk_vscale_get_type(void);
extern GtkWidget *gtk_vscale_new(GtkAdjustment * adjustment);

```

```

extern GtkWidget *gtk_vscale_new_with_range(gdouble min, gdouble
max,
                                         gdouble step);
extern GType gtk_vscrollbar_get_type(void);
extern GtkWidget *gtk_vscrollbar_new(GtkAdjustment * adjustment);
extern GType gtk_vseparator_get_type(void);
extern GtkWidget *gtk_vseparator_new(void);
extern gboolean gtk_widget_activate(GtkWidget * widget);
extern void gtk_widget_add_accelerator(GtkWidget * widget,
                                       const gchar * accel_signal,
                                       GtkAccelGroup * accel_group,
                                       guint accel_key,
                                       GdkModifierType accel_mods,
                                       GtkAccelFlags accel_flags);
extern void gtk_widget_add_events(GtkWidget * widget, gint events);
extern void gtk_widget_add_mnemonic_label(GtkWidget * widget,
                                           GtkWidget * label);
extern gboolean gtk_widget_can_activate_accel(GtkWidget * widget,
                                              guint signal_id);
extern gboolean gtk_widget_child_focus(GtkWidget * widget,
                                       GtkDirectionType direction);
extern void gtk_widget_child_notify(GtkWidget * widget,
                                     const gchar * child_property);
extern
                                           GParamSpec
*gtk_widget_class_find_style_property(GtkWidgetClass *
                                     klass,
                                     const gchar *
                                     property_name);
extern void gtk_widget_class_install_style_property(GtkWidgetClass
* klass,
                                           GParamSpec * pspec);
extern
                                           void
gtk_widget_class_install_style_property_parser(GtkWidgetClass *
                                             klass,
                                             GParamSpec *
                                             pspec,
                                             GtkRcPropertyParser
                                             parser);
extern
                                           GParamSpec
**gtk_widget_class_list_style_properties(GtkWidgetClass *
                                         klass,
                                         guint *
                                         n_properties);
extern void gtk_widget_class_path(GtkWidget * widget, guint *
path_length,
                                  gchar * *path, gchar * *path_reversed);
extern PangoContext *gtk_widget_create_pango_context(GtkWidget *
widget);
extern PangoLayout *gtk_widget_create_pango_layout(GtkWidget *
widget,
                                                    const gchar * text);
extern void gtk_widget_destroy(GtkWidget * widget);
extern void gtk_widget_destroyed(GtkWidget * widget,
                                  GtkWidget * *widget_pointer);
extern void gtk_widget_ensure_style(GtkWidget * widget);
extern gboolean gtk_widget_event(GtkWidget * widget, GdkEvent *
event);
extern GType gtk_widget_flags_get_type(void);
extern void gtk_widget_freeze_child_notify(GtkWidget * widget);
extern AtkObject *gtk_widget_get_accessible(GtkWidget * widget);
extern GtkAction *gtk_widget_get_action(GtkWidget * widget);
extern GtkWidget *gtk_widget_get_ancestor(GtkWidget * widget,
                                           GType widget_type);
extern void gtk_widget_get_child_requisition(GtkWidget * widget,

```

```

requisition);
extern gboolean gtk_widget_get_child_visible(GtkWidget * widget);
extern GtkClipboard *gtk_widget_get_clipboard(GtkWidget * widget,
                                              GdkAtom selection);
extern GdkColormap *gtk_widget_get_colormap(GtkWidget * widget);
extern gchar *gtk_widget_get_composite_name(GtkWidget * widget);
extern GdkColormap *gtk_widget_get_default_colormap(void);
extern GtkTextDirection gtk_widget_get_default_direction(void);
extern GtkStyle *gtk_widget_get_default_style(void);
extern GdkVisual *gtk_widget_get_default_visual(void);
extern GtkTextDirection gtk_widget_get_direction(GtkWidget *
widget);
extern GdkDisplay *gtk_widget_get_display(GtkWidget * widget);
extern gint gtk_widget_get_events(GtkWidget * widget);
extern GdkExtensionMode gtk_widget_get_extension_events(GtkWidget *
widget);
extern GtkRcStyle *gtk_widget_get_modifier_style(GtkWidget *
widget);
extern const gchar *gtk_widget_get_name(GtkWidget * widget);
extern gboolean gtk_widget_get_no_show_all(GtkWidget * widget);
extern PangoContext *gtk_widget_get_pango_context(GtkWidget *
widget);
extern GtkWidget *gtk_widget_get_parent(GtkWidget * widget);
extern GdkWindow *gtk_widget_get_parent_window(GtkWidget * widget);
extern void gtk_widget_get_pointer(GtkWidget * widget, gint * x,
gint * y);
extern GdkWindow *gtk_widget_get_root_window(GtkWidget * widget);
extern GdkScreen *gtk_widget_get_screen(GtkWidget * widget);
extern GtkSettings *gtk_widget_get_settings(GtkWidget * widget);
extern void gtk_widget_get_size_request(GtkWidget * widget, gint *
width,
gint * height);
extern GtkStyle *gtk_widget_get_style(GtkWidget * widget);
extern GtkWidget *gtk_widget_get_toplevel(GtkWidget * widget);
extern GType gtk_widget_get_type(void);
extern GdkVisual *gtk_widget_get_visual(GtkWidget * widget);
extern void gtk_widget_grab_default(GtkWidget * widget);
extern void gtk_widget_grab_focus(GtkWidget * widget);
extern gboolean gtk_widget_has_screen(GtkWidget * widget);
extern GType gtk_widget_help_type_get_type(void);
extern void gtk_widget_hide(GtkWidget * widget);
extern void gtk_widget_hide_all(GtkWidget * widget);
extern gboolean gtk_widget_hide_on_delete(GtkWidget * widget);
extern void gtk_widget_input_shape_combine_mask(GtkWidget * widget,
GdkBitmap * shape_mask,
gint offset_x,
gint offset_y);
extern gboolean gtk_widget_intersect(GtkWidget * widget,
const GdkRectangle * area,
GdkRectangle * intersection);
extern gboolean gtk_widget_is_ancestor(GtkWidget * widget,
GtkWidget * ancestor);
extern gboolean gtk_widget_is_composited(GtkWidget * widget);
extern gboolean gtk_widget_is_focus(GtkWidget * widget);
extern GList *gtk_widget_list_accel_closures(GtkWidget * widget);
extern GList *gtk_widget_list_mnemonic_labels(GtkWidget * widget);
extern void gtk_widget_map(GtkWidget * widget);
extern gboolean gtk_widget_mnemonic_activate(GtkWidget * widget,
gboolean group_cycling);
extern void gtk_widget_modify_base(GtkWidget * widget, GtkStateType
state,
const GdkColor * color);
extern void gtk_widget_modify_bg(GtkWidget * widget, GtkStateType
state,

```

```

        const GdkColor * color);
extern void gtk_widget_modify_fg(GtkWidget * widget, GtkStateType
state,
        const GdkColor * color);
extern void gtk_widget_modify_font(GtkWidget * widget,
        PangoFontDescription * font_desc);
extern void gtk_widget_modify_style(GtkWidget * widget,
        GtkRcStyle * style);
extern void gtk_widget_modify_text(GtkWidget * widget, GtkStateType
state,
        const GdkColor * color);
extern GtkWidget *gtk_widget_new(GType type,
        const gchar * first_property_name, ...);
extern void gtk_widget_path(GtkWidget * widget, guint * path_length,
        gchar * *path, gchar * *path_reversed);
extern void gtk_widget_pop_colormap(void);
extern void gtk_widget_pop_composite_child(void);
extern void gtk_widget_push_colormap(GdkColormap * cmap);
extern void gtk_widget_push_composite_child(void);
extern void gtk_widget_queue_draw(GtkWidget * widget);
extern void gtk_widget_queue_draw_area(GtkWidget * widget, gint x,
gint y,
        gint width, gint height);
extern void gtk_widget_queue_resize(GtkWidget * widget);
extern void gtk_widget_queue_resize_no_redraw(GtkWidget * widget);
extern void gtk_widget_realize(GtkWidget * widget);
extern GtkWidget *gtk_widget_ref(GtkWidget * widget);
extern GdkRegion *gtk_widget_region_intersect(GtkWidget * widget,
        const GdkRegion * region);
extern gboolean gtk_widget_remove_accelerator(GtkWidget * widget,
        GtkAccelGroup * accel_group,
        guint accel_key,
        GdkModifierType accel_mods);
extern void gtk_widget_remove_mnemonic_label(GtkWidget * widget,
        GtkWidget * label);
extern GdkPixbuf *gtk_widget_render_icon(GtkWidget * widget,
        const gchar * stock_id,
        GtkIconSize size,
        const gchar * detail);
extern void gtk_widget_reparent(GtkWidget * widget,
        GtkWidget * new_parent);
extern void gtk_widget_reset_rc_styles(GtkWidget * widget);
extern void gtk_widget_reset_shapes(GtkWidget * widget);
extern gint gtk_widget_send_expose(GtkWidget * widget, GdkEvent *
event);
extern void gtk_widget_set_accel_path(GtkWidget * widget,
        const gchar * accel_path,
        GtkAccelGroup * accel_group);
extern void gtk_widget_set_app_paintable(GtkWidget * widget,
        gboolean app_paintable);
extern void gtk_widget_set_child_visible(GtkWidget * widget,
        gboolean is_visible);
extern void gtk_widget_set_colormap(GtkWidget * widget,
        GdkColormap * colormap);
extern void gtk_widget_set_composite_name(GtkWidget * widget,
        const gchar * name);
extern void gtk_widget_set_default_colormap(GdkColormap *
colormap);
extern void gtk_widget_set_default_direction(GtkTextDirection dir);
extern void gtk_widget_set_direction(GtkWidget * widget,
        GtkTextDirection dir);
extern void gtk_widget_set_double_buffered(GtkWidget * widget,
        gboolean double_buffered);
extern void gtk_widget_set_events(GtkWidget * widget, gint events);
extern void gtk_widget_set_extension_events(GtkWidget * widget,
        GdkExtensionMode mode);

```



```

extern void gtk_widget_set_name(GtkWidget * widget, const gchar *
name);
extern void gtk_widget_set_no_show_all(GtkWidget * widget,
gboolean no_show_all);
extern void gtk_widget_set_parent(GtkWidget * widget, GtkWidget *
parent);
extern void gtk_widget_set_parent_window(GtkWidget * widget,
GdkWindow * parent_window);
extern void gtk_widget_set_redraw_on_allocate(GtkWidget * widget,
gboolean
redraw_on_allocate);
extern gboolean gtk_widget_set_scroll_adjustments(GtkWidget *
widget,
GtkAdjustment *
hadjustment,
GtkAdjustment *
vadjustment);
extern void gtk_widget_set_sensitive(GtkWidget * widget,
gboolean sensitive);
extern void gtk_widget_set_size_request(GtkWidget * widget, gint
width,
gint height);
extern void gtk_widget_set_state(GtkWidget * widget, GtkStateType
state);
extern void gtk_widget_set_style(GtkWidget * widget, GtkStyle *
style);
extern void gtk_widget_shape_combine_mask(GtkWidget * widget,
GdkBitmap * shape_mask,
gint offset_x, gint offset_y);
extern void gtk_widget_show(GtkWidget * widget);
extern void gtk_widget_show_all(GtkWidget * widget);
extern void gtk_widget_show_now(GtkWidget * widget);
extern void gtk_widget_size_allocate(GtkWidget * widget,
GtkAllocation * allocation);
extern void gtk_widget_size_request(GtkWidget * widget,
GtkRequisition * requisition);
extern void gtk_widget_style_get(GtkWidget * widget,
const gchar * first_property_name, ...);
extern void gtk_widget_style_get_property(GtkWidget * widget,
const gchar * property_name,
GValue * value);
extern void gtk_widget_style_get_valist(GtkWidget * widget,
const gchar * first_property_name,
va_list var_args);
extern void gtk_widget_thaw_child_notify(GtkWidget * widget);
extern gboolean gtk_widget_translate_coordinates(GtkWidget *
src_widget,
GtkWidget * dest_widget,
gint src_x, gint src_y,
gint * dest_x,
gint * dest_y);
extern void gtk_widget_unmap(GtkWidget * widget);
extern void gtk_widget_unparent(GtkWidget * widget);
extern void gtk_widget_unrealize(GtkWidget * widget);
extern void gtk_widget_unref(GtkWidget * widget);
extern gboolean gtk_window_activate_default(GtkWindow * window);
extern gboolean gtk_window_activate_focus(GtkWindow * window);
extern gboolean gtk_window_activate_key(GtkWindow * window,
GdkEventKey * event);
extern void gtk_window_add_accel_group(GtkWindow * window,
GtkAccelGroup * accel_group);
extern void gtk_window_add_mnemonic(GtkWindow * window, guint
keyval,
GtkWidget * target);
extern void gtk_window_begin_move_drag(GtkWindow * window, gint
button,

```

```

        gint root_x, gint root_y,
        guint32 timestamp);
extern void gtk_window_begin_resize_drag(GtkWindow * window,
        GdkWindowEdge edge, gint button,
        gint root_x, gint root_y,
        guint32 timestamp);
extern void gtk_window_deiconify(GtkWindow * window);
extern void gtk_window_fullscreen(GtkWindow * window);
extern gboolean gtk_window_get_accept_focus(GtkWindow * window);
extern gboolean gtk_window_get_decorated(GtkWindow * window);
extern GList *gtk_window_get_default_icon_list(void);
extern void gtk_window_get_default_size(GtkWindow * window, gint *
width,
        gint * height);
extern gboolean gtk_window_get_deletable(GtkWindow * window);
extern gboolean gtk_window_get_destroy_with_parent(GtkWindow *
window);
extern GtkWidget *gtk_window_get_focus(GtkWindow * window);
extern gboolean gtk_window_get_focus_on_map(GtkWindow * window);
extern void gtk_window_get_frame_dimensions(GtkWindow * window,
        gint * left, gint * top,
        gint * right, gint * bottom);
extern GdkGravity gtk_window_get_gravity(GtkWindow * window);
extern GtkWidgetGroup *gtk_window_get_group(GtkWindow * window);
extern gboolean gtk_window_get_has_frame(GtkWindow * window);
extern GdkPixbuf *gtk_window_get_icon(GtkWindow * window);
extern GList *gtk_window_get_icon_list(GtkWindow * window);
extern const gchar *gtk_window_get_icon_name(GtkWindow * window);
extern GdkModifierType gtk_window_get_mnemonic_modifier(GtkWindow
*
        window);
extern gboolean gtk_window_get_modal(GtkWindow * window);
extern void gtk_window_get_position(GtkWindow * window, gint *
root_x,
        gint * root_y);
extern gboolean gtk_window_get_resizable(GtkWindow * window);
extern const gchar *gtk_window_get_role(GtkWindow * window);
extern GdkScreen *gtk_window_get_screen(GtkWindow * window);
extern void gtk_window_get_size(GtkWindow * window, gint * width,
        gint * height);
extern gboolean gtk_window_get_skip_pager_hint(GtkWindow * window);
extern gboolean gtk_window_get_skip_taskbar_hint(GtkWindow *
window);
extern const gchar *gtk_window_get_title(GtkWindow * window);
extern GtkWidget *gtk_window_get_transient_for(GtkWindow * window);
extern GType gtk_window_get_type(void);
extern GdkWindowTypeHint gtk_window_get_type_hint(GtkWindow *
window);
extern gboolean gtk_window_get_urgency_hint(GtkWindow * window);
extern void gtk_window_group_add_window(GtkWindowGroup *
window_group,
        GtkWidget * window);
extern GType gtk_window_group_get_type(void);
extern GtkWidgetGroup *gtk_window_group_new(void);
extern void gtk_window_group_remove_window(GtkWindowGroup *
window_group,
        GtkWidget * window);
extern gboolean gtk_window_has_toplevel_focus(GtkWindow * window);
extern void gtk_window_iconify(GtkWindow * window);
extern gboolean gtk_window_is_active(GtkWindow * window);
extern GList *gtk_window_list_toplevels(void);
extern void gtk_window_maximize(GtkWindow * window);
extern gboolean gtk_window_mnemonic_activate(GtkWindow * window,
        guint keyval,
        GdkModifierType modifier);
extern void gtk_window_move(GtkWindow * window, gint x, gint y);

```

```

extern GtkWidget *gtk_window_new(GtkWindowType type);
extern gboolean gtk_window_parse_geometry(GtkWindow * window,
                                         const gchar * geometry);
extern GType gtk_window_position_get_type(void);
extern void gtk_window_present(GtkWindow * window);
extern void gtk_window_present_with_time(GtkWindow * window,
                                         guint32 timestamp);
extern gboolean gtk_window_propagate_key_event(GtkWindow * window,
                                              GdkEventKey * event);
extern void gtk_window_remove_accel_group(GtkWindow * window,
                                          GtkAccelGroup * accel_group);
extern void gtk_window_remove_mnemonic(GtkWindow * window, guint
keyval,
                                         GtkWidget * target);
extern void gtk_window_reshow_with_initial_size(GtkWindow *
window);
extern void gtk_window_resize(GtkWindow * window, gint width, gint
height);
extern void gtk_window_set_accept_focus(GtkWindow * window,
                                       gboolean setting);
extern void gtk_window_set_auto_startup_notification(gboolean
setting);
extern void gtk_window_set_decorated(GtkWindow * window, gboolean
setting);
extern void gtk_window_set_default(GtkWindow * window,
                                   GtkWidget * default_widget);
extern void gtk_window_set_default_icon(GdkPixbuf * icon);
extern gboolean gtk_window_set_default_icon_from_file(const gchar
*
                                         filename,
                                         GError * *err);
extern void gtk_window_set_default_icon_list(GList * list);
extern void gtk_window_set_default_icon_name(const gchar * name);
extern void gtk_window_set_default_size(GtkWindow * window, gint
width,
                                         gint height);
extern void gtk_window_set_deletable(GtkWindow * window, gboolean
setting);
extern void gtk_window_set_destroy_with_parent(GtkWindow * window,
                                              gboolean setting);
extern void gtk_window_set_focus(GtkWindow * window, GtkWidget *
focus);
extern void gtk_window_set_focus_on_map(GtkWindow * window,
                                         gboolean setting);
extern void gtk_window_set_frame_dimensions(GtkWindow * window,
gint left,
                                         gint top, gint right,
                                         gint bottom);
extern void gtk_window_set_geometry_hints(GtkWindow * window,
                                          GtkWidget * geometry_widget,
                                          GdkGeometry * geometry,
                                          GdkWindowHints geom_mask);
extern void gtk_window_set_gravity(GtkWindow * window, GdkGravity
gravity);
extern void gtk_window_set_has_frame(GtkWindow * window, gboolean
setting);
extern void gtk_window_set_icon(GtkWindow * window, GdkPixbuf *
icon);
extern gboolean gtk_window_set_icon_from_file(GtkWindow * window,
                                              const gchar * filename,
                                              GError * *err);
extern void gtk_window_set_icon_list(GtkWindow * window, GList *
list);
extern void gtk_window_set_icon_name(GtkWindow * window,
                                     const gchar * name);
extern void gtk_window_set_keep_above(GtkWindow * window,

```

```

        gboolean setting);
extern void gtk_window_set_keep_below(GtkWindow * window,
        gboolean setting);
extern void gtk_window_set_mnemonic_modifier(GtkWindow * window,
        GdkModifierType modifier);
extern void gtk_window_set_modal(GtkWindow * window, gboolean
modal);
extern void gtk_window_set_position(GtkWindow * window,
        GtkWindowPosition position);
extern void gtk_window_set_resizable(GtkWindow * window,
        gboolean resizable);
extern void gtk_window_set_role(GtkWindow * window, const gchar *
role);
extern void gtk_window_set_screen(GtkWindow * window, GdkScreen *
screen);
extern void gtk_window_set_skip_pager_hint(GtkWindow * window,
        gboolean setting);
extern void gtk_window_set_skip_taskbar_hint(GtkWindow * window,
        gboolean setting);
extern void gtk_window_set_title(GtkWindow * window, const gchar *
title);
extern void gtk_window_set_transient_for(GtkWindow * window,
        GtkWindow * parent);
extern void gtk_window_set_type_hint(GtkWindow * window,
        GdkWindowTypeHint hint);
extern void gtk_window_set_urgency_hint(GtkWindow * window,
        gboolean setting);
extern void gtk_window_set_wmclass(GtkWindow * window,
        const gchar * wmclass_name,
        const gchar * wmclass_class);
extern void gtk_window_stick(GtkWindow * window);
extern GType gtk_window_type_get_type(void);
extern void gtk_window_unfullscreen(GtkWindow * window);
extern void gtk_window_unmaximize(GtkWindow * window);
extern void gtk_window_unstick(GtkWindow * window);
extern GType gtk_wrap_mode_get_type(void);

```

17.32.2 gtk-2.0/gtk/gtkenums.h

```

typedef enum {
    GTK_STATE_NORMAL = 0,
    GTK_STATE_ACTIVE = 1,
    GTK_STATE_PRELIGHT = 2,
    GTK_STATE_SELECTED = 3,
    GTK_STATE_INSENSITIVE = 4
} GtkStateType;
typedef enum {
    GTK_SORT_ASCENDING = 0,
    GTK_SORT_DESCENDING = 1
} GtkSortType;
typedef enum {
    GTK_UPDATE_CONTINUOUS = 0,
    GTK_UPDATE_DISCONTINUOUS = 1,
    GTK_UPDATE_DELAYED = 2
} GtkUpdateType;
typedef enum {
    GTK_ORIENTATION_HORIZONTAL = 0,
    GTK_ORIENTATION_VERTICAL = 1
} GtkOrientation;
typedef enum {
    GTK_BUTTONBOX_DEFAULT_STYLE = 0,
    GTK_BUTTONBOX_SPREAD = 1,
    GTK_BUTTONBOX_EDGE = 2,
    GTK_BUTTONBOX_START = 3,
    GTK_BUTTONBOX_END = 4
}

```

```

} GtkButtonBoxStyle;
typedef enum {
    GTK_JUSTIFY_LEFT = 0,
    GTK_JUSTIFY_RIGHT = 1,
    GTK_JUSTIFY_CENTER = 2,
    GTK_JUSTIFY_FILL = 3
} GtkJustification;
typedef enum {
    GTK_TEXT_DIR_NONE = 0,
    GTK_TEXT_DIR_LTR = 1,
    GTK_TEXT_DIR_RTL = 2
} GtkTextDirection;
typedef enum {
    GTK_WRAP_NONE = 0,
    GTK_WRAP_CHAR = 1,
    GTK_WRAP_WORD = 2,
    GTK_WRAP_WORD_CHAR = 3
} GtkWrapMode;
typedef enum {
    GTK_CURVE_TYPE_LINEAR = 0,
    GTK_CURVE_TYPE_SPLINE = 1,
    GTK_CURVE_TYPE_FREE = 2
} GtkCurveType;
typedef enum {
    GTK_PACK_START = 0,
    GTK_PACK_END = 1
} GtkPackType;
typedef enum {
    GTK_ARROW_UP = 0,
    GTK_ARROW_DOWN = 1,
    GTK_ARROW_LEFT = 2,
    GTK_ARROW_RIGHT = 3
} GtkArrowType;
typedef enum {
    GTK_SHADOW_NONE = 0,
    GTK_SHADOW_IN = 1,
    GTK_SHADOW_OUT = 2,
    GTK_SHADOW_ETCHED_IN = 3,
    GTK_SHADOW_ETCHED_OUT = 4
} GtkShadowType;
typedef enum {
    GTK_DIR_TAB_FORWARD = 0,
    GTK_DIR_TAB_BACKWARD = 1,
    GTK_DIR_UP = 2,
    GTK_DIR_DOWN = 3,
    GTK_DIR_LEFT = 4,
    GTK_DIR_RIGHT = 5
} GtkDirectionType;
typedef enum {
    GTK_ICON_SIZE_INVALID = 0,
    GTK_ICON_SIZE_MENU = 1,
    GTK_ICON_SIZE_SMALL_TOOLBAR = 2,
    GTK_ICON_SIZE_LARGE_TOOLBAR = 3,
    GTK_ICON_SIZE_BUTTON = 4,
    GTK_ICON_SIZE_DND = 5,
    GTK_ICON_SIZE_DIALOG = 6
} GtkIconSize;
typedef enum {
    GTK_SELECTION_NONE = 0,
    GTK_SELECTION_SINGLE = 1,
    GTK_SELECTION_BROWSE = 2,
    GTK_SELECTION_MULTIPLE = 3,
    GTK_SELECTION_EXTENDED = 3
} GtkSelectionMode;
typedef enum {
    GTK_CORNER_TOP_LEFT = 0,

```

```

        GTK_CORNER_BOTTOM_LEFT = 1,
        GTK_CORNER_TOP_RIGHT = 2,
        GTK_CORNER_BOTTOM_RIGHT = 3
    } GtkCornerType;
typedef enum {
    GTK_POS_LEFT = 0,
    GTK_POS_RIGHT = 1,
    GTK_POS_TOP = 2,
    GTK_POS_BOTTOM = 3
} GtkPositionType;
typedef enum {
    GTK_TOOLBAR_ICONS = 0,
    GTK_TOOLBAR_TEXT = 1,
    GTK_TOOLBAR_BOTH = 2,
    GTK_TOOLBAR_BOTH_HORIZ = 3
} GtkToolbarStyle;
typedef enum {
    GTK_RELIEF_NORMAL = 0,
    GTK_RELIEF_HALF = 1,
    GTK_RELIEF_NONE = 2
} GtkReliefStyle;
typedef enum {
    GTK_PIXELS = 0,
    GTK_INCHES = 1,
    GTK_CENTIMETERS = 2
} GtkMetricType;
typedef enum {
    GTK_EXPAND = 1,
    GTK_SHRINK = 2,
    GTK_FILL = 4
} GtkAttachOptions;
typedef enum {
    GTK_MESSAGE_INFO = 0,
    GTK_MESSAGE_WARNING = 1,
    GTK_MESSAGE_QUESTION = 2,
    GTK_MESSAGE_ERROR = 3
} GtkMessageType;
typedef enum {
    GTK_POLICY_ALWAYS = 0,
    GTK_POLICY_AUTOMATIC = 1,
    GTK_POLICY_NEVER = 2
} GtkPolicyType;
typedef enum {
    GTK_RESIZE_PARENT = 0,
    GTK_RESIZE_QUEUE = 1,
    GTK_RESIZE_IMMEDIATE = 2
} GtkResizeMode;
typedef enum {
    GTK_WINDOW_TOPLEVEL = 0,
    GTK_WINDOW_POPUP = 1
} GtkWindowType;
typedef enum {
    GTK_WIN_POS_NONE = 0,
    GTK_WIN_POS_CENTER = 1,
    GTK_WIN_POS_MOUSE = 2,
    GTK_WIN_POS_CENTER_ALWAYS = 3,
    GTK_WIN_POS_CENTER_ON_PARENT = 4
} GtkWindowPosition;
typedef enum {
    GTK_EXPANDER_COLLAPSED = 0,
    GTK_EXPANDER_SEMI_COLLAPSED = 1,
    GTK_EXPANDER_SEMI_EXPANDED = 2,
    GTK_EXPANDER_EXPANDED = 3
} GtkExpanderStyle;
typedef enum {
    GTK_PATH_PRIO_LOWEST = 0,

```

```

    GTK_PATH_PRIO_GTK = 4,
    GTK_PATH_PRIO_APPLICATION = 8,
    GTK_PATH_PRIO_THEME = 10,
    GTK_PATH_PRIO_RC = 12,
    GTK_PATH_PRIO_HIGHEST = 15
} GtkPathPriorityType;
typedef enum {
    GTK_PATH_WIDGET = 0,
    GTK_PATH_WIDGET_CLASS = 1,
    GTK_PATH_CLASS = 2
} GtkPathType;
typedef enum {
    GTK_SCROLL_NONE = 0,
    GTK_SCROLL_JUMP = 1,
    GTK_SCROLL_STEP_BACKWARD = 2,
    GTK_SCROLL_STEP_FORWARD = 3,
    GTK_SCROLL_PAGE_BACKWARD = 4,
    GTK_SCROLL_PAGE_FORWARD = 5,
    GTK_SCROLL_STEP_UP = 6,
    GTK_SCROLL_STEP_DOWN = 7,
    GTK_SCROLL_PAGE_UP = 8,
    GTK_SCROLL_PAGE_DOWN = 9,
    GTK_SCROLL_STEP_LEFT = 10,
    GTK_SCROLL_STEP_RIGHT = 11,
    GTK_SCROLL_PAGE_LEFT = 12,
    GTK_SCROLL_PAGE_RIGHT = 13,
    GTK_SCROLL_START = 14,
    GTK_SCROLL_END = 15
} GtkScrollType;
typedef enum {
    GTK_MOVEMENT_LOGICAL_POSITIONS = 0,
    GTK_MOVEMENT_VISUAL_POSITIONS = 1,
    GTK_MOVEMENT_WORDS = 2,
    GTK_MOVEMENT_DISPLAY_LINES = 3,
    GTK_MOVEMENT_DISPLAY_LINE_ENDS = 4,
    GTK_MOVEMENT_PARAGRAPHS = 5,
    GTK_MOVEMENT_PARAGRAPH_ENDS = 6,
    GTK_MOVEMENT_PAGES = 7,
    GTK_MOVEMENT_BUFFER_ENDS = 8,
    GTK_MOVEMENT_HORIZONTAL_PAGES = 9
} GtkMovementStep;
typedef enum {
    GTK_MENU_DIR_PARENT = 0,
    GTK_MENU_DIR_CHILD = 1,
    GTK_MENU_DIR_NEXT = 2,
    GTK_MENU_DIR_PREV = 3
} GtkMenuDirectionType;
typedef enum {
    GTK_SCROLL_STEPS = 0,
    GTK_SCROLL_PAGES = 1,
    GTK_SCROLL_ENDS = 2,
    GTK_SCROLL_HORIZONTAL_STEPS = 3,
    GTK_SCROLL_HORIZONTAL_PAGES = 4,
    GTK_SCROLL_HORIZONTAL_ENDS = 5
} GtkScrollStep;
typedef enum {
    GTK_DELETE_CHARS = 0,
    GTK_DELETE_WORD_ENDS = 1,
    GTK_DELETE_WORDS = 2,
    GTK_DELETE_DISPLAY_LINES = 3,
    GTK_DELETE_DISPLAY_LINE_ENDS = 4,
    GTK_DELETE_PARAGRAPH_ENDS = 5,
    GTK_DELETE_PARAGRAPHS = 6,
    GTK_DELETE_WHITESPACE = 7
} GtkDeleteType;
typedef enum {

```

```

        GTK_VISIBILITY_NONE = 0,
        GTK_VISIBILITY_PARTIAL = 1,
        GTK_VISIBILITY_FULL = 2
    } GtkVisibility;
typedef enum {
    GTK_IM_PREEDIT_NOTHING,
    GTK_IM_PREEDIT_CALLBACK,
    GTK_IM_PREEDIT_NONE
} GtkIMPreeditStyle;
typedef enum {
    GTK_ANCHOR_CENTER,
    GTK_ANCHOR_NORTH,
    GTK_ANCHOR_NORTH_WEST,
    GTK_ANCHOR_NORTH_EAST,
    GTK_ANCHOR_SOUTH,
    GTK_ANCHOR_SOUTH_WEST,
    GTK_ANCHOR_SOUTH_EAST,
    GTK_ANCHOR_WEST,
    GTK_ANCHOR_EAST,
    GTK_ANCHOR_N = GTK_ANCHOR_NORTH,
    GTK_ANCHOR_NW = GTK_ANCHOR_NORTH_WEST,
    GTK_ANCHOR_NE = GTK_ANCHOR_NORTH_EAST,
    GTK_ANCHOR_S = GTK_ANCHOR_SOUTH,
    GTK_ANCHOR_SW = GTK_ANCHOR_SOUTH_WEST,
    GTK_ANCHOR_SE = GTK_ANCHOR_SOUTH_EAST,
    GTK_ANCHOR_W = GTK_ANCHOR_WEST,
    GTK_ANCHOR_E = GTK_ANCHOR_EAST
} GtkAnchorType;
typedef enum {
    GTK_IM_STATUS_NOTHING,
    GTK_IM_STATUS_CALLBACK,
    GTK_IM_STATUS_NONE
} GtkIMStatusStyle;
typedef enum {
    GTK_PACK_DIRECTION_LTR,
    GTK_PACK_DIRECTION_RTL,
    GTK_PACK_DIRECTION_TTB,
    GTK_PACK_DIRECTION_BTT
} GtkPackDirection;
typedef enum {
    GTK_PAGE_ORIENTATION_PORTRAIT,
    GTK_PAGE_ORIENTATION_LANDSCAPE,
    GTK_PAGE_ORIENTATION_REVERSE_PORTRAIT,
    GTK_PAGE_ORIENTATION_REVERSE_LANDSCAPE
} GtkPageOrientation;
typedef enum {
    GTK_PRINT_QUALITY_LOW,
    GTK_PRINT_QUALITY_NORMAL,
    GTK_PRINT_QUALITY_HIGH,
    GTK_PRINT_QUALITY_DRAFT
} GtkPrintQuality;
typedef enum {
    GTK_PRINT_DUPLEX_SIMPLEX,
    GTK_PRINT_DUPLEX_HORIZONTAL,
    GTK_PRINT_DUPLEX_VERTICAL
} GtkPrintDuplex;
typedef enum {
    GTK_PRINT_PAGES_ALL,
    GTK_PRINT_PAGES_CURRENT,
    GTK_PRINT_PAGES_RANGES,
    GTK_PRINT_PAGES_SELECTION
} GtkPrintPages;
typedef enum {
    GTK_PAGE_SET_ALL,
    GTK_PAGE_SET_EVEN,
    GTK_PAGE_SET_ODD

```



```

} GtkPageSet;
typedef enum {
    GTK_NUMBER_UP_LAYOUT_LEFT_TO_RIGHT_TOP_TO_BOTTOM,
    GTK_NUMBER_UP_LAYOUT_LEFT_TO_RIGHT_BOTTOM_TO_TOP,
    GTK_NUMBER_UP_LAYOUT_RIGHT_TO_LEFT_TOP_TO_BOTTOM,
    GTK_NUMBER_UP_LAYOUT_RIGHT_TO_LEFT_BOTTOM_TO_TOP,
    GTK_NUMBER_UP_LAYOUT_TOP_TO_BOTTOM_LEFT_TO_RIGHT,
    GTK_NUMBER_UP_LAYOUT_TOP_TO_BOTTOM_RIGHT_TO_LEFT,
    GTK_NUMBER_UP_LAYOUT_BOTTOM_TO_TOP_LEFT_TO_RIGHT,
    GTK_NUMBER_UP_LAYOUT_BOTTOM_TO_TOP_RIGHT_TO_LEFT
} GtkNumberUpLayout;
typedef enum {
    GTK_UNIT_PIXEL,
    GTK_UNIT_POINTS,
    GTK_UNIT_INCH,
    GTK_UNIT_MM
} GtkUnit;
typedef enum {
    GTK_SENSITIVITY_AUTO,
    GTK_SENSITIVITY_ON,
    GTK_SENSITIVITY_OFF
} GtkSensitivityType;
typedef enum {
    GTK_TREE_VIEW_GRID_LINES_NONE,
    GTK_TREE_VIEW_GRID_LINES_HORIZONTAL,
    GTK_TREE_VIEW_GRID_LINES_VERTICAL,
    GTK_TREE_VIEW_GRID_LINES_BOTH
} GtkTreeViewGridLines;

```

17.32.3 gtk-2.0/gtk/gtkprintoperationpreview.h

```

#define __GTK_PRINT_OPERATION_PREVIEW_H__
#define GTK_TYPE_PRINT_OPERATION_PREVIEW
(gtk_print_operation_preview_get_type ())
#define GTK_PRINT_OPERATION_PREVIEW(obj)
(G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_PRINT_OPERATION_PREVIEW, GtkPrintOperationPreview))
#define GTK_IS_PRINT_OPERATION_PREVIEW(obj)
(G_TYPE_CHECK_INSTANCE_TYPE ((obj),
GTK_TYPE_PRINT_OPERATION_PREVIEW))
#define GTK_PRINT_OPERATION_PREVIEW_GET_IFACE(obj)
(G_TYPE_INSTANCE_GET_INTERFACE ((obj),
GTK_TYPE_PRINT_OPERATION_PREVIEW, GtkPrintOperationPreviewIface))

typedef struct _GtkPrintOperationPreview GtkPrintOperationPreview;
typedef struct _GtkPrintOperationPreviewIface {
    GTypeInterface g_iface;
    void (*ready) (void);
    void (*got_page_size) (void);
    void (*render_page) (void);
    gboolean (*is_selected) (void);
    void (*end_preview) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
    void (*_gtk_reserved7) (void);
} GtkPrintOperationPreviewIface;
extern void
gtk_print_operation_preview_end_preview(GtkPrintOperationPreview *
preview);
extern GType gtk_print_operation_preview_get_type(void);
extern gboolean

```

```

gtk_print_operation_preview_is_selected(GtkPrintOperationPreview *
preview,
                                     gint page_nr);

extern void
gtk_print_operation_preview_render_page(GtkPrintOperationPreview *
preview,
                                     gint page_nr);

```

17.32.4 gtk-2.0/gtk/gtkrecentfilter.h

```

#define __GTK_RECENT_FILTER_H__
#define GTK_TYPE_RECENT_FILTER (gtk_recent_filter_get_type ())
#define GTK_RECENT_FILTER(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_RECENT_FILTER, GtkRecentFilter))
#define GTK_IS_RECENT_FILTER(obj) (G_TYPE_CHECK_INSTANCE_TYPE
((obj), GTK_TYPE_RECENT_FILTER))

typedef struct _GtkRecentFilter GtkRecentFilter;
typedef struct _GtkRecentFilterInfo {
    GtkRecentFilterFlags contains;
    const char *uri;
    const char *display_name;
    const char *mime_type;
    const char **applications;
    const char **groups;
    gint age;
} GtkRecentFilterInfo;
typedef enum {
    GTK_RECENT_FILTER_URI,
    GTK_RECENT_FILTER_DISPLAY_NAME,
    GTK_RECENT_FILTER_MIME_TYPE,
    GTK_RECENT_FILTER_APPLICATION,
    GTK_RECENT_FILTER_GROUP,
    GTK_RECENT_FILTER_AGE
} GtkRecentFilterFlags;
typedef gboolean(*GtkRecentFilterFunc) (const GtkRecentFilterInfo
*
                                     filter_info, gpointer user_data);

extern void gtk_recent_filter_add_age(GtkRecentFilter * filter,
gint days);
extern void gtk_recent_filter_add_application(GtkRecentFilter *
filter,
                                     const char *application);
extern void gtk_recent_filter_add_custom(GtkRecentFilter * filter,
                                     GtkRecentFilterFlags needed,
                                     GtkRecentFilterFunc func,
                                     gpointer data,
                                     GDestroyNotify data_destroy);
extern void gtk_recent_filter_add_group(GtkRecentFilter * filter,
                                     const char *group);
extern void gtk_recent_filter_add_mime_type(GtkRecentFilter *
filter,
                                     const char *mime_type);
extern void gtk_recent_filter_add_pattern(GtkRecentFilter * filter,
                                     const char *pattern);
extern void gtk_recent_filter_add_pixbuf_formats(GtkRecentFilter *
filter);
extern gboolean gtk_recent_filter_filter(GtkRecentFilter * filter,
                                     const GtkRecentFilterInfo *
filter_info);
extern const char *gtk_recent_filter_get_name(GtkRecentFilter *
filter);
extern
                                     GtkRecentFilterFlags
gtk_recent_filter_get_needed(GtkRecentFilter *
                                     filter);

```

```

extern GType gtk_recent_filter_get_type(void);
extern GtkRecentFilter *gtk_recent_filter_new(void);
extern void gtk_recent_filter_set_name(GtkRecentFilter * filter,
                                     const char *name);

```

17.32.5 gtk-2.0/gtk/gtkrecentmanager.h

```

#define _GTK_RECENT_MANAGER_H_
#define GTK_TYPE_RECENT_INFO (gtk_recent_info_get_type ())
#define GTK_RECENT_MANAGER_ERROR GTK_RECENT_MANAGER_ERROR
(gtk_recent_manager_error_quark ())
#define GTK_TYPE_RECENT_MANAGER (gtk_recent_manager_get_type ())
#define GTK_RECENT_MANAGER_CLASS(klass) (G_TYPE_CHECK_CLASS_CAST
((klass), GTK_TYPE_RECENT_MANAGER, GtkRecentManagerClass))
#define GTK_IS_RECENT_MANAGER_CLASS(klass)
(G_TYPE_CHECK_CLASS_TYPE ((klass), GTK_TYPE_RECENT_MANAGER))
#define GTK_RECENT_MANAGER(obj) (G_TYPE_CHECK_INSTANCE_CAST ((obj),
GTK_TYPE_RECENT_MANAGER, GtkRecentManager))
#define GTK_IS_RECENT_MANAGER(obj) (G_TYPE_CHECK_INSTANCE_TYPE
((obj), GTK_TYPE_RECENT_MANAGER))
#define GTK_RECENT_MANAGER_GET_CLASS(obj)
(G_TYPE_INSTANCE_GET_CLASS ((obj), GTK_TYPE_RECENT_MANAGER,
GtkRecentManagerClass))

typedef struct _GtkRecentInfo GtkRecentInfo;
typedef struct _GtkRecentData {
    gchar *display_name;
    gchar *description;
    gchar *mime_type;
    gchar *app_name;
    gchar *app_exec;
    gchar **groups;
    gboolean is_private;
} GtkRecentData;
typedef struct _GtkRecentManager {
    GObject parent_instance;
    GtkRecentManagerPrivate *priv;
} GtkRecentManager;
typedef struct _GtkRecentManagerClass {
    GObjectClass parent_class;
    void (*changed) (void);
    void (*_gtk_recent1) (void);
    void (*_gtk_recent2) (void);
    void (*_gtk_recent3) (void);
    void (*_gtk_recent4) (void);
} GtkRecentManagerClass;
typedef struct _GtkRecentManagerPrivate GtkRecentManagerPrivate;
typedef enum {
    GTK_RECENT_MANAGER_ERROR_NOT_FOUND,
    GTK_RECENT_MANAGER_ERROR_INVALID_URI,
    GTK_RECENT_MANAGER_ERROR_INVALID_ENCODING,
    GTK_RECENT_MANAGER_ERROR_NOT_REGISTERED,
    GTK_RECENT_MANAGER_ERROR_READ,
    GTK_RECENT_MANAGER_ERROR_WRITE,
    GTK_RECENT_MANAGER_ERROR_UNKNOWN
} GtkRecentManagerError;
extern gboolean gtk_recent_info_exists(GtkRecentInfo * info);
extern time_t gtk_recent_info_get_added(GtkRecentInfo * info);
extern gint gtk_recent_info_get_age(GtkRecentInfo * info);
extern gboolean gtk_recent_info_get_application_info(GtkRecentInfo
* info,
                                     const char *app_name,
                                     const char **app_exec,
                                     guint * count,
                                     time_t * time_);

```

```

extern gchar **gtk_recent_info_get_applications(GtkRecentInfo *
info,
                                           gsize * length);
extern const char *gtk_recent_info_get_description(GtkRecentInfo *
info);
extern const char *gtk_recent_info_get_display_name(GtkRecentInfo
* info);
extern gchar **gtk_recent_info_get_groups(GtkRecentInfo * info,
                                           gsize * length);
extern GdkPixbuf *gtk_recent_info_get_icon(GtkRecentInfo * info,
                                           gint size);
extern const char *gtk_recent_info_get_mime_type(GtkRecentInfo *
info);
extern time_t gtk_recent_info_get_modified(GtkRecentInfo * info);
extern gboolean gtk_recent_info_get_private_hint(GtkRecentInfo *
info);
extern gchar *gtk_recent_info_get_short_name(GtkRecentInfo * info);
extern GType gtk_recent_info_get_type(void);
extern const char *gtk_recent_info_get_uri(GtkRecentInfo * info);
extern gchar *gtk_recent_info_get_uri_display(GtkRecentInfo *
info);
extern time_t gtk_recent_info_get_visited(GtkRecentInfo * info);
extern gboolean gtk_recent_info_has_application(GtkRecentInfo *
info,
                                           const char *app_name);
extern gboolean gtk_recent_info_has_group(GtkRecentInfo * info,
                                           const char *group_name);
extern gboolean gtk_recent_info_is_local(GtkRecentInfo * info);
extern gchar *gtk_recent_info_last_application(GtkRecentInfo *
info);
extern gboolean gtk_recent_info_match(GtkRecentInfo * info_a,
                                      GtkRecentInfo * info_b);
extern GtkRecentInfo *gtk_recent_info_ref(GtkRecentInfo * info);
extern void gtk_recent_info_unref(GtkRecentInfo * info);
extern gboolean gtk_recent_manager_add_full(GtkRecentManager *
manager,
                                           const char *uri,
                                           const GtkRecentData *
recent_data);
extern gboolean gtk_recent_manager_add_item(GtkRecentManager *
manager,
                                           const char *uri);
extern GQuark gtk_recent_manager_error_quark(void);
extern GtkRecentManager *gtk_recent_manager_get_default(void);
extern GList *gtk_recent_manager_get_items(GtkRecentManager *
manager);
extern gint gtk_recent_manager_get_limit(GtkRecentManager *
manager);
extern GType gtk_recent_manager_get_type(void);
extern gboolean gtk_recent_manager_has_item(GtkRecentManager *
manager,
                                           const char *uri);
extern
                                           GtkRecentInfo
*gtk_recent_manager_lookup_item(GtkRecentManager *
manager,
                                const char *uri,
                                GError * *error);
extern gboolean gtk_recent_manager_move_item(GtkRecentManager *
manager,
                                           const char *uri,
                                           const char *new_uri,
                                           GError * *error);
extern GtkRecentManager *gtk_recent_manager_new(void);
extern gint gtk_recent_manager_purge_items(GtkRecentManager *
manager,
                                           GError * *error);

```

```

extern gboolean gtk_recent_manager_remove_item(GtkRecentManager *
manager,
                                             const char *uri,
                                             GError * *error);
extern void      gtk_recent_manager_set_limit(GtkRecentManager *
manager,
                                             gint limit);

```

17.32.6 gtk-2.0/gtk/gtkstatusicon.h

```

#define __GTK_STATUS_ICON_H__
#define GTK_TYPE_STATUS_ICON (gtk_status_icon_get_type ())
#define GTK_STATUS_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_CAST ((k), GTK_TYPE_STATUS_ICON, GtkStatusIconClass))
#define GTK_IS_STATUS_ICON_CLASS(k) (G_TYPE_CHECK_CLASS_TYPE ((k), GTK_TYPE_STATUS_ICON))
#define GTK_STATUS_ICON(o) (G_TYPE_CHECK_INSTANCE_CAST ((o), GTK_TYPE_STATUS_ICON, GtkStatusIcon))
#define GTK_IS_STATUS_ICON(o) (G_TYPE_CHECK_INSTANCE_TYPE ((o), GTK_TYPE_STATUS_ICON))
#define GTK_STATUS_ICON_GET_CLASS(o) (G_TYPE_INSTANCE_GET_CLASS ((o), GTK_TYPE_STATUS_ICON, GtkStatusIconClass))

typedef struct _GtkStatusIcon {
    GObject parent_instance;
    GtkStatusIconPrivate *priv;
} GtkStatusIcon;

typedef struct _GtkStatusIconClass {
    GObjectClass parent_class;
    void (*activate) (void);
    void (*popup_menu) (void);
    gboolean(*size_changed) (void);
    gboolean(*button_press_event) (GtkStatusIcon *, GdkEventButton
*);
    gboolean(*button_release_event) (GtkStatusIcon *,
GdkEventButton *);
    gboolean(*scroll_event) (GtkStatusIcon *, GdkEventButton *);
    gboolean(*query_tooltip) (GtkStatusIcon *, gint, gint,
gboolean,
                             GtkTooltip *);
    void *__gtk_reserved5;
    void *__gtk_reserved6;
} GtkStatusIconClass;

typedef struct _GtkStatusIconPrivate GtkStatusIconPrivate;
typedef struct _GtkTooltip GtkTooltip;
extern gboolean gtk_status_icon_get_blinking(GtkStatusIcon *
status_icon);
extern gboolean gtk_status_icon_get_geometry(GtkStatusIcon *
status_icon,
                                             GdkScreen * *screen,
                                             GdkRectangle * area,
                                             GtkOrientation *
orientation);
extern const char *gtk_status_icon_get_icon_name(GtkStatusIcon *
status_icon);
extern GdkPixbuf *gtk_status_icon_get_pixbuf(GtkStatusIcon *
status_icon);
extern gint gtk_status_icon_get_size(GtkStatusIcon * status_icon);
extern const char *gtk_status_icon_get_stock(GtkStatusIcon *
status_icon);
extern GtkImageType gtk_status_icon_get_storage_type(GtkStatusIcon
*
                                             status_icon);
extern GType gtk_status_icon_get_type(void);

```

```

extern gboolean gtk_status_icon_get_visible(GtkStatusIcon *
status_icon);
extern gboolean gtk_status_icon_is_embedded(GtkStatusIcon *
status_icon);
extern GtkStatusIcon *gtk_status_icon_new(void);
extern GtkStatusIcon *gtk_status_icon_new_from_file(const char
*filename);
extern GtkStatusIcon *gtk_status_icon_new_from_icon_name(const
char
                                     *icon_name);
extern GtkStatusIcon *gtk_status_icon_new_from_pixbuf(GdkPixbuf *
pixbuf);
extern GtkStatusIcon *gtk_status_icon_new_from_stock(const char
*stock_id);
extern void gtk_status_icon_position_menu(GtkMenu * menu, gint * x,
                                     gint * y, gboolean * push_in,
                                     gpointer user_data);
extern void gtk_status_icon_set_blinking(GtkStatusIcon *
status_icon,
                                     gboolean blinking);
extern void gtk_status_icon_set_from_file(GtkStatusIcon *
status_icon,
                                     const char *filename);
extern void gtk_status_icon_set_from_icon_name(GtkStatusIcon *
status_icon,
                                     const char *icon_name);
extern void gtk_status_icon_set_from_pixbuf(GtkStatusIcon *
status_icon,
                                     GdkPixbuf * pixbuf);
extern void gtk_status_icon_set_from_stock(GtkStatusIcon *
status_icon,
                                     const char *stock_id);
extern void gtk_status_icon_set_tooltip(GtkStatusIcon *
status_icon,
                                     const gchar * tooltip_text);
extern void gtk_status_icon_set_visible(GtkStatusIcon *
status_icon,
                                     gboolean visible);

```

17.32.7 gtk-2.0/gtk/gtktextbufferrichtext.h

```

#define __GTK_TEXT_BUFFER_RICH_TEXT_H__

typedef guint8 *(*GtkTextBufferSerializeFunc) (GtkTextBuffer *,
                                               GtkTextBuffer *,
                                               GtkTextIter *,
                                               GtkTextIter *, gsize *,
                                               gpointer);
typedef gboolean(*GtkTextBufferDeserializeFunc) (GtkTextBuffer *,
                                               GtkTextBuffer *,
                                               GtkTextIter *,
                                               const unsigned char *,
                                               gsize, gboolean, gpointer,
                                               GError * *);
extern gboolean gtk_text_buffer_deserialize(GtkTextBuffer *
register_buffer,
                                           GtkTextBuffer *
content_buffer,
                                           GdkAtom format,
                                           GtkTextIter * iter,
                                           const unsigned char *data,
                                           gsize length, GError * *error);
extern gboolean
gtk_text_buffer_deserialize_get_can_create_tags(GtkTextBuffer *
buffer,

```

```

                                GdkAtom format);
extern void
gtk_text_buffer_deserialize_set_can_create_tags(GtkTextBuffer *
                                                buffer,
                                                GdkAtom format,
                                                gboolean

can_create_tags);
extern GdkAtom
*gtk_text_buffer_get_deserialize_formats(GtkTextBuffer *
                                         buffer,
                                         gint * n_formats);
extern GdkAtom
*gtk_text_buffer_get_serialize_formats(GtkTextBuffer *
                                       buffer,
                                       gint * n_formats);
extern GdkAtom
gtk_text_buffer_register_deserialize_format(GtkTextBuffer *
                                             buffer,
                                             const char
                                             *mime_type,

GtkTextBufferDeserializeFunc
                                             function,
                                             gpointer
                                             user_data,
                                             GDestroyNotify

user_data_destroy);
extern GdkAtom
gtk_text_buffer_register_deserialize_tagset(GtkTextBuffer *
                                             buffer,
                                             const char
                                             *tagset_name);
extern GdkAtom
gtk_text_buffer_register_serialize_format(GtkTextBuffer *
                                          buffer,
                                          const char
                                          *mime_type,

GtkTextBufferSerializeFunc
                                          function,
                                          gpointer
                                          user_data,
                                          GDestroyNotify
                                          user_data_destroy);
extern GdkAtom
gtk_text_buffer_register_serialize_tagset(GtkTextBuffer *
                                          buffer,
                                          const char
                                          *tagset_name);
extern guint8 *gtk_text_buffer_serialize(GtkTextBuffer *
register_buffer,
                                          GtkTextBuffer * content_buffer,
                                          GdkAtom format,
                                          GtkTextIter * start,
                                          GtkTextIter * end,
                                          gsize * length);
extern void
gtk_text_buffer_unregister_deserialize_format(GtkTextBuffer *
                                              buffer,
                                              GdkAtom format);
extern void
gtk_text_buffer_unregister_serialize_format(GtkTextBuffer *
                                             buffer,
                                             GdkAtom format);

```

17.32.8 gtk-unix-print-2.0/gtk/gtkpagesetupunixdialog.h

```

typedef struct _GtkPageSetupUnixDialog {
    GtkDialog parent_instance;
    GtkPageSetupUnixDialogPrivate *priv;
} GtkPageSetupUnixDialog;

typedef struct _GtkPageSetupUnixDialogClass {
    GtkDialogClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
    void (*_gtk_reserved7) (void);
} GtkPageSetupUnixDialogClass;

typedef struct _GtkPageSetupUnixDialogPrivate
GtkPageSetupUnixDialogPrivate;

extern GtkPageSetup

*gtk_page_setup_unix_dialog_get_page_setup(GtkPageSetupUnixDialog
*
                                         dialog);

extern GtkPrintSettings

*gtk_page_setup_unix_dialog_get_print_settings(GtkPageSetupUnixDi
alog *
                                         dialog);

extern GType gtk_page_setup_unix_dialog_get_type(void);
extern GtkWidget *gtk_page_setup_unix_dialog_new(const char *title,
                                         GtkWindow * parent);

extern void
gtk_page_setup_unix_dialog_set_page_setup(GtkPageSetupUnixDialog *
dialog,
                                         GtkPageSetup * page_setup);

extern void
gtk_page_setup_unix_dialog_set_print_settings(GtkPageSetupUnixDia
log *
                                         dialog,
                                         GtkPrintSettings *
print_settings);

```

17.32.9 gtk-unix-print-2.0/gtk/gtkprinter.h

```

typedef enum {
    GTK_PRINT_CAPABILITY_PAGE_SET = 1 << 0,
    GTK_PRINT_CAPABILITY_COPIES = 1 << 1,
    GTK_PRINT_CAPABILITY_COLLATE = 1 << 2,
    GTK_PRINT_CAPABILITY_REVERSE = 1 << 3,
    GTK_PRINT_CAPABILITY_SCALE = 1 << 4,
    GTK_PRINT_CAPABILITY_GENERATE_PDF = 1 << 5,
    GTK_PRINT_CAPABILITY_GENERATE_PS = 1 << 6,
    GTK_PRINT_CAPABILITY_PREVIEW = 1 << 7,
    GTK_PRINT_CAPABILITY_NUMBER_UP = 1 << 8,
    GTK_PRINT_CAPABILITY_NUMBER_UP_LAYOUT = 1 << 9
} GtkPrintCapabilities;

typedef struct _GtkPrinter {
    GObject parent_instance;
    GtkPrinterPrivate *priv;
} GtkPrinter;

typedef struct _GtkPrinterClass {
    GObjectClass parent_class;

```



```

    void (*details_acquired) (GtkPrinter * printer, gboolean
success);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
    void (*_gtk_reserved7) (void);
} GtkPrinterClass;
typedef struct _GtkPrinterPrivate GtkPrinterPrivate;
typedef struct _GtkPrintBackend GtkPrintBackend;
typedef gboolean(*GtkPrinterFunc) (GtkPrinter *, gpointer);
extern void gtk_enumerate_printers(GtkPrinterFunc func, gpointer
data,
                                GDestroyNotify destroy, gboolean
wait);
extern GType gtk_print_capabilities_get_type(void);
extern gboolean gtk_printer_accepts_pdf(GtkPrinter * printer);
extern gboolean gtk_printer_accepts_ps(GtkPrinter * printer);
extern gint gtk_printer_compare(GtkPrinter * a, GtkPrinter * b);
extern GtkPrintBackend *gtk_printer_get_backend(GtkPrinter *
printer);
extern const char *gtk_printer_get_description(GtkPrinter *
printer);
extern const char *gtk_printer_get_icon_name(GtkPrinter * printer);
extern gint gtk_printer_get_job_count(GtkPrinter * printer);
extern const char *gtk_printer_get_location(GtkPrinter * printer);
extern const char *gtk_printer_get_name(GtkPrinter * printer);
extern const char *gtk_printer_get_state_message(GtkPrinter *
printer);
extern GType gtk_printer_get_type(void);
extern gboolean gtk_printer_is_active(GtkPrinter * printer);
extern gboolean gtk_printer_is_default(GtkPrinter * printer);
extern gboolean gtk_printer_is_virtual(GtkPrinter * printer);
extern GtkPrinter *gtk_printer_new(const char *name,
                                GtkPrintBackend * backend,
                                gboolean virtual_);

```

17.32.10 gtk-unix-print-2.0/gtk/gtkprintjob.h

```

typedef struct _GtkPrintJob {
    GObject parent_instance;
    GtkPrintJobPrivate *priv;
    GtkPrintPages print_pages;
    GtkPageRange *page_ranges;
    gint num_page_ranges;
    GtkPageSet page_set;
    gint num_copies;
    gdouble scale;
    unsigned int rotate_to_orientation:1;
    unsigned int collate:1;
    unsigned int reverse:1;
    guint number_up;
    GtkNumberUpLayout number_up_layout;
} GtkPrintJob;
typedef struct _GtkPrintJobClass {
    GObjectClass parent_class;
    void (*status_changed) (void);
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
}

```

```

    void (*_gtk_reserved7) (void);
} GtkPrintJobClass;
typedef struct _GtkPrintJobPrivate GtkPrintJobPrivate;
typedef void (*GtkPrintJobCompleteFunc) (GtkPrintJob *, gpointer,
                                           GError *);

extern GtkPrinter *gtk_print_job_get_printer(GtkPrintJob * job);
extern GtkPrintSettings *gtk_print_job_get_settings(GtkPrintJob *
job);
extern GtkPrintStatus gtk_print_job_get_status(GtkPrintJob * job);
extern cairo_surface_t *gtk_print_job_get_surface(GtkPrintJob *
job,
                                           GError * *error);
extern const char *gtk_print_job_get_title(GtkPrintJob * job);
extern gboolean gtk_print_job_get_track_print_status(GtkPrintJob *
job);
extern GType gtk_print_job_get_type(void);
extern GtkPrintJob *gtk_print_job_new(const char *title,
                                       GtkPrinter * printer,
                                       GtkPrintSettings * settings,
                                       GtkPageSetup * page_setup);
extern void gtk_print_job_send(GtkPrintJob * job,
                               GtkPrintJobCompleteFunc callback,
                               gpointer user_data,
                               GDestroyNotify
dnotify);
extern gboolean gtk_print_job_set_source_file(GtkPrintJob * job,
                                              const char *filename,
                                              GError * *error);
extern void gtk_print_job_set_track_print_status(GtkPrintJob * job,
                                                  gboolean track_status);

```

17.32.11 gtk-unix-print-2.0/gtk/gtkprintunixdialog.h

```

typedef struct _GtkPrintUnixDialog {
    GtkDialog parent_instance;
    GtkPrintUnixDialogPrivate *priv;
} GtkPrintUnixDialog;
typedef struct _GtkPrintUnixDialogClass {
    GtkDialogClass parent_class;
    void (*_gtk_reserved1) (void);
    void (*_gtk_reserved2) (void);
    void (*_gtk_reserved3) (void);
    void (*_gtk_reserved4) (void);
    void (*_gtk_reserved5) (void);
    void (*_gtk_reserved6) (void);
    void (*_gtk_reserved7) (void);
} GtkPrintUnixDialogClass;
typedef struct GtkPrintUnixDialogPrivate GtkPrintUnixDialogPrivate;
extern void
gtk_print_unix_dialog_add_custom_tab(GtkPrintUnixDialog *
                                     dialog, GtkWidget * child,
                                     GtkWidget * tab_label);
extern gint
gtk_print_unix_dialog_get_current_page(GtkPrintUnixDialog *
                                       dialog);
extern GtkPageSetup
*gtk_print_unix_dialog_get_page_setup(GtkPrintUnixDialog *
dialog);
extern GtkPrinter
*gtk_print_unix_dialog_get_selected_printer(GtkPrintUnixDialog
*
                                       dialog);
extern GtkPrintSettings
*gtk_print_unix_dialog_get_settings(GtkPrintUnixDialog *
dialog);
extern GType gtk_print_unix_dialog_get_type(void);

```

```

extern GtkWidget *gtk_print_unix_dialog_new(const char *title,
                                           GtkWidget * parent);

extern
void
gtk_print_unix_dialog_set_current_page(GtkPrintUnixDialog *
                                       dialog,
                                       gint current_page);

extern void
gtk_print_unix_dialog_set_manual_capabilities(GtkPrintUnixDialog *
                                              dialog,
                                              GtkPrintCapabilities
                                              capabilities);

extern
void
gtk_print_unix_dialog_set_page_setup(GtkPrintUnixDialog *
                                     dialog,
                                     GtkPageSetup *
                                     page_setup);

extern void gtk_print_unix_dialog_set_settings(GtkPrintUnixDialog
* dialog,
                                              GtkPrintSettings *
                                              settings);

```

17.32.12 gtk-unix-print-2.0/gtk/gtkunixprint.h

```

#define __GTK_UNIX_PRINT_H_INSIDE__
#define __GTK_UNIX_PRINT_H__

```

17.33 Interface Definitions for libgtk-x11-2.0

The interfaces defined on the following pages are included in libgtk-x11-2.0 and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 17.31 shall behave as described in the referenced base document.

gtk_button_get_image

Name

gtk_button_get_image — get button image widget

Synopsis

```

#include <gtk/gtk.h>
GtkWidget *gtk_button_get_image(GtkButton *button);

```

Description

The `gtk_button_get_image()` gets the widget that is currently set as the image of *button*. This may have been explicitly set by `gtk_button_set_image()` or constructed by `gtk_button_new_from_stock()`.

Return Value

The `gtk_button_get_image()` function returns a `GtkWidget` pointer referring to the image. If there is no associated image, `NULL` is returned.

Errors

No errors are defined.

gtk_button_set_image**Name**

gtk_button_set_image — set button image widget

Synopsis

```
#include <gtk/gtk.h>
void gtk_button_set_image(GtkButton *button, GtkWidget *widget);
```

Description

The `gtk_button_set_image()` function sets the image of *button* to the widget given by *widget*. Whether the image is displayed or not depends on the setting of the `gtk-button-images` property. It is not necessary to call `gtk_widget_show()` on the image.

Errors

No errors are defined.

gtk_toolbar_get_tooltips**Name**

gtk_toolbar_get_tooltips — retrieve whether tooltips are enabled

Synopsis

```
#include <gtk/gtk.h>
gboolean gtk_toolbar_get_tooltips(GtkToolbar * toolbar);
```

Description

`gtk_toolbar_get_tooltips()` shall retrieve whether the tooltips of *toolbar* are enabled.

This interface has been deprecated since version 2.14 of the GTK+ specification.

Return Value

`gtk_toolbar_get_tooltips()` shall return `TRUE` if the tooltips of *toolbar* are enabled, and `FALSE` if they are not.

See Also

`gtk_toolbar_set_tooltips()`

gtk_toolbar_set_tooltips

Name

gtk_toolbar_set_tooltips — set whether a toolbar's tooltips are active

Synopsis

```
#include <gtk/gtk.h>
void gtk_toolbar_set_tooltips(GtkToolbar * toolbar, gboolean enable);
```

Description

gtk_toolbar_set_tooltips() shall set whether the tooltips of *toolbar* are active. If the value of the parameter *enable* is `TRUE`, then the tooltips shall be made active; otherwise, they shall be made inactive.

This call may have no effect. The interface has been deprecated since version 2.14 of the GTK+ specification.

See Also

gtk_toolbar_get_tooltips()

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

XIII Qt Libraries

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23360-1-3:2021

18 Libraries

18.1 Introduction

An conforming implementation shall support the following Qt libraries which provide interfaces for creating rich user applications, either graphical or console.

QtCore

A general-purpose application development library, not specific to graphical user interfaces. QtCore provides a main loop, threading classes, tool classes, template container classes, IO classes, and so on.

QtGui

The GUI part of Qt, provides a paint engine, text rendering engine, widgets, printing, PDF generation, accessibility classes, and so on.

QtXml

The XML support for Qt, provides fast XML parsing and DOM classes to traverse the XML tree.

QtOpenGL

The OpenGL integration for Qt, provides OpenGL rendering and embedding.

QtSql

The database part of Qt, provides access to databases and models that connect databases into the model/view framework of QtGui.

QtSvg

Renders Svg files into any context, for displaying on the screen, printing or embedding in PDF.

QtNetwork

Provides network server and client classes, TCP/IP, UDP, Http, Ftp, Url parsing support, and so on.

There are three main parts to the definition of each of these libraries.

The "Interfaces" section defines the required library name and version, and the required public symbols (interfaces and global data), as well as symbol versions, if any.

The "Interface Definitions" section provides complete or partial definitions of certain interfaces where either this specification is the source specification, or where there are variations from the source specification. If an interface definition requires one or more header files, one of those headers shall include the function prototype for the interface.

For source definitions of interfaces which include a reference to a header file, the contents of such header files form a part of the specification. The "Data Definitions" section provides the binary-level details for the header files from the source specifications, such as values for macros and enumerated types, as well as structure layouts, sizes and padding, etc. These data definitions, although presented in the form of header files for convenience, should not be taken as representing complete header files, as they are a supplement to the source

specifications. Application developers should follow the guidelines of the source specifications when determining which header files need to be included to completely resolve all references.

Note: While the Data Definitions supplement the source specifications, this specification itself does not require conforming implementations to supply any header files.

18.2 Interfaces for libQtCore

Table 18-1 defines the library name and shared object name for the libQtCore library

Table 18-1 libQtCore Definition

Library:	libQtCore
SONAME:	libQtCore.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI
 [LSB] This Specification
 [QtCore] QtCore 4.2.0
 [QtXml] QtXml 4.2.0

18.2.1 Qt4 Core Functions

18.2.1.1 Class data for QDynamicPropertyChangeEvent

The virtual table for the QDynamicPropertyChangeEvent class is described by Table 18-2

Table 18-2 Primary vtable for QDynamicPropertyChangeEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDynamicPropertyChangeEvent
vfunc[0]:	QDynamicPropertyChangeEvent::~~QDynamicPropertyChangeEvent()
vfunc[1]:	QDynamicPropertyChangeEvent::~~QDynamicPropertyChangeEvent()

18.2.1.2 Class data for QTimeLine

The virtual table for the QTimeLine class is described by Table 18-3

Table 18-3 Primary vtable for QTimeLine

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTimeLine

vfunc[0]:	QTimeLine::metaObject() const
vfunc[1]:	QTimeLine::qt_metacast(char const*)
vfunc[2]:	QTimeLine::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTimeLine::~~QTimeLine()
vfunc[4]:	QTimeLine::~~QTimeLine()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTimeLine::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QTimeLine::valueForTime(int) const

18.2.1.3 Interfaces for Qt4 Core Functions

An LSB conforming implementation shall provide the generic functions for Qt4 Core Functions specified in Table 18-4, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-4 libQtCore - Qt4 Core Functions Function Interfaces

_Z11qt_assert_xPKcS0_S0_i [QtCore]	_Z12qInstallPathv [QtCore]
_Z12qSharedBuildv [QtCore]	_Z13qErrnoWarningPKcz [QtCore]
_Z13qErrnoWarningiPKcz [QtCore]	_Z14qSystemWarningPKci [QtCore]
_Z15qAddPostRoutinePFvvE [QtCore]	_Z15qt_error_stringi [QtCore]
_Z16qInstallPathBinsv [QtCore]	_Z16qInstallPathDatav [QtCore]
_Z16qInstallPathDocsv [QtCore]	_Z16qInstallPathLibsv [QtCore]
_Z16qt_check_pointerPKci [QtCore]	_Z17qt_message_output9QtMsgTypePKc [QtCore]
_Z18qInstallMsgHandlerPFv9QtMsgTypePKcE [QtCore]	_Z18qRemovePostRoutinePFvvE [QtCore]
_Z19qInstallPathHeadersv [QtCore]	_Z19qInstallPathPluginsv [QtCore]
_Z19qInstallPathSysconfv [QtCore]	_Z20qt_qFindChild_helperPK7QObjectRK7QStringRK11QMetaObject [QtCore]

_Z21qRegisterResourceDataIPKhS0_S0_ [QtCore]	_Z23qUnregisterResourceDataIPKhS0_S0_ [QtCore]
_Z23qt_qFindChildren_helperPK7QObjectRK7QStringPK7QRegExpRK11QMetaObjectP5QListIPvE [QtCore]	_Z24qInstallPathTranslationsv [QtCore]
_Z32qt_register_signal_spy_callbacksRK21QSignalSpyCallbackSet [QtCore]	_Z37qRegisterStaticPluginInstanceFunctionPFP7QObjectvE [QtCore]
_Z5qFreePv [LSB]	_Z5qHashRK10QByteArray [QtCore]
_Z5qHashRK7QString [QtCore]	_Z5qrandv [QtXml]
_Z6qDebugPKcz [QtCore]	_Z6qFatalPKcz [QtCore]
_Z6qsjrandj [QtXml]	_Z7qgetenvPKc [LSB]
Z7qstrcmpPKcS0 [QtCore]	_Z7qstrcpyPcPKc [QtCore]
_Z7qstrdupPKc [QtCore]	_Z8qAppNamev [QtCore]
_Z8qVersionv [QtCore]	_Z8qWarningPKcz [QtCore]
Z8qstrcmpPKcS0 [QtCore]	_Z8qstrncpyPcPKcj [QtCore]
_Z9qChecksumPKcj [QtCore]	_Z9qCriticalPKcz [QtCore]
_Z9qstrnicmpPKcS0_j [QtCore]	_Z9qt_assertPKcS0_i [QtCore]
_Zls6QDebug6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_Zls6QDebugN8QVariant4TypeE [QtCore]
_Zls6QDebugRK11QModelIndex [QtCore]	_ZlsR11QDataStreamN8QVariant4TypeE [QtCore]
_ZrsR11QDataStreamRN8QVariant4TypeE [QtCore]	

An LSB conforming implementation shall provide the generic data interfaces for Qt4 Core Functions specified in Table 18-5, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-5 libQtCore - Qt4 Core Functions Data Interfaces

_ZN10QByteArray11shared_nullE [QtCore]	_ZN10QEventLoop16staticMetaObjectE [QtCore]
_ZN11QTranslator16staticMetaObjectE [QtCore]	_ZN11QVectorData11shared_nullE [QtCore]
_ZN13QPluginLoader16staticMetaObjectE [QtCore]	_ZN13QSignalMapper16staticMetaObjectE [QtCore]
_ZN14QTemporaryFile16staticMetaObjectE [QtCore]	_ZN15QLinkedListData11shared_nullE [QtCore]
_ZN15QSocketNotifier16staticMetaObjectE [QtCore]	_ZN16QCoreApplication16staticMetaObjectE [QtCore]
_ZN16QCoreApplication4selfE [QtCore]	_ZN16QTextCodecPlugin16staticMetaObjectE [QtCore]

_ZN18QAbstractItemModel16staticMetaObjectE [QtCore]	_ZN18QAbstractListModel16staticMetaObjectE [QtCore]
_ZN18QFileSystemWatcher16staticMetaObjectE [QtXml]	_ZN19QAbstractTableModel16staticMetaObjectE [QtCore]
_ZN21QObjectCleanupHandler16staticMetaObjectE [QtCore]	_ZN24QAbstractEventDispatcher16staticMetaObjectE [QtCore]
_ZN5QFile16staticMetaObjectE [QtCore]	_ZN6QTimer16staticMetaObjectE [QtCore]
_ZN7QBuffer16staticMetaObjectE [QtCore]	_ZN7QObject16staticMetaObjectE [QtCore]
_ZN7QObject18staticQtMetaObjectE [QtCore]	_ZN7QString11shared_nullE [QtCore]
_ZN7QString16codecForCStringsE [QtCore]	_ZN7QString4nullE [QtCore]
_ZN7QThread16staticMetaObjectE [QtCore]	_ZN8QLibrary16staticMetaObjectE [QtCore]
_ZN8QMapData11shared_nullE [QtCore]	_ZN8QProcess16staticMetaObjectE [QtCore]
_ZN8QVariant7handlerE [QtCore]	_ZN9QHashData11shared_nullE [QtCore]
_ZN9QIODevice16staticMetaObjectE [QtCore]	_ZN9QListData11shared_nullE [QtCore]
_ZN9QMimeData16staticMetaObjectE [QtCore]	_ZN9QSettings16staticMetaObjectE [QtCore]
_ZN9QTimeLine16staticMetaObjectE [QtXml]	_ZTI10QEventLoop [CXXABI-1.86]
_ZTI10QTextCodec [CXXABI-1.86]	_ZTI11QChildEvent [CXXABI-1.86]
_ZTI11QDataStream [CXXABI-1.86]	_ZTI11QTextStream [CXXABI-1.86]
_ZTI11QTimerEvent [CXXABI-1.86]	_ZTI11QTranslator [CXXABI-1.86]
_ZTI12QCustomEvent [CXXABI-1.86]	_ZTI13QFSFileEngine [CXXABI-1.86]
_ZTI13QFontLaoCodec [CXXABI-1.86]	_ZTI13QPluginLoader [CXXABI-1.86]
_ZTI13QSignalMapper [CXXABI-1.86]	_ZTI13QSystemLocale [CXXABI-1.86]
_ZTI14QFactoryLoader [CXXABI-1.86]	_ZTI14QMetaCallEvent [CXXABI-1.86]
_ZTI14QObjectPrivate [CXXABI-1.86]	_ZTI14QTemporaryFile [CXXABI-1.86]
_ZTI15QDateTimeParser [CXXABI-1.86]	_ZTI15QObjectUserData [CXXABI-1.86]

_ZTI15QSocketNotifier [CXXABI-1.86]	_ZTI16QCoreApplication [CXXABI-1.86]
_ZTI16QIODevicePrivate [CXXABI-1.86]	_ZTI16QTextCodecPlugin [CXXABI-1.86]
_ZTI17QFactoryInterface [CXXABI-1.86]	_ZTI18QAbstractItemModel [CXXABI-1.86]
_ZTI18QAbstractListModel [CXXABI-1.86]	_ZTI18QFileSystemWatcher [CXXABI-1.86]
_ZTI19QAbstractFileEngine [CXXABI-1.86]	_ZTI19QAbstractTableModel [CXXABI-1.86]
_ZTI20QEventDispatcherUNIX [CXXABI-1.86]	_ZTI21QObjectCleanupHandler [CXXABI-1.86]
_ZTI23QCoreApplicationPrivate [CXXABI-1.86]	_ZTI24QAbstractEventDispatcher [CXXABI-1.86]
_ZTI26QAbstractFileEngineHandler [CXXABI-1.86]	_ZTI26QTextCodecFactoryInterface [CXXABI-1.86]
_ZTI27QDynamicPropertyChangeEvent [CXXABI-1.86]	_ZTI27QEventDispatcherUNIXPrivate [CXXABI-1.86]
_ZTI5QFile [CXXABI-1.86]	_ZTI6QEvent [CXXABI-1.86]
_ZTI6QTimer [CXXABI-1.86]	_ZTI7QBuffer [CXXABI-1.86]
_ZTI7QObject [CXXABI-1.86]	_ZTI7QThread [CXXABI-1.86]
_ZTI8QLibrary [CXXABI-1.86]	_ZTI8QProcess [CXXABI-1.86]
_ZTI9QIODevice [CXXABI-1.86]	_ZTI9QMimeData [CXXABI-1.86]
_ZTI9QSettings [CXXABI-1.86]	_ZTI9QTimeLine [CXXABI-1.86]
_ZTV10QEventLoop [CXXABI-1.86]	_ZTV10QTextCodec [CXXABI-1.86]
_ZTV11QChildEvent [CXXABI-1.86]	_ZTV11QDataStream [CXXABI-1.86]
_ZTV11QTextStream [CXXABI-1.86]	_ZTV11QTimerEvent [CXXABI-1.86]
_ZTV11QTranslator [CXXABI-1.86]	_ZTV12QCustomEvent [CXXABI-1.86]
_ZTV13QFSFileEngine [CXXABI-1.86]	_ZTV13QFontLaoCodec [CXXABI-1.86]
_ZTV13QPluginLoader [CXXABI-1.86]	_ZTV13QSignalMapper [CXXABI-1.86]
_ZTV13QSystemLocale [CXXABI-1.86]	_ZTV14QFactoryLoader [CXXABI-1.86]
_ZTV14QMetaCallEvent [CXXABI-1.86]	_ZTV14QObjectPrivate [CXXABI-1.86]
_ZTV14QTemporaryFile [CXXABI-1.86]	_ZTV15QDateTimeParser [CXXABI-1.86]

_ZTV15QObjectUserData [CXXABI-1.86]	_ZTV15QSocketNotifier [CXXABI-1.86]
_ZTV16QCoreApplication [CXXABI-1.86]	_ZTV16QIODevicePrivate [CXXABI-1.86]
_ZTV16QTextCodecPlugin [CXXABI-1.86]	_ZTV17QFactoryInterface [CXXABI-1.86]
_ZTV18QAbstractItemModel [CXXABI-1.86]	_ZTV18QAbstractListModel [CXXABI-1.86]
_ZTV18QFileSystemWatcher [CXXABI-1.86]	_ZTV19QAbstractFileEngine [CXXABI-1.86]
_ZTV19QAbstractTableModel [CXXABI-1.86]	_ZTV20QEventDispatcherUNIX [CXXABI-1.86]
_ZTV21QObjectCleanupHandler [CXXABI-1.86]	_ZTV23QCoreApplicationPrivate [CXXABI-1.86]
_ZTV24QAbstractEventDispatcher [CXXABI-1.86]	_ZTV26QAbstractFileEngineHandler [CXXABI-1.86]
_ZTV26QTextCodecFactoryInterface [CXXABI-1.86]	_ZTV27QDynamicPropertyChangeEvent [CXXABI-1.86]
_ZTV27QEventDispatcherUNIXPrivate [CXXABI-1.86]	_ZTV5QFile [CXXABI-1.86]
_ZTV6QEvent [CXXABI-1.86]	_ZTV6QTimer [CXXABI-1.86]
_ZTV7QBuffer [CXXABI-1.86]	_ZTV7QObject [CXXABI-1.86]
_ZTV7QThread [CXXABI-1.86]	_ZTV8QLibrary [CXXABI-1.86]
_ZTV8QProcess [CXXABI-1.86]	_ZTV9QIODevice [CXXABI-1.86]
_ZTV9QMimeData [CXXABI-1.86]	_ZTV9QSettings [CXXABI-1.86]
_ZTV9QTimeLine [CXXABI-1.86]	

18.2.2 Qt4 Models

18.2.2.1 Class data for QAbstractItemModel

The virtual table for the QAbstractItemModel class is described by Table 18-6

Table 18-6 Primary vtable for QAbstractItemModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractItemModel
vfunc[0]:	QAbstractItemModel::metaObject() const
vfunc[1]:	QAbstractItemModel::qt_metacast(char const*)

vfunc[2]:	QAbstractItemModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractItemModel::~QAbstractItemModel()
vfunc[4]:	QAbstractItemModel::~QAbstractItemModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	QAbstractItemModel::hasChildren(QModelIndex const&) const
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	QAbstractItemModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QAbstractItemModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QAbstractItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractItemModel::dropMimeData(QMimeData const*,

	Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QAbstractItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QAbstractItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QAbstractItemModel::flags(QModelIndex const&) const
vfunc[34]:	QAbstractItemModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()

The Run-Time Type Information for the QAbstractItemModel class is described by Table 18-7

Table 18-7 typeinfo for QAbstractItemModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractItemModel
basetype:	typeinfo for QObject

18.2.2.2 Class data for QAbstractTableModel

The virtual table for the QAbstractTableModel class is described by Table 18-8

Table 18-8 Primary vtable for QAbstractTableModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractTableModel
vfunc[0]:	QAbstractTableModel::metaObject() const
vfunc[1]:	QAbstractTableModel::qt_metacast(char const*)
vfunc[2]:	QAbstractTableModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractTableModel::~QAbstractTableModel()
vfunc[4]:	QAbstractTableModel::~QAbstractTableModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractTableModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QAbstractTableModel::parent(QModelIndex const&) const
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	QAbstractTableModel::hasChildren(QModelIndex const&) const
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	QAbstractItemModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QAbstractItemModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QAbstractItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)

vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractTableModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QAbstractItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QAbstractItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QAbstractItemModel::flags(QModelIndex const&) const
vfunc[34]:	QAbstractItemModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()

The Run Time Type Information for the QAbstractTableModel class is described by Table 18-9

Table 18-9 typeinfo for QAbstractTableModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractTableModel
basetype:	typeinfo for QAbstractItemModel

18.2.2.3 Class data for QAbstractListModel

The virtual table for the QAbstractListModel class is described by Table 18-10

Table 18-10 Primary vtable for QAbstractListModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractListModel
vfunc[0]:	QAbstractListModel::metaObject() const
vfunc[1]:	QAbstractListModel::qt_metacast(ch ar const*)
vfunc[2]:	QAbstractListModel::qt_metacall(Q MetaObject::Call, int, void**)
vfunc[3]:	QAbstractListModel::~~QAbstractList Model()
vfunc[4]:	QAbstractListModel::~~QAbstractList Model()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractListModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QAbstractListModel::parent(QModel Index const&) const
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	QAbstractListModel::columnCount(QModelIndex const&) const

vfunc[16]:	QAbstractListModel::hasChildren(QModelIndex const&) const
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	QAbstractItemModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QAbstractItemModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QAbstractItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractListModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QAbstractItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QAbstractItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QAbstractItemModel::flags(QModelIndex const&) const
vfunc[34]:	QAbstractItemModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const

vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()

The Run Time Type Information for the QAbstractListModel class is described by Table 18-11

Table 18-11 typeinfo for QAbstractListModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractListModel
basetype:	typeinfo for QAbstractItemModel

18.2.2.4 Interfaces for Qt4 Models

An LSB conforming implementation shall provide the generic functions for Qt4 Models specified in Table 18-12, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-12 libQtCore - Qt4 Models Function Interfaces

_ZN18QAbstractItemModel10decodeDataEiiRK11QModelIndexR11QDataStream [QtCore]	_ZN18QAbstractItemModel10insertRowsEiiRK11QModelIndex [QtCore]
_ZN18QAbstractItemModel10removeRowsEiiRK11QModelIndex [QtCore]	_ZN18QAbstractItemModel11dataChangedERK11QModelIndexS2_ [QtCore]
_ZN18QAbstractItemModel11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN18QAbstractItemModel11qt_metacastEPKc [QtCore]
_ZN18QAbstractItemModel11setItemDataERK11QModelIndexRK4QMapli8QVariantE [QtCore]	_ZN18QAbstractItemModel12dropMimeTypeDataEPK9QMimeDataN2Qt10DropActionEiiRK11QModelIndex [QtCore]
_ZN18QAbstractItemModel13endInsertRowsEv [QtCore]	_ZN18QAbstractItemModel13endRemoveRowsEv [QtCore]
_ZN18QAbstractItemModel13insertColumnsEiiRK11QModelIndex [QtCore]	_ZN18QAbstractItemModel13layoutChangedEv [QtCore]
_ZN18QAbstractItemModel13removeColumnsEiiRK11QModelIndex [QtCore]	_ZN18QAbstractItemModel13setHeaderDataEiN2Qt11OrientationERK8QVarianti [QtCore]

_ZN18QAbstractItemModel15beginInsertRowsERK11QModelIndex [QtCore]	_ZN18QAbstractItemModel15beginRemoveRowsERK11QModelIndex [QtCore]
_ZN18QAbstractItemModel16endInsertColumnsEv [QtCore]	_ZN18QAbstractItemModel16endRemoveColumnsEv [QtCore]
_ZN18QAbstractItemModel17headerDataChangedEN2Qt11OrientationEii [QtCore]	_ZN18QAbstractItemModel18beginInsertColumnsERK11QModelIndex [QtCore]
_ZN18QAbstractItemModel18beginRemoveColumnsERK11QModelIndex [QtCore]	_ZN18QAbstractItemModel21changePersistentIndexERK11QModelIndexS2_ [QtCore]
_ZN18QAbstractItemModel22layoutAboutToBeChangedEv [QtXml]	_ZN18QAbstractItemModel23setSupportedDragActionsE6QFlagsIN2Qt10DropActionEE [QtXml]
ZN18QAbstractItemModel25changePersistentIndexListERK5QListI11QModelIndexES4 [QtCore]	_ZN18QAbstractItemModel4sortEiN2Qt9SortOrderE [QtCore]
_ZN18QAbstractItemModel5resetEv [QtCore]	_ZN18QAbstractItemModel6revertEv [QtCore]
_ZN18QAbstractItemModel6submitEv [QtCore]	_ZN18QAbstractItemModel7setDataERK11QModelIndexRK8QVariant [QtCore]
_ZN18QAbstractItemModel9fetchMoreERK11QModelIndex [QtCore]	_ZN18QAbstractItemModelC1EP7QObject [QtCore]
_ZN18QAbstractItemModelC2EP7QObject [QtCore]	_ZN18QAbstractItemModelD0Ev [QtCore]
_ZN18QAbstractItemModelD1Ev [QtCore]	_ZN18QAbstractItemModelD2Ev [QtCore]
_ZN18QAbstractListModel11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN18QAbstractListModel11qt_metacastEPKc [QtCore]
_ZN18QAbstractListModel12dropMimeTypeDataEPK9QMimeTypeDataN2Qt10DropActionEiiRK11QModelIndex [QtCore]	_ZN18QAbstractListModelC1EP7QObject [QtCore]
_ZN18QAbstractListModelC2EP7QObject [QtCore]	_ZN18QAbstractListModelD0Ev [QtCore]
_ZN18QAbstractListModelD1Ev [QtCore]	_ZN18QAbstractListModelD2Ev [QtCore]
_ZN19QAbstractTableModel11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN19QAbstractTableModel11qt_metacastEPKc [QtCore]
_ZN19QAbstractTableModel12dropMimeTypeDataEPK9QMimeTypeDataN2Qt10DropActionEiiRK11QModelIndex [QtCore]	_ZN19QAbstractTableModelC1EP7QObject [QtCore]

DropActionEiiRK11QModelIndex [QtCore]	
_ZN19QAbstractTableModelC2EP7Q Object [QtCore]	_ZN19QAbstractTableModelD0Ev [QtCore]
_ZN19QAbstractTableModelD1Ev [QtCore]	_ZN19QAbstractTableModelD2Ev [QtCore]
_ZN21QPersistentModelIndexC1ER K11QModelIndex [QtCore]	_ZN21QPersistentModelIndexC1ER KS_ [QtCore]
_ZN21QPersistentModelIndexC1Ev [QtCore]	_ZN21QPersistentModelIndexC2ER K11QModelIndex [QtCore]
ZN21QPersistentModelIndexC2ER KS [QtCore]	_ZN21QPersistentModelIndexC2Ev [QtCore]
_ZN21QPersistentModelIndexD1Ev [QtCore]	_ZN21QPersistentModelIndexD2Ev [QtCore]
_ZN21QPersistentModelIndexaSERK 11QModelIndex [QtCore]	_ZN21QPersistentModelIndexaSERK S_ [QtCore]
_ZNK18QAbstractItemModel10enco deDataERK5QList11QModelIndexE R11QDataStream [QtCore]	_ZNK18QAbstractItemModel10head erDataEiN2Qt11OrientationEi [QtCore]
_ZNK18QAbstractItemModel10meta ObjectEv [QtCore]	_ZNK18QAbstractItemModel11hasC hildrenERK11QModelIndex [QtCore]
_ZNK18QAbstractItemModel12canF etchMoreERK11QModelIndex [QtCore]	_ZNK18QAbstractItemModel19persi stentIndexListEv [QtXml]
_ZNK18QAbstractItemModel20supp ortedDragActionsEv [QtXml]	_ZNK18QAbstractItemModel20supp ortedDropActionsEv [QtCore]
_ZNK18QAbstractItemModel4spanE RK11QModelIndex [QtCore]	_ZNK18QAbstractItemModel5buddy ERK11QModelIndex [QtCore]
_ZNK18QAbstractItemModel5flagsE RK11QModelIndex [QtCore]	_ZNK18QAbstractItemModel5match ERK11QModelIndexiRK8QVarianti6 QFlagsIN2Qt9MatchFlagEE [QtCore]
_ZNK18QAbstractItemModel8hasIn dexEiiRK11QModelIndex [QtCore]	_ZNK18QAbstractItemModel8itemD ataERK11QModelIndex [QtCore]
_ZNK18QAbstractItemModel8mime DataERK5QList11QModelIndexE [QtCore]	_ZNK18QAbstractItemModel9mime TypesEv [QtCore]
_ZNK18QAbstractListModel10meta ObjectEv [QtCore]	_ZNK18QAbstractListModel11colum nCountERK11QModelIndex [QtCore]
_ZNK18QAbstractListModel11hasCh ildrenERK11QModelIndex [QtCore]	_ZNK18QAbstractListModel5indexE iiRK11QModelIndex [QtCore]
_ZNK18QAbstractListModel6parent ERK11QModelIndex [QtCore]	_ZNK19QAbstractTableModel10met aObjectEv [QtCore]

_Znk19QAbstractTableModel11hasChildrenERK11QModelIndex [QtCore]	_Znk19QAbstractTableModel5indexEiiRK11QModelIndex [QtCore]
_Znk19QAbstractTableModel6parentERK11QModelIndex [QtCore]	_Znk21QPersistentModelIndex10internalIdEv [LSB]
_Znk21QPersistentModelIndex15internalPointerEv [LSB]	_Znk21QPersistentModelIndex3rowEv [QtCore]
_Znk21QPersistentModelIndex4dataEi [QtCore]	_Znk21QPersistentModelIndex5childEii [QtCore]
_Znk21QPersistentModelIndex5flagsEv [QtXml]	_Znk21QPersistentModelIndex5modelEv [QtCore]
_Znk21QPersistentModelIndex6columnEv [QtCore]	_Znk21QPersistentModelIndex6parentEv [QtCore]
_Znk21QPersistentModelIndex7isValidEv [QtCore]	_Znk21QPersistentModelIndex7siblingEii [QtCore]
_Znk21QPersistentModelIndexcvRK11QModelIndexEv [QtCore]	_Znk21QPersistentModelIndexeqRK11QModelIndex [QtCore]
Znk21QPersistentModelIndexeqERKS [QtCore]	_Znk21QPersistentModelIndexltERKS_ [QtCore]
_Znk21QPersistentModelIndexneERK11QModelIndex [QtCore]	_Zls6QDebugRK21QPersistentModelIndex [QtCore]

18.2.3 Qt4 Internationalization

18.2.3.1 Class data for QTextCodec

The virtual table for the QTextCodec class is described by Table 18-13

Table 18-13 Primary vtable for QTextCodec

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextCodec
vfunc[0]:	__cxa_pure_virtual
vfunc[1]:	QTextCodec::aliases() const
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	QTextCodec::~~QTextCodec()
vfunc[6]:	QTextCodec::~~QTextCodec()

The Run Time Type Information for the QTextCodec class is described by Table 18-14

Table 18-14 typeinfo for QTextCodec

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTextCodec

18.2.3.2 Class data for QTextCodecFactoryInterface

The virtual table for the QTextCodecFactoryInterface class is described by Table 18-15

Table 18-15 Primary vtable for QTextCodecFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextCodecFactoryInterface
vfunc[0]:	NULL or QTextCodecFactoryInterface::~~QText CodecFactoryInterface()
vfunc[1]:	NULL or QTextCodecFactoryInterface::~~QText CodecFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QTextCodecFactoryInterface class is described by Table 18-16

Table 18-16 typeinfo for QTextCodecFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextCodecFactoryInterface
basetype:	typeinfo for QFactoryInterface

18.2.3.3 Class data for QTranslator

The virtual table for the QTranslator class is described by Table 18-17

Table 18-17 Primary vtable for QTranslator

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTranslator
vfunc[0]:	QTranslator::metaObject() const
vfunc[1]:	QTranslator::qt_metacast(char const*)

vfunc[2]:	QTranslator::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTranslator::~~QTranslator()
vfunc[4]:	QTranslator::~~QTranslator()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QTranslator::translate(char const*, char const*, char const*) const
vfunc[13]:	QTranslator::isEmpty() const

The Run Time Type Information for the QTranslator class is described by Table 18-18

Table 18-18 typeinfo for QTranslator

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTranslator
basetype:	typeinfo for QObject

18.2.3.4 Class data for QSystemLocale

The virtual table for the QSystemLocale class is described by Table 18-19

Table 18-19 Primary vtable for QSystemLocale

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSystemLocale
vfunc[0]:	QSystemLocale::~~QSystemLocale()
vfunc[1]:	QSystemLocale::~~QSystemLocale()
vfunc[2]:	QSystemLocale::query(QSystemLocale::QueryType, QVariant) const
vfunc[3]:	QSystemLocale::fallbackLocale() const

18.2.3.5 Interfaces for Qt4 Internationalization

An LSB conforming implementation shall provide the generic functions for Qt4 Internationalization specified in Table 18-20, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-20 libQtCore - Qt4 Internationalization Function Interfaces

_ZN10QTextCodec11codecForMibEi [QtCore]	_ZN10QTextCodec12codecForHtmlE RK10QByteArray [QtCore]
_ZN10QTextCodec12codecForName ERK10QByteArray [QtCore]	_ZN10QTextCodec13availableMibsEv [QtCore]
_ZN10QTextCodec14codecForLocale Ev [QtCore]	_ZN10QTextCodec15availableCodec sEv [QtCore]
ZN10QTextCodec17setCodecForLoc aleEPS [QtCore]	_ZN10QTextCodec6localeEv [QtCore]
_ZN10QTextCodecC1Ev [QtCore]	_ZN10QTextCodecC2Ev [QtCore]
_ZN10QTextCodecD0Ev [QtCore]	_ZN10QTextCodecD1Ev [QtCore]
_ZN10QTextCodecD2Ev [QtCore]	_ZN11QTranslator11qt_metacallEN1 1QMetaObject4CallEiPPv [QtCore]
_ZN11QTranslator11qt_metacastEPK c [QtCore]	_ZN11QTranslator4loadEPKHi [QtCore]
_ZN11QTranslator4loadERK7QStrin gS2_S2_S2_ [QtCore]	_ZN11QTranslatorC1EP7QObject [QtCore]
_ZN11QTranslatorC1EP7QObjectPK c [QtCore]	_ZN11QTranslatorC2EP7QObject [QtCore]
_ZN11QTranslatorC2EP7QObjectPK c [QtCore]	_ZN11QTranslatorD0Ev [QtCore]
_ZN11QTranslatorD1Ev [QtCore]	_ZN11QTranslatorD2Ev [QtCore]
_ZN12QTextDecoder9toUnicodeEPK ci [QtCore]	_ZN12QTextDecoder9toUnicodeERK 10QByteArray [QtCore]
_ZN12QTextDecoderD1Ev [QtCore]	_ZN12QTextDecoderD2Ev [QtCore]
_ZN12QTextEncoder11fromUnicode EPK5QChari [QtCore]	_ZN12QTextEncoder11fromUnicode ERK7QString [QtCore]
_ZN12QTextEncoder11fromUnicode ERK7QStringRi [QtCore]	_ZN12QTextEncoderD1Ev [QtCore]
_ZN12QTextEncoderD2Ev [QtCore]	_ZN13QSystemLocaleC1Ev [QtXml]
_ZN13QSystemLocaleC2Ev [QtXml]	_ZN13QSystemLocaleD0Ev [QtXml]
_ZN13QSystemLocaleD1Ev [QtXml]	_ZN13QSystemLocaleD2Ev [QtXml]
ZN7QLocale10setDefaultERKS [QtCore]	_ZN7QLocale15countryToStringENS _7CountryE [QtCore]
_ZN7QLocale16languageToStringENS _8LanguageE [QtCore]	_ZN7QLocale16setNumberOptionsE 6QFlagsINS_12NumberOptionEE [QtXml]

_ZN7QLocale6systemEv [QtCore]	_ZN7QLocaleC1ENS_8LanguageENS_7CountryE [QtCore]
_ZN7QLocaleC1ERK7QString [QtCore]	_ZN7QLocaleC1ERKS_ [QtCore]
_ZN7QLocaleC1Ev [QtCore]	_ZN7QLocaleC2ENS_8LanguageENS_7CountryE [QtCore]
_ZN7QLocaleC2ERK7QString [QtCore]	_ZN7QLocaleC2ERKS_ [QtCore]
_ZN7QLocaleC2Ev [QtCore]	_ZN7QLocaleaSERKS_ [QtCore]
_ZNK10QTextCodec11fromUnicodeERK7QString [QtCore]	_ZNK10QTextCodec11fromUnicodeERK7QStringRi [QtCore]
_ZNK10QTextCodec11makeDecoderEv [QtCore]	_ZNK10QTextCodec11makeEncoderEv [QtCore]
_ZNK10QTextCodec7aliasesEv [QtCore]	_ZNK10QTextCodec9canEncodeE5QChar [QtCore]
_ZNK10QTextCodec9canEncodeERK7QString [QtCore]	_ZNK10QTextCodec9toUnicodeEPKc [QtCore]
_ZNK10QTextCodec9toUnicodeERK10QByteArray [QtCore]	_ZNK10QTextCodec9toUnicodeERK10QByteArrayi [QtCore]
_ZNK11QTranslator10metaObjectEv [QtCore]	_ZNK11QTranslator7isEmptyEv [QtCore]
_ZNK11QTranslator9translateEPKcS1_S1_ [QtCore]	_ZNK11QTranslator9translateEPKcS1_S1_i [QtXml]
_ZNK13QSystemLocale14fallbackLocaleEv [QtXml]	_ZNK13QSystemLocale5queryENS_9QueryTypeE8QVariant [QtXml]
_ZNK7QLocale10dateFormatENS_10FormatTypeE [QtCore]	_ZNK7QLocale10timeFormatENS_10FormatTypeE [QtCore]
_ZNK7QLocale10toLongLongERK7QStringPbi [QtCore]	_ZNK7QLocale11exponentialEv [QtCore]
_ZNK7QLocale11toULongLongERK7QStringPbi [QtCore]	_ZNK7QLocale12decimalPointEv [QtCore]
_ZNK7QLocale12negativeSignEv [QtCore]	_ZNK7QLocale13numberOptionsEv [QtXml]
_ZNK7QLocale14groupSeparatorEv [QtCore]	_ZNK7QLocale4nameEv [QtCore]
_ZNK7QLocale5toIntERK7QStringPbi [QtCore]	_ZNK7QLocale6toUIntERK7QStringPbi [QtCore]
_ZNK7QLocale7countryEv [QtCore]	_ZNK7QLocale7dayNameEiNS_10FormatTypeE [QtXml]
_ZNK7QLocale7percentEv [QtCore]	_ZNK7QLocale7toFloatERK7QStringPb [QtCore]

_ZNK7QLocale7toShortERK7QStringPbi [QtCore]	_ZNK7QLocale8languageEv [QtCore]
_ZNK7QLocale8toDoubleERK7QStringPb [QtCore]	_ZNK7QLocale8toStringERK5QDateNS_10FormatTypeE [QtCore]
_ZNK7QLocale8toStringERK5QDateRK7QString [QtCore]	_ZNK7QLocale8toStringERK5QTimeNS_10FormatTypeE [QtCore]
_ZNK7QLocale8toStringERK5QTimeRK7QString [QtCore]	_ZNK7QLocale8toStringEdci [QtCore]
_ZNK7QLocale8toStringEx [QtCore]	_ZNK7QLocale8toStringEy [QtCore]
_ZNK7QLocale8toUShortERK7QStringPbi [QtCore]	_ZNK7QLocale9monthNameEiNS_10FormatTypeE [QtXml]
_ZNK7QLocale9zeroDigitEv [QtCore]	_ZlsR11QDataStreamRK7QLocale [QtCore]
_ZrsR11QDataStreamR7QLocale [QtCore]	

18.2.4 Qt4 Containers

18.2.4.1 Class data for QMimeData

The virtual table for the QMimeData class is described by Table 18-21

Table 18-21 Primary vtable for QMimeData

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMimeData
vfunc[0]:	QMimeData::metaObject() const
vfunc[1]:	QMimeData::qt_metacast(char const*)
vfunc[2]:	QMimeData::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMimeData::~QMimeData()
vfunc[4]:	QMimeData::~QMimeData()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QMimeType::hasFormat(QString const&) const
vfunc[13]:	QMimeType::formats() const
vfunc[14]:	QMimeType::retrieveData(QString const&, QVariant::Type) const

The Run Time Type Information for the QMimeType class is described by Table 18-22

Table 18-22 typeidinfo for QMimeType

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeidinfo name for QMimeType
basetype:	typeidinfo for QObject

18.2.4.2 Interfaces for Qt4 Containers

An LSB conforming implementation shall provide the generic functions for Qt4 Containers specified in Table 18-23, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-23 libQtCore - Qt4 Containers Function Interfaces

_ZN11QVectorData4growEiiiib [QtCore]	_ZN11QVectorData6mallocEiiiPS_ [QtCore]
_ZN8QMapData10createDataEv [QtCore]	_ZN8QMapData11node_createEPPNS_4NodeEi [QtCore]
_ZN8QMapData11node_deleteEPPNS_4NodeEiS1_ [QtCore]	_ZN8QMapData16continueFreeDataEi [QtCore]
_ZN9QHashData12allocateNodeEv [LSB]	_ZN9QHashData12previousNodeEPNS_4NodeE [QtCore]
_ZN9QHashData13detach_helperEPFvPNS_4NodeEPvEi [QtCore]	_ZN9QHashData14destroyAndFreeEv [QtCore]
_ZN9QHashData6rehashEi [QtCore]	_ZN9QHashData8freeNodeEPv [LSB]
_ZN9QHashData8nextNodeEPNS_4NodeE [QtCore]	_ZN9QListData4moveEii [QtCore]
_ZN9QListData5eraseEPPv [QtCore]	_ZN9QListData6appendERKS_ [QtCore]
_ZN9QListData6appendEv [QtCore]	_ZN9QListData6detachEv [QtCore]
_ZN9QListData6insertEi [QtCore]	_ZN9QListData6removeEi [QtCore]
_ZN9QListData6removeEii [QtCore]	_ZN9QListData7prependEv [QtCore]

_ZN9QListData7reallocEi [QtCore]	_ZN9QMimeData11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN9QMimeData11qt_metacastEPKc [QtCore]	_ZN9QMimeData12setColorDataERK8QVariant [QtCore]
_ZN9QMimeData12setImageDataERK8QVariant [QtCore]	_ZN9QMimeData5clearEv [QtCore]
_ZN9QMimeData7setDataERK7QStringRK10QByteArray [QtCore]	_ZN9QMimeData7setHtmlERK7QString [QtCore]
_ZN9QMimeData7setTextERK7QString [QtCore]	_ZN9QMimeData7setUrlsERK5QListI4QUrlE [QtCore]
_ZN9QMimeDataC1Ev [QtCore]	_ZN9QMimeDataC2Ev [QtCore]
_ZN9QMimeDataD0Ev [QtCore]	_ZN9QMimeDataD1Ev [QtCore]
_ZN9QMimeDataD2Ev [QtCore]	_ZNK9QMimeData10metaObjectEv [QtCore]
_ZNK9QMimeData12retrieveDataERK7QStringN8QVariant4TypeE [QtCore]	_ZNK9QMimeData4dataERK7QString [QtCore]
_ZNK9QMimeData4htmlEv [QtCore]	_ZNK9QMimeData4textEv [QtCore]
_ZNK9QMimeData4urlsEv [QtCore]	_ZNK9QMimeData7formatsEv [QtCore]
_ZNK9QMimeData7hasHtmlEv [QtCore]	_ZNK9QMimeData7hasTextEv [QtCore]
_ZNK9QMimeData7hasUrlsEv [QtCore]	_ZNK9QMimeData8hasColorEv [QtCore]
_ZNK9QMimeData8hasImageEv [QtCore]	_ZNK9QMimeData9colorDataEv [QtCore]
_ZNK9QMimeData9hasFormatERK7QString [QtCore]	_ZNK9QMimeData9imageDataEv [QtCore]

18.2.5 Qt4 Events

18.2.5.1 Class data for QEventLoop

The virtual table for the QEventLoop class is described by Table 18-24

Table 18-24 Primary vtable for QEventLoop

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QEventLoop
vfunc[0]:	QEventLoop::metaObject() const
vfunc[1]:	QEventLoop::qt_metacast(char const*)

vfunc[2]:	QEventLoop::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QEventLoop::~~QEventLoop()
vfunc[4]:	QEventLoop::~~QEventLoop()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QEventLoop class is described by Table 18-25

Table 18-25 typeinfo for QEventLoop

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QEventLoop
basetype:	typeinfo for QObject

18.2.5.2 Class data for QEvent

The virtual table for the QEvent class is described by Table 18-26

Table 18-26 Primary vtable for QEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QEvent
vfunc[0]:	QEvent::~~QEvent()
vfunc[1]:	QEvent::~~QEvent()

The Run Time Type Information for the QEvent class is described by Table 18-27

Table 18-27 typeinfo for QEvent

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QEvent

18.2.5.3 Class data for QTimerEvent

The virtual table for the QTimerEvent class is described by Table 18-28

Table 18-28 Primary vtable for QTimerEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTimerEvent
vfunc[0]:	QTimerEvent::~~QTimerEvent()
vfunc[1]:	QTimerEvent::~~QTimerEvent()

The Run Time Type Information for the QTimerEvent class is described by Table 18-29

Table 18-29 typeinfo for QTimerEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTimerEvent
basetype:	typeinfo for QEvent

18.2.5.4 Class data for QChildEvent

The virtual table for the QChildEvent class is described by Table 18-30

Table 18-30 Primary vtable for QChildEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QChildEvent
vfunc[0]:	QChildEvent::~~QChildEvent()
vfunc[1]:	QChildEvent::~~QChildEvent()

The Run Time Type Information for the QChildEvent class is described by Table 18-31

Table 18-31 typeinfo for QChildEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QChildEvent
basetype:	typeinfo for QEvent

18.2.5.5 Class data for QCustomEvent

The virtual table for the QCustomEvent class is described by Table 18-32

Table 18-32 Primary vtable for QCustomEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCustomEvent

vfunc[0]:	QCustomEvent::~~QCustomEvent()
vfunc[1]:	QCustomEvent::~~QCustomEvent()

The Run Time Type Information for the QCustomEvent class is described by Table 18-33

Table 18-33 typeinfo for QCustomEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QCustomEvent
basetype:	typeinfo for QEvent

18.2.5.6 Class data for QAbstractEventDispatcher

The virtual table for the QAbstractEventDispatcher class is described by Table 18-34

Table 18-34 Primary vtable for QAbstractEventDispatcher

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractEventDispatcher
vfunc[0]:	QAbstractEventDispatcher::metaObject() const
vfunc[1]:	QAbstractEventDispatcher::qt_metacast(char const*)
vfunc[2]:	QAbstractEventDispatcher::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractEventDispatcher::~~QAbstractEventDispatcher()
vfunc[4]:	QAbstractEventDispatcher::~~QAbstractEventDispatcher()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual

vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	__cxa_pure_virtual
vfunc[19]:	__cxa_pure_virtual
vfunc[20]:	__cxa_pure_virtual
vfunc[21]:	__cxa_pure_virtual
vfunc[22]:	__cxa_pure_virtual
vfunc[23]:	QAbstractEventDispatcher::startingUp()
vfunc[24]:	QAbstractEventDispatcher::closingDown()

The Run Time Type Information for the QAbstractEventDispatcher class is described by Table 18-35

Table 18-35 typeid for QAbstractEventDispatcher

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeid name for QAbstractEventDispatcher
basetype:	typeid for QObject

18.2.5.7 Interfaces for Qt4 Events

An LSB conforming implementation shall provide the generic functions for Qt4 Events specified in Table 18-36, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-36 libQtCore - Qt4 Events Function Interfaces

_ZN10QEventLoop11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN10QEventLoop11qt_metacastEPKc [QtCore]
_ZN10QEventLoop13processEventsE6QFlagsINS_17ProcessEventsFlagEE [QtCore]	_ZN10QEventLoop13processEventsE6QFlagsINS_17ProcessEventsFlagEEi [QtCore]
_ZN10QEventLoop4execE6QFlagsINS_17ProcessEventsFlagEE [QtCore]	_ZN10QEventLoop4exitEi [QtCore]
_ZN10QEventLoop4quitEv [QtCore]	_ZN10QEventLoop6wakeUpEv [QtCore]
_ZN10QEventLoopC1EP7QObject [QtCore]	_ZN10QEventLoopC2EP7QObject [QtCore]

_ZN10QEventLoopD0Ev [QtCore]	_ZN10QEventLoopD1Ev [QtCore]
_ZN10QEventLoopD2Ev [QtCore]	_ZN11QChildEventC1EN6QEvent4TypeEP7QObject [QtCore]
_ZN11QChildEventC2EN6QEvent4TypeEP7QObject [QtCore]	_ZN11QChildEventD0Ev [QtCore]
_ZN11QChildEventD1Ev [QtCore]	_ZN11QChildEventD2Ev [QtCore]
_ZN11QTimerEventC1Ei [QtCore]	_ZN11QTimerEventC2Ei [QtCore]
_ZN11QTimerEventD0Ev [QtCore]	_ZN11QTimerEventD1Ev [QtCore]
_ZN11QTimerEventD2Ev [QtCore]	_ZN12QCustomEventC1EiPv [QtCore]
_ZN12QCustomEventC2EiPv [QtCore]	_ZN12QCustomEventD0Ev [QtCore]
_ZN12QCustomEventD1Ev [QtCore]	_ZN12QCustomEventD2Ev [QtCore]
_ZN24QAbstractEventDispatcher10startingUpEv [QtCore]	_ZN24QAbstractEventDispatcher11closingDownEv [QtCore]
_ZN24QAbstractEventDispatcher11filterEventEPv [QtCore]	_ZN24QAbstractEventDispatcher11qt_metacallEN11QMetaObject4CallEiPv [QtCore]
_ZN24QAbstractEventDispatcher11qt_metacastEPKc [QtCore]	_ZN24QAbstractEventDispatcher12aboutToBlockEv [QtCore]
_ZN24QAbstractEventDispatcher13registerTimerEiP7QObject [QtCore]	_ZN24QAbstractEventDispatcher14setEventFilterEPFbPvE [QtCore]
_ZN24QAbstractEventDispatcher5awakeEv [QtCore]	_ZN24QAbstractEventDispatcher8instanceEP7QThread [QtCore]
_ZN24QAbstractEventDispatcherC1EP7QObject [QtCore]	_ZN24QAbstractEventDispatcherC2EP7QObject [QtCore]
_ZN24QAbstractEventDispatcherD0Ev [QtCore]	_ZN24QAbstractEventDispatcherD1Ev [QtCore]
_ZN24QAbstractEventDispatcherD2Ev [QtCore]	_ZN27QDynamicPropertyChangeEventC1ERK10QByteArray [QtXml]
_ZN27QDynamicPropertyChangeEventC2ERK10QByteArray [QtXml]	_ZN27QDynamicPropertyChangeEventD0Ev [QtXml]
_ZN27QDynamicPropertyChangeEventD1Ev [QtXml]	_ZN27QDynamicPropertyChangeEventD2Ev [QtXml]
_ZN6QEventC1ENS_4TypeE [QtCore]	_ZN6QEventC2ENS_4TypeE [QtCore]
_ZN6QEventD0Ev [QtCore]	_ZN6QEventD1Ev [QtCore]
_ZN6QEventD2Ev [QtCore]	_ZNK10QEventLoop10metaObjectEv [QtCore]
_ZNK10QEventLoop9isRunningEv [QtCore]	_ZNK24QAbstractEventDispatcher10metaObjectEv [QtCore]

18.2.6 Qt4 Arrays

18.2.6.1 Interfaces for Qt4 Arrays

An LSB conforming implementation shall provide the generic functions for Qt4 Arrays specified in Table 18-37, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-37 libQtCore - Qt4 Arrays Function Interfaces

_Z11qUncompressPKhi [QtCore]	_Z9qCompressPKhii [QtCore]
ZN10QByteArray10fromBase64ERKS [QtCore]	_ZN10QByteArray11fromRawDataEPKci [QtCore]
_ZN10QByteArray4chopEi [QtCore]	_ZN10QByteArray4fillEci [QtCore]
_ZN10QByteArray5clearEv [QtCore]	_ZN10QByteArray6appendEPKc [QtCore]
ZN10QByteArray6appendERKS [QtCore]	_ZN10QByteArray6appendEc [QtCore]
_ZN10QByteArray6expandEi [QtCore]	_ZN10QByteArray6insertEiPKc [QtCore]
ZN10QByteArray6insertEiRKS [QtCore]	_ZN10QByteArray6insertEic [QtCore]
_ZN10QByteArray6numberEdci [QtCore]	_ZN10QByteArray6numberEii [QtCore]
_ZN10QByteArray6numberEji [QtCore]	_ZN10QByteArray6numberExi [QtCore]
_ZN10QByteArray6numberEyi [QtCore]	_ZN10QByteArray6removeEii [QtCore]
_ZN10QByteArray6resizeEi [QtCore]	_ZN10QByteArray6setNumEdci [QtCore]
_ZN10QByteArray6setNumExi [QtCore]	_ZN10QByteArray6setNumEyi [QtCore]
_ZN10QByteArray7prependEPKc [QtCore]	_ZN10QByteArray7prependERKS_ [QtCore]
_ZN10QByteArray7prependEc [QtCore]	_ZN10QByteArray7reallocEi [QtCore]
_ZN10QByteArray7replaceERKS_S1_ [QtCore]	_ZN10QByteArray7replaceEcRKS_ [QtCore]
_ZN10QByteArray7replaceEcc [QtCore]	_ZN10QByteArray7replaceEiiRKS_ [QtCore]
_ZN10QByteArray8truncateEi [QtCore]	_ZN10QByteArrayC1EPKc [QtCore]
_ZN10QByteArrayC1EPKci [QtCore]	_ZN10QByteArrayC1Eic [QtCore]
_ZN10QByteArrayC2EPKc [QtCore]	_ZN10QByteArrayC2EPKci [QtCore]

_ZN10QByteArrayC2Eic [QtCore]	_ZN10QByteArrayaSEPKc [QtCore]
ZN10QByteArrayaSERKS [QtCore]	_ZN17QByteArrayMatcher10setPatternERK10QByteArray [QtCore]
_ZN17QByteArrayMatcherC1ERK10QByteArray [QtCore]	_ZN17QByteArrayMatcherC1ERKS_ [QtCore]
_ZN17QByteArrayMatcherC1Ev [QtCore]	_ZN17QByteArrayMatcherC2ERK10QByteArray [QtCore]
ZN17QByteArrayMatcherC2ERKS [QtCore]	_ZN17QByteArrayMatcherC2Ev [QtCore]
_ZN17QByteArrayMatcherD1Ev [QtCore]	_ZN17QByteArrayMatcherD2Ev [QtCore]
ZN17QByteArrayMatcheraSERKS [QtCore]	_ZN9QBitArray4fillEbii [QtCore]
_ZN9QBitArray6resizeEi [QtCore]	_ZN9QBitArrayC1Eib [QtCore]
_ZN9QBitArrayC2Eib [QtCore]	_ZN9QBitArrayaSERKS_ [QtCore]
ZN9QBitArrayeOERKS [QtCore]	_ZN9QBitArrayoRERKS_ [QtCore]
_ZNK10QByteArray10simplifiedEv [QtCore]	_ZNK10QByteArray10startsWithEPKc [QtCore]
ZNK10QByteArray10startsWithERKS [QtCore]	_ZNK10QByteArray10startsWithEc [QtCore]
_ZNK10QByteArray10toLongLongEPbi [QtCore]	_ZNK10QByteArray11lastIndexOfERKS_i [QtCore]
_ZNK10QByteArray11lastIndexOfEci [QtCore]	_ZNK10QByteArray11toULongLongEPbi [QtCore]
_ZNK10QByteArray13leftJustifiedEicb [QtCore]	_ZNK10QByteArray14rightJustifiedEicb [QtCore]
_ZNK10QByteArray3midEii [QtCore]	_ZNK10QByteArray4leftEi [QtCore]
_ZNK10QByteArray5countEPKc [QtCore]	_ZNK10QByteArray5countERKS_ [QtCore]
_ZNK10QByteArray5countEc [QtCore]	_ZNK10QByteArray5rightEi [QtCore]
_ZNK10QByteArray5splitEc [QtCore]	_ZNK10QByteArray5toIntEPbi [QtCore]
_ZNK10QByteArray6isNullEv [QtCore]	_ZNK10QByteArray6toLongEPbi [QtCore]
_ZNK10QByteArray6toUIntEPbi [QtCore]	_ZNK10QByteArray7indexOfERKS_i [QtCore]
_ZNK10QByteArray7indexOfEci [QtCore]	_ZNK10QByteArray7toFloatEPb [QtCore]

_ZnK10QByteArray7toLowerEv [QtCore]	_ZnK10QByteArray7toShortEPbi [QtCore]
_ZnK10QByteArray7toULongEPbi [QtCore]	_ZnK10QByteArray7toUpperEv [QtCore]
_ZnK10QByteArray7trimmedEv [QtCore]	_ZnK10QByteArray8endsWithEPKc [QtCore]
_ZnK10QByteArray8endsWithERKS _ [QtCore]	_ZnK10QByteArray8endsWithEc [QtCore]
_ZnK10QByteArray8toBase64Ev [QtCore]	_ZnK10QByteArray8toDoubleEPb [QtCore]
_ZnK10QByteArray8toUShortEPbi [QtCore]	_ZnK17QByteArrayMatcher7indexI nERK10QByteArrayi [QtCore]
_ZnK9QBitArray5countEb [QtCore]	_ZnK9QBitArraycoEv [QtCore]
ZanRK9QBitArrayS1 [QtCore]	_ZeoRK9QBitArrayS1_ [QtCore]
_ZlsR11QDataStreamRK10QByteArray [QtCore]	_ZlsR11QDataStreamRK9QBitArray [QtCore]
ZorRK9QBitArrayS1 [QtCore]	_ZrsR11QDataStreamR10QByteArray [QtCore]
_ZrsR11QDataStreamR9QBitArray [QtCore]	

18.2.7 Qt4 Plugins

18.2.7.1 Class data for QTextCodecPlugin

The virtual table for the QTextCodecPlugin class is described by Table 18-38

Table 18-38 Primary vtable for QTextCodecPlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextCodecPlugin
vfunc[0]:	QTextCodecPlugin::metaObject() const
vfunc[1]:	QTextCodecPlugin::qt_metacast(char const*)
vfunc[2]:	QTextCodecPlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTextCodecPlugin::~~QTextCodecPlugin()
vfunc[4]:	QTextCodecPlugin::~~QTextCodecPlugin()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual
vfunc[17]:	QTextCodecPlugin::keys() const
vfunc[18]:	QTextCodecPlugin::create(QString const&)

The Run Time Type Information for the QTextCodecPlugin class is described by Table 18-39

Table 18-39 typeid for QTextCodecPlugin

Base Vtable	vtable for _cxxabiv1::__vmi_class _type_info	
Name	typeid name for QTextCodecPlugin	
flags:	0	
basetype:	typeid for QObject	2
basetype:	typeid for QTextCodecFactoryInte rface	2050

18.2.7.2 Class data for QPluginLoader

The virtual table for the QPluginLoader class is described by Table 18-40

Table 18-40 Primary vtable for QPluginLoader

Base Offset	0
Virtual Base Offset	0
RTTI	typeid for QPluginLoader
vfunc[0]:	QPluginLoader::metaObject() const

vfunc[1]:	QPluginLoader::qt_metacast(char const*)
vfunc[2]:	QPluginLoader::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QPluginLoader::~~QPluginLoader()
vfunc[4]:	QPluginLoader::~~QPluginLoader()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QPluginLoader class is described by Table 18-41

Table 18-41 typeinfo for QPluginLoader

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPluginLoader
basetype:	typeinfo for QObject

18.2.7.3 Class data for QLibrary

The virtual table for the QLibrary class is described by Table 18-42

Table 18-42 Primary vtable for QLibrary

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QLibrary
vfunc[0]:	QLibrary::metaObject() const
vfunc[1]:	QLibrary::qt_metacast(char const*)
vfunc[2]:	QLibrary::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QLibrary::~~QLibrary()
vfunc[4]:	QLibrary::~~QLibrary()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QLibrary class is described by Table 18-43

Table 18-43 typeinfo for QLibrary

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QLibrary
basetype:	typeinfo for QObject

18.2.7.4 Interfaces for Qt4 Plugins

An LSB conforming implementation shall provide the generic functions for Qt4 Plugins specified in Table 18-44, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-44 libQtCore - Qt4 Plugins Function Interfaces

_ZN13QPluginLoader11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN13QPluginLoader11qt_metacastEPKc [QtCore]
_ZN13QPluginLoader11setFileNameERK7QString [QtCore]	_ZN13QPluginLoader15staticInstancesEv [QtCore]
_ZN13QPluginLoader4loadEv [QtCore]	_ZN13QPluginLoader6unloadEv [QtCore]
_ZN13QPluginLoader8instanceEv [QtCore]	_ZN13QPluginLoaderC1EP7QObject [QtCore]
_ZN13QPluginLoaderC1ERK7QStringP7QObject [QtCore]	_ZN13QPluginLoaderC2EP7QObject [QtCore]
_ZN13QPluginLoaderC2ERK7QStringP7QObject [QtCore]	_ZN13QPluginLoaderD0Ev [QtCore]
_ZN13QPluginLoaderD1Ev [QtCore]	_ZN13QPluginLoaderD2Ev [QtCore]
_ZN16QTextCodecPlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN16QTextCodecPlugin11qt_metacastEPKc [QtCore]
_ZN16QTextCodecPlugin6createERK7QString [QtCore]	_ZN16QTextCodecPluginC1EP7QObject [QtCore]

_ZN16QTextCodecPluginC2EP7QOb ject [QtCore]	_ZN16QTextCodecPluginD0Ev [QtCore]
_ZN16QTextCodecPluginD1Ev [QtCore]	_ZN16QTextCodecPluginD2Ev [QtCore]
_ZN8QLibrary11qt_metacallEN11Q MetaObject4CallEiPPv [QtCore]	_ZN8QLibrary11qt_metacastEPKc [QtCore]
_ZN8QLibrary11setFileNameERK7Q String [QtCore]	_ZN8QLibrary12setLoadHintsE6QFl agsINS_8LoadHintEE [QtXml]
_ZN8QLibrary21setFileNameAndVe rsionERK7QStringi [QtCore]	_ZN8QLibrary4loadEv [QtCore]
_ZN8QLibrary6unloadEv [QtCore]	_ZN8QLibrary7resolveEPKc [QtCore]
_ZN8QLibrary7resolveERK7QString PKc [QtCore]	_ZN8QLibrary7resolveERK7QStringi PKc [QtCore]
_ZN8QLibrary9isLibraryERK7QStrin g [QtCore]	_ZN8QLibraryC1EP7QObject [QtCore]
_ZN8QLibraryC1ERK7QStringP7QO bject [QtCore]	_ZN8QLibraryC1ERK7QStringiP7Q Object [QtCore]
_ZN8QLibraryC2EP7QObject [QtCore]	_ZN8QLibraryC2ERK7QStringP7QO bject [QtCore]
_ZN8QLibraryC2ERK7QStringiP7Q Object [QtCore]	_ZN8QLibraryD0Ev [QtCore]
_ZN8QLibraryD1Ev [QtCore]	_ZN8QLibraryD2Ev [QtCore]
_ZNK13QPluginLoader10metaObject Ev [QtCore]	_ZNK13QPluginLoader11errorString Ev [QtXml]
_ZNK13QPluginLoader8fileNameEv [QtCore]	_ZNK13QPluginLoader8isLoadingEv [QtCore]
_ZNK16QTextCodecPlugin10metaO bjectEv [QtCore]	_ZNK16QTextCodecPlugin4keysEv [QtCore]
_ZNK8QLibrary10metaObjectEv [QtCore]	_ZNK8QLibrary11errorStringEv [QtXml]
_ZNK8QLibrary8fileNameEv [QtCore]	_ZNK8QLibrary8isLoadingEv [QtCore]
_ZNK8QLibrary9loadHintsEv [QtXml]	

18.2.8 Qt4 Date and Time

18.2.8.1 Class data for QTimer

The virtual table for the QTimer class is described by Table 18-45

Table 18-45 Primary vtable for QTimer

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QTimer
vfunc[0]:	QTimer::metaObject() const
vfunc[1]:	QTimer::qt_metacast(char const*)
vfunc[2]:	QTimer::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTimer::~~QTimer()
vfunc[4]:	QTimer::~~QTimer()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTimer::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QTimer class is described by Table 18-46

Table 18-46 typeinfo for QTimer

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTimer
basetype:	typeinfo for QObject

18.2.8.2 Interfaces for Qt4 Date and Time

An LSB conforming implementation shall provide the generic functions for Qt4 Date and Time specified in Table 18-47, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-47 libQtCore - Qt4 Date and Time Function Interfaces

_ZN11QBasicTimer4stopEv [QtCore]	_ZN11QBasicTimer5startEiP7QObject [QtCore]
_ZN5QDate10fromStringERK7QStringN2Qt10DateFormatE [QtCore]	_ZN5QDate10fromStringERK7QStringS2_ [QtCore]
_ZN5QDate10isLeapYearEi [QtCore]	_ZN5QDate11currentDateEv [QtCore]
_ZN5QDate11longDayNameEi [QtCore]	_ZN5QDate12shortDayNameEi [QtCore]

_ZN5QDate13longMonthNameEi [QtCore]	_ZN5QDate14shortMonthNameEi [QtCore]
_ZN5QDate17gregorianToJulianEiii [QtCore]	_ZN5QDate17julianToGregorianEjRiS0_S0_ [QtCore]
_ZN5QDate6setYMD Eiii [QtCore]	_ZN5QDate7isValidEiii [QtCore]
_ZN5QDate7setDateEiii [QtXml]	_ZN5QDateC1Eiii [QtCore]
_ZN5QDateC2Eiii [QtCore]	_ZN5QTime10fromStringERK7QStringN2Qt10DateFormatE [QtCore]
ZN5QTime10fromStringERK7QStringS2 [QtCore]	_ZN5QTime11currentTimeEv [QtCore]
_ZN5QTime5startEv [QtCore]	_ZN5QTime6setHMS Eiii [QtCore]
_ZN5QTime7isValidEiiii [QtCore]	_ZN5QTime7restartEv [QtCore]
_ZN5QTimeC1Eiiii [QtCore]	_ZN5QTimeC2Eiii [QtCore]
_ZN6QTimer10singleShotEiP7QObjectPKc [QtCore]	_ZN6QTimer10timerEventEP11QTimerEvent [QtCore]
_ZN6QTimer11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN6QTimer11qt_metacastEPKc [QtCore]
_ZN6QTimer11setIntervalEi [QtCore]	_ZN6QTimer4stopEv [QtCore]
_ZN6QTimer5startEi [QtCore]	_ZN6QTimer5startEib [QtCore]
_ZN6QTimer5startEv [QtCore]	_ZN6QTimer7timeoutEv [QtCore]
_ZN6QTimerC1EP7QObject [QtCore]	_ZN6QTimerC1EP7QObjectPKc [QtCore]
_ZN6QTimerC2EP7QObject [QtCore]	_ZN6QTimerC2EP7QObjectPKc [QtCore]
_ZN6QTimerD0Ev [QtCore]	_ZN6QTimerD1Ev [QtCore]
_ZN6QTimerD2Ev [QtCore]	_ZN9QDateTime10fromStringERK7QStringN2Qt10DateFormatE [QtCore]
ZN9QDateTime10fromStringERK7QStringS2 [QtCore]	_ZN9QDateTime10fromTime_tEj [QtXml]
_ZN9QDateTime11setTimeSpecEN2Qt8TimeSpecE [QtCore]	_ZN9QDateTime15currentDateTimeEv [QtCore]
_ZN9QDateTime7setDateERK5QDate [QtCore]	_ZN9QDateTime7setTimeERK5QTime [QtCore]
_ZN9QDateTime9setTime_tEj [QtCore]	_ZN9QDateTimeC1ERK5QDate [QtCore]
_ZN9QDateTimeC1ERK5QDateRK5QTimeN2Qt8TimeSpecE [QtCore]	_ZN9QDateTimeC1ERKS_ [QtCore]
_ZN9QDateTimeC1Ev [QtCore]	_ZN9QDateTimeC2ERK5QDate [QtCore]

_ZN9QDateTimeC2ERK5QDateRK5QTimeN2Qt8TimeSpecE [QtCore]	_ZN9QDateTimeC2ERKS_ [QtCore]
_ZN9QDateTimeC2Ev [QtCore]	_ZN9QDateTimeD1Ev [QtCore]
_ZN9QDateTimeD2Ev [QtCore]	_ZN9QDateTimeeaSERKS_ [QtCore]
_ZN9QTimeLine10timerEventEP11QTimerEvent [QtXml]	_ZN9QTimeLine11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN9QTimeLine11qt_metacastEPKc [QtXml]	_ZN9QTimeLine11setDurationEi [QtXml]
_ZN9QTimeLine11setEndFrameEi [QtXml]	_ZN9QTimeLine12frameChangedEi [QtXml]
_ZN9QTimeLine12setDirectionENS_9DirectionE [QtXml]	_ZN9QTimeLine12setLoopCountEi [QtXml]
_ZN9QTimeLine12stateChangedENS_5StateE [QtXml]	_ZN9QTimeLine12valueChangedEd [QtXml]
_ZN9QTimeLine13setCurveShapeENS_10CurveShapeE [QtXml]	_ZN9QTimeLine13setFrameRangeEii [QtXml]
_ZN9QTimeLine13setStartFrameEi [QtXml]	_ZN9QTimeLine14setCurrentTimeEi [QtXml]
_ZN9QTimeLine15toggleDirectionEv [QtXml]	_ZN9QTimeLine17setUpdateIntervalEi [QtXml]
_ZN9QTimeLine4stopEv [QtXml]	_ZN9QTimeLine5startEv [QtXml]
_ZN9QTimeLine8finishedEv [QtXml]	_ZN9QTimeLine9setPausedEb [QtXml]
_ZN9QTimeLineC1EiP7QObject [QtXml]	_ZN9QTimeLineC2EiP7QObject [QtXml]
_ZN9QTimeLineD0Ev [QtXml]	_ZN9QTimeLineD1Ev [QtXml]
_ZN9QTimeLineD2Ev [QtXml]	_ZNK5QDate10daysInYearEv [QtCore]
_ZNK5QDate10weekNumberEPi [QtCore]	_ZNK5QDate11daysInMonthEv [QtCore]
_ZNK5QDate3dayEv [QtCore]	_ZNK5QDate4yearEv [QtCore]
_ZNK5QDate5monthEv [QtCore]	_ZNK5QDate6daysToERKS_ [QtCore]
_ZNK5QDate7addDaysEi [QtCore]	_ZNK5QDate7isValidEv [QtCore]
_ZNK5QDate8addYearsEi [QtCore]	_ZNK5QDate8toStringEN2Qt10DateFormatE [QtCore]
_ZNK5QDate8toStringERK7QString [QtCore]	_ZNK5QDate9addMonthsEi [QtCore]
_ZNK5QDate9dayOfWeekEv [QtCore]	_ZNK5QDate9dayOfYearEv [QtCore]
_ZNK5QTime4hourEv [QtCore]	_ZNK5QTime4msecEv [QtCore]

_Znk5QTime6minuteEv [QtCore]	_Znk5QTime6secondEv [QtCore]
Znk5QTime6secsToERKS [QtCore]	_Znk5QTime7addSecsEi [QtCore]
_Znk5QTime7elapsedEv [QtCore]	_Znk5QTime7isValidEv [QtCore]
Znk5QTime7msecsToERKS [QtCore]	_Znk5QTime8addMsecsEi [QtCore]
_Znk5QTime8toStringEN2Qt10Date FormatE [QtCore]	_Znk5QTime8toStringERK7QString [QtCore]
_Znk6QTimer10metaObjectEv [QtCore]	_Znk9QDateTime10toTimeSpecEN2 Qt8TimeSpecE [QtCore]
_Znk9QDateTime4dateEv [QtCore]	_Znk9QDateTime4timeEv [QtCore]
Znk9QDateTime6daysToERKS [QtCore]	_Znk9QDateTime6isNullEv [QtCore]
Znk9QDateTime6secsToERKS [QtCore]	_Znk9QDateTime7addDaysEi [QtCore]
_Znk9QDateTime7addSecsEi [QtCore]	_Znk9QDateTime7isValidEv [QtCore]
_Znk9QDateTime8addMsecsEx [QtCore]	_Znk9QDateTime8addYearsEi [QtCore]
_Znk9QDateTime8timeSpecEv [QtCore]	_Znk9QDateTime8toStringEN2Qt10 DateFormatE [QtCore]
_Znk9QDateTime8toStringERK7QSt ring [QtCore]	_Znk9QDateTime8toTime_tEv [QtCore]
_Znk9QDateTime9addMonthsEi [QtCore]	_Znk9QDateTimeeqERKS_ [QtCore]
Znk9QDateTime10ERKS [QtCore]	_Znk9QTimeLine10curveShapeEv [QtXml]
_Znk9QTimeLine10metaObjectEv [QtXml]	_Znk9QTimeLine10startFrameEv [QtXml]
_Znk9QTimeLine11currentTimeEv [QtXml]	_Znk9QTimeLine12currentFrameEv [QtXml]
_Znk9QTimeLine12currentValueEv [QtXml]	_Znk9QTimeLine12frameForTimeEi [QtXml]
_Znk9QTimeLine12valueForTimeEi [QtXml]	_Znk9QTimeLine14updateIntervalE v [QtXml]
_Znk9QTimeLine5stateEv [QtXml]	_Znk9QTimeLine8durationEv [QtXml]
_Znk9QTimeLine8endFrameEv [QtXml]	_Znk9QTimeLine9directionEv [QtXml]
_Znk9QTimeLine9loopCountEv [QtXml]	_Zls6QDebugRK5QDate [QtCore]

_Zls6QDebugRK5QTime [QtCore]	_Zls6QDebugRK9QDateTime [QtCore]
_ZlsR11QDataStreamRK5QDate [QtCore]	_ZlsR11QDataStreamRK5QTime [QtCore]
_ZlsR11QDataStreamRK9QDateTime [QtCore]	_ZrsR11QDataStreamR5QDate [QtCore]
_ZrsR11QDataStreamR5QTime [QtCore]	_ZrsR11QDataStreamR9QDateTime [QtCore]

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Date and Time specified in Table 18-48, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-48 libQtCore - Qt4 Date and Time Deprecated Interfaces

_ZN5QDate17gregorianToJulianEiii [QtCore]	_ZN5QDate17julianToGregorianEjRi S0_S0_ [QtCore]
_ZN5QDate6setYMDEiii [QtCore]	

18.2.9 Qt4 Miscellaneous

18.2.9.1 Class data for QFactoryInterface

The virtual table for the QFactoryInterface class is described by Table 18-49

Table 18-49 Primary vtable for QFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFactoryInterface
vfunc[0]:	NULL or QFactoryInterface::~~QFactoryInterface()
vfunc[1]:	NULL or QFactoryInterface::~~QFactoryInterface()
vfunc[2]:	__cxa_pure_virtual

The Run Time Type Information for the QFactoryInterface class is described by Table 18-50

Table 18-50 typeinfo for QFactoryInterface

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QFactoryInterface

18.2.9.2 Interfaces for Qt4 Miscellaneous

An LSB conforming implementation shall provide the generic functions for Qt4 Miscellaneous specified in Table 18-51, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-51 libQtCore - Qt4 Miscellaneous Function Interfaces

_ZN12QLibraryInfo16licensedProductsEv [QtCore]	_ZN12QLibraryInfo8buildKeyEv [QtCore]
_ZN12QLibraryInfo8licenseeEv [QtCore]	_ZN12QLibraryInfo8locationENS_15LibraryLocationE [QtCore]
_ZN5QUuid10createUuidEv [QtCore]	_ZN5QUuidC1EPKc [QtCore]
_ZN5QUuidC1ERK7QString [QtCore]	_ZN5QUuidC2EPKc [QtCore]
_ZN5QUuidC2ERK7QString [QtCore]	_ZN7QRegExp10setMinimalEb [QtCore]
_ZN7QRegExp10setPatternERK7QString [QtCore]	_ZN7QRegExp11errorStringEv [QtCore]
_ZN7QRegExp13capturedTextsEv [QtCore]	_ZN7QRegExp16setPatternSyntaxENS_13PatternSyntaxE [QtCore]
_ZN7QRegExp18setCaseSensitivityEN2Qt15CaseSensitivityE [QtCore]	_ZN7QRegExp3capEi [QtCore]
_ZN7QRegExp3posEi [QtCore]	_ZN7QRegExp6escapeERK7QString [QtCore]
_ZN7QRegExpC1ERK7QStringN2Qt15CaseSensitivityENS_13PatternSyntaxE [QtCore]	_ZN7QRegExpC1ERKS_ [QtCore]
_ZN7QRegExpC1Ev [QtCore]	_ZN7QRegExpC2ERK7QStringN2Qt15CaseSensitivityENS_13PatternSyntaxE [QtCore]
ZN7QRegExpC2ERKS [QtCore]	_ZN7QRegExpC2Ev [QtCore]
_ZN7QRegExpD1Ev [QtCore]	_ZN7QRegExpD2Ev [QtCore]
ZN7QRegExppaSERKS [QtCore]	_ZNK5QUuid6isNullEv [QtCore]
_ZNK5QUuid7variantEv [QtCore]	_ZNK5QUuid7versionEv [QtCore]
_ZNK5QUuid8toStringEv [QtCore]	_ZNK5QUuidgtERKS_ [LSB]
ZNK5QUuidltERKS [LSB]	_ZNK7QRegExp10exactMatchERK7QString [QtCore]
_ZNK7QRegExp11lastIndexInERK7QStringiNS_9CaretModeE [QtCore]	_ZNK7QRegExp11numCapturesEv [QtCore]
_ZNK7QRegExp13matchedLengthEv [QtCore]	_ZNK7QRegExp13patternSyntaxEv [QtCore]

_ZNK7QRegExp15caseSensitivityEv [QtCore]	_ZNK7QRegExp7indexInERK7QStri ngiNS_9CaretModeE [QtCore]
_ZNK7QRegExp7isEmptyEv [QtCore]	_ZNK7QRegExp7isValidEv [QtCore]
_ZNK7QRegExp7patternEv [QtCore]	_ZNK7QRegExp9isMinimalEv [QtCore]
ZNK7QRegExpEqERKS [QtCore]	_ZlsR11QDataStreamRK5QUuid [QtCore]
_ZlsR11QDataStreamRK7QRegExp [QtCore]	_ZrsR11QDataStreamR5QUuid [QtCore]
_ZrsR11QDataStreamR7QRegExp [QtCore]	

18.2.10 Qt4 Text

18.2.10.1 Interfaces for Qt4 Text

An LSB conforming implementation shall provide the generic functions for Qt4 Text specified in Table 18-52, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-52 libQtCore - Qt4 Text Function Interfaces

_ZN14QStringMatcher10setPatternE RK7QString [QtCore]	_ZN14QStringMatcher18setCaseSens itivityEN2Qt15CaseSensitivityE [QtCore]
_ZN14QStringMatcherC1ERK7QStri ngN2Qt15CaseSensitivityE [QtCore]	_ZN14QStringMatcherC1ERKS_ [QtCore]
_ZN14QStringMatcherC1Ev [QtCore]	_ZN14QStringMatcherC2ERK7QStri ngN2Qt15CaseSensitivityE [QtCore]
ZN14QStringMatcherC2ERKS [QtCore]	_ZN14QStringMatcherC2Ev [QtCore]
_ZN14QStringMatcherD1Ev [QtCore]	_ZN14QStringMatcherD2Ev [QtCore]
ZN14QStringMatcheraSERKS [QtCore]	_ZN5QChar9fromAsciiEc [QtCore]
_ZN5QCharC1Ec [QtCore]	_ZN5QCharC1Eh [QtCore]
_ZN5QCharC2Ec [QtCore]	_ZN5QCharC2Eh [QtCore]
_ZN7QString10fromLatin1EPKci [QtCore]	_ZN7QString10setUnicodeEPK5QCh ari [QtCore]
_ZN7QString11fromRawDataEPK5Q Chari [QtCore]	_ZN7QString13fromLocal8BitEPKci [QtCore]
_ZN7QString14fromWCharArrayEP Kwi [QtXml]	_ZN7QString16fromAscii_helperEPK ci [QtXml]

_ZN7QString17fromLatin1_helperEPKci [QtCore]	_ZN7QString4chopEi [QtCore]
_ZN7QString4fillE5QChar [QtCore]	_ZN7QString4freeEPNS_4DataE [LSB]
_ZN7QString6appendE5QChar [QtCore]	_ZN7QString6appendERK13QLatin1String [QtCore]
ZN7QString6appendERKS [QtCore]	_ZN7QString6expandEi [QtCore]
_ZN7QString6insertEi5QChar [QtCore]	_ZN7QString6insertEiPK5QChar [QtCore]
_ZN7QString6insertEiRK13QLatin1String [QtCore]	_ZN7QString6numberEdci [QtCore]
_ZN7QString6numberEii [QtCore]	_ZN7QString6numberEji [QtCore]
_ZN7QString6numberEli [QtCore]	_ZN7QString6numberEmi [QtCore]
_ZN7QString6numberExi [QtCore]	_ZN7QString6numberEyi [QtCore]
_ZN7QString6removeE5QCharN2Qt15CaseSensitivityE [QtCore]	_ZN7QString6removeERKS_N2Qt15CaseSensitivityE [QtCore]
_ZN7QString6removeEii [QtCore]	_ZN7QString6resizeEi [QtCore]
_ZN7QString6setNumEdci [QtCore]	_ZN7QString6setNumExi [QtCore]
_ZN7QString6setNumEyi [QtCore]	_ZN7QString7reallocEi [QtCore]
_ZN7QString7reallocEv [QtCore]	_ZN7QString7replaceE5QCharRKS_N2Qt15CaseSensitivityE [QtCore]
_ZN7QString7replaceE5QCharS0_N2Qt15CaseSensitivityE [QtCore]	_ZN7QString7replaceERK7QRegExpRKS_ [QtCore]
_ZN7QString7replaceERKS_S1_N2Qt15CaseSensitivityE [QtCore]	_ZN7QString7replaceEii5QChar [QtCore]
_ZN7QString7replaceEiiPK5QChar [QtCore]	_ZN7QString7replaceEiiRKS_ [QtCore]
_ZN7QString7sprintfEPKcz [QtCore]	_ZN7QString8fromUcs4EPKji [QtXml]
_ZN7QString8fromUtf8EPKci [QtCore]	_ZN7QString8truncateEi [QtCore]
_ZN7QString9fromAsciiEPKci [QtCore]	_ZN7QString9fromUtf16EPKti [QtCore]
_ZN7QStringC1E5QChar [QtCore]	_ZN7QStringC1EPK5QChar [QtCore]
_ZN7QStringC1Ei5QChar [QtCore]	_ZN7QStringC2E5QChar [QtCore]
_ZN7QStringC2EPK5QChar [QtCore]	_ZN7QStringC2Ei5QChar [QtCore]
_ZN7StringaSE5QChar [QtCore]	_ZN7StringaSERKS_ [QtCore]

_Znk14QStringMatcher7indexInERK7QStringi [QtCore]	_Znk5QChar10digitValueEv [QtCore]
_Znk5QChar11hasMirroredEv [QtCore]	_Znk5QChar12mirroredCharEv [QtCore]
_Znk5QChar13decompositionEv [QtCore]	_Znk5QChar14combiningClassEv [QtCore]
_Znk5QChar14unicodeVersionEv [QtCore]	_Znk5QChar16decompositionTagEv [QtCore]
_Znk5QChar16isLetterOrNumberEv [QtCore]	_Znk5QChar6isMarkEv [QtCore]
_Znk5QChar7isDigitEv [QtCore]	_Znk5QChar7isPrintEv [QtCore]
_Znk5QChar7isPunctEv [QtCore]	_Znk5QChar7isSpaceEv [QtCore]
_Znk5QChar7joiningEv [QtCore]	_Znk5QChar7toAsciiEv [QtCore]
_Znk5QChar7toLowerEv [QtCore]	_Znk5QChar7toUpperEv [QtCore]
_Znk5QChar8categoryEv [QtCore]	_Znk5QChar8isLetterEv [QtCore]
_Znk5QChar8isNumberEv [QtCore]	_Znk5QChar8isSymbolEv [QtCore]
_Znk5QChar9directionEv [QtCore]	_Znk7QString10normalizedENS_17 NormalizationFormE [QtCore]
_Znk7QString10normalizedENS_17 NormalizationFormEN5QChar14UnicodeVersionE [QtCore]	_Znk7QString10simplifiedEv [QtCore]
_Znk7QString10startsWithERK13QLatin1StringN2Qt15CaseSensitivityE [QtCore]	_Znk7QString10startsWithERK5QCharN2Qt15CaseSensitivityE [QtCore]
_Znk7QString10startsWithERKS_N2Qt15CaseSensitivityE [QtCore]	_Znk7QString10toLongLongEPbi [QtCore]
_Znk7QString11lastIndexOfE5QCharN2Qt15CaseSensitivityE [QtCore]	_Znk7QString11lastIndexOfERK7QRegExpi [QtCore]
_Znk7QString11lastIndexOfERKS_iN2Qt15CaseSensitivityE [QtCore]	_Znk7QString11toLocal8BitEv [QtCore]
_Znk7QString11toULongLongEPbi [QtCore]	_Znk7QString12ascii_helperEv [QtCore]
_Znk7QString12toWCharArrayEPw [QtXml]	_Znk7QString13latin1_helperEv [QtCore]
_Znk7QString13leftJustifiedEi5QCharb [QtCore]	_Znk7QString14rightJustifiedEi5QCharb [QtCore]
Znk7QString18localeAwareCompareERKS [QtCore]	_Znk7QString3argE5QCharIRKS0_ [QtCore]
_Znk7QString3argERKS_iRK5QChar [QtCore]	_Znk7QString3argEciRK5QChar [QtCore]

_Znk7QString3argEdiciRK5QChar [QtCore]	_Znk7QString3argExiiRK5QChar [QtCore]
_Znk7QString3argEyiiRK5QChar [QtCore]	_Znk7QString3midEii [QtCore]
_Znk7QString4leftEi [QtCore]	_Znk7QString5countE5QCharN2Qt15CaseSensitivityE [QtCore]
_Znk7QString5countERK7QRegExp [QtCore]	_Znk7QString5countERKS_N2Qt15CaseSensitivityE [QtCore]
_Znk7QString5rightEi [QtCore]	_Znk7QString5splitERK5QCharNS_13SplitBehaviorEN2Qt15CaseSensitivityE [QtCore]
_Znk7QString5splitERK7QRegExpNS_13SplitBehaviorE [QtCore]	_Znk7QString5splitERKS_NS_13SplitBehaviorEN2Qt15CaseSensitivityE [QtCore]
_Znk7QString5toIntEPbi [QtCore]	_Znk7QString5utf16Ev [QtCore]
_Znk7QString6toLongEPbi [QtCore]	_Znk7QString6toULongEPbi [QtCore]
_Znk7QString6toUcs4Ev [QtXml]	_Znk7QString6toUtf8Ev [QtCore]
_Znk7QString7compareERK13QLatin1StringN2Qt15CaseSensitivityE [QtXml]	_Znk7QString7compareERKS_ [QtCore]
_Znk7QString7compareERKS_N2Qt15CaseSensitivityE [QtXml]	_Znk7QString7indexOfE5QCharIN2Qt15CaseSensitivityE [QtCore]
_Znk7QString7indexOfERK7QRegExp [QtCore]	_Znk7QString7indexOfERKS_iN2Qt15CaseSensitivityE [QtCore]
_Znk7QString7sectionERK7QRegExp [QtCore]	_Znk7QString7sectionERKS_ii6QFlagsINS_11SectionFlagEE [QtCore]
_Znk7QString7toAsciiEv [QtCore]	_Znk7QString7toFloatEPb [QtCore]
_Znk7QString7toLowerEv [QtCore]	_Znk7QString7toShortEPbi [QtCore]
_Znk7QString7toULongEPbi [QtCore]	_Znk7QString7toUpperEv [QtCore]
_Znk7QString7trimmedEv [QtCore]	_Znk7QString8endsWithERK13QLatin1StringN2Qt15CaseSensitivityE [QtCore]
_Znk7QString8endsWithERK5QCharN2Qt15CaseSensitivityE [QtCore]	_Znk7QString8endsWithERKS_N2Qt15CaseSensitivityE [QtCore]
Znk7QString8multiArgEiPPKS [QtCore]	_Znk7QString8toDoubleEPb [QtCore]
_Znk7QString8toLatin1Ev [QtCore]	_Znk7QString8toUShortEPbi [QtCore]
_Znk7QStringeqERK13QLatin1String [QtCore]	_Znk7QStringeqERKS_ [QtCore]

_ZNK7QStringgtERK13QLatin1String [QtCore]	_ZNK7QStringltERK13QLatin1String [QtCore]
ZNK7QStringltERKS [QtCore]	_ZlsR11QDataStreamRK5QChar [QtCore]
_ZlsR11QDataStreamRK7QString [QtCore]	_ZrsR11QDataStreamR5QChar [QtCore]
_ZrsR11QDataStreamR7QString [QtCore]	

18.2.11 Qt4 Input/Output

18.2.11.1 Class data for QIODevice

The virtual table for the QIODevice class is described by Table 18-53

Table 18-53 Primary vtable for QIODevice

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QIODevice
vfunc[0]:	QIODevice::metaObject() const
vfunc[1]:	QIODevice::qt_metacast(char const*)
vfunc[2]:	QIODevice::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QIODevice::~~QIODevice()
vfunc[4]:	QIODevice::~~QIODevice()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QIODevice::isSequential() const
vfunc[13]:	QIODevice::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QIODevice::close()
vfunc[15]:	QIODevice::pos() const
vfunc[16]:	QIODevice::size() const

vfunc[17]:	QIODevice::seek(long long)
vfunc[18]:	QIODevice::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QIODevice::bytesAvailable() const
vfunc[21]:	QIODevice::bytesToWrite() const
vfunc[22]:	QIODevice::canReadLine() const
vfunc[23]:	QIODevice::waitForReadyRead(int)
vfunc[24]:	QIODevice::waitForBytesWritten(int)
vfunc[25]:	__cxa_pure_virtual
vfunc[26]:	QIODevice::readLineData(char* long long)
vfunc[27]:	__cxa_pure_virtual

The Run Time Type Information for the QIODevice class is described by Table 18-54

Table 18-54 typeinfo for QIODevice

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QIODevice
basetype:	typeinfo for QObject

18.2.11.2 Class data for QDataStream

The virtual table for the QDataStream class is described by Table 18-55

Table 18-55 Primary vtable for QDataStream

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDataStream
vfunc[0]:	QDataStream::~~QDataStream()
vfunc[1]:	QDataStream::~~QDataStream()

The Run Time Type Information for the QDataStream class is described by Table 18-56

Table 18-56 typeinfo for QDataStream

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QDataStream

18.2.11.3 Class data for QTextStream

The virtual table for the QTextStream class is described by Table 18-57

Table 18-57 Primary vtable for QTextStream

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextStream
vfunc[0]:	QTextStream::~~QTextStream()
vfunc[1]:	QTextStream::~~QTextStream()

The Run Time Type Information for the QTextStream class is described by Table 18-58

Table 18-58 typeinfo for QTextStream

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTextStream

18.2.11.4 Class data for QFile

The virtual table for the QFile class is described by Table 18-59

Table 18-59 Primary vtable for QFile

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFile
vfunc[0]:	QFile::metaObject() const
vfunc[1]:	QFile::qt_metacast(char const*)
vfunc[2]:	QFile::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFile::~~QFile()
vfunc[4]:	QFile::~~QFile()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QFile::isSequential() const
vfunc[13]:	QFile::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QFile::close()
vfunc[15]:	QFile::pos() const
vfunc[16]:	QFile::size() const
vfunc[17]:	QFile::seek(long long)
vfunc[18]:	QFile::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QIODevice::bytesAvailable() const
vfunc[21]:	QIODevice::bytesToWrite() const
vfunc[22]:	QIODevice::canReadLine() const
vfunc[23]:	QIODevice::waitForReadyRead(int)
vfunc[24]:	QIODevice::waitForBytesWritten(int)
vfunc[25]:	QFile::readData(char*, long long)
vfunc[26]:	QFile::readLineData(char*, long long)
vfunc[27]:	QFile::writeData(char const*, long long)
vfunc[28]:	QFile::fileEngine() const

The Run Time Type Information for the QFile class is described by Table 18-60

Table 18-60 typeinfo for QFile

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFile
basetype:	typeinfo for QIODevice

18.2.11.5 Class data for QTemporaryFile

The virtual table for the QTemporaryFile class is described by Table 18-61

Table 18-61 Primary vtable for QTemporaryFile

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTemporaryFile
vfunc[0]:	QTemporaryFile::metaObject() const
vfunc[1]:	QTemporaryFile::qt_metacast(char const*)

vfunc[2]:	QTemporaryFile::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTemporaryFile::~~QTemporaryFile()
vfunc[4]:	QTemporaryFile::~~QTemporaryFile()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QFile::isSequential() const
vfunc[13]:	QTemporaryFile::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QFile::close()
vfunc[15]:	QFile::pos() const
vfunc[16]:	QFile::size() const
vfunc[17]:	QFile::seek(long long)
vfunc[18]:	QFile::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QIODevice::bytesAvailable() const
vfunc[21]:	QIODevice::bytesToWrite() const
vfunc[22]:	QIODevice::canReadLine() const
vfunc[23]:	QIODevice::waitForReadyRead(int)
vfunc[24]:	QIODevice::waitForBytesWritten(int)
vfunc[25]:	QFile::readData(char*, long long)
vfunc[26]:	QFile::readLineData(char*, long long)
vfunc[27]:	QFile::writeData(char const*, long long)
vfunc[28]:	QTemporaryFile::fileEngine() const

The Run Time Type Information for the QTemporaryFile class is described by Table 18-62

Table 18-62 typeinfo for QTemporaryFile

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTemporaryFile
basetype:	typeinfo for QFile

18.2.11.6 Class data for QAbstractFileEngine

The virtual table for the QAbstractFileEngine class is described by Table 18-63

Table 18-63 Primary vtable for QAbstractFileEngine

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractFileEngine
vfunc[0]:	QAbstractFileEngine::~QAbstractFileEngine()
vfunc[1]:	QAbstractFileEngine::~~QAbstractFileEngine()
vfunc[2]:	QAbstractFileEngine::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[3]:	QAbstractFileEngine::close()
vfunc[4]:	QAbstractFileEngine::flush()
vfunc[5]:	QAbstractFileEngine::size() const
vfunc[6]:	QAbstractFileEngine::pos() const
vfunc[7]:	QAbstractFileEngine::seek(long long)
vfunc[8]:	QAbstractFileEngine::isSequential() const
vfunc[9]:	QAbstractFileEngine::remove()
vfunc[10]:	QAbstractFileEngine::copy(QString const&)
vfunc[11]:	QAbstractFileEngine::rename(QString const&)
vfunc[12]:	QAbstractFileEngine::link(QString const&)
vfunc[13]:	QAbstractFileEngine::mkdir(QString const&, bool) const
vfunc[14]:	QAbstractFileEngine::rmdir(QString const&, bool) const
vfunc[15]:	QAbstractFileEngine::setSize(long long)

vfunc[16]:	QAbstractFileEngine::caseSensitive() const
vfunc[17]:	QAbstractFileEngine::isRelativePath() const
vfunc[18]:	QAbstractFileEngine::entryList(QFlags<QDir::Filter>, QStringList const&) const
vfunc[19]:	QAbstractFileEngine::fileFlags(QFlags<QAbstractFileEngine::FileFlag>) const
vfunc[20]:	QAbstractFileEngine::setPermissions(unsigned int)
vfunc[21]:	QAbstractFileEngine::fileName(QAbstractFileEngine::FileName) const
vfunc[22]:	QAbstractFileEngine::ownerId(QAbstractFileEngine::FileOwner) const
vfunc[23]:	QAbstractFileEngine::owner(QAbstractFileEngine::FileOwner) const
vfunc[24]:	QAbstractFileEngine::fileTime(QAbstractFileEngine::FileTime) const
vfunc[25]:	QAbstractFileEngine::setFileName(QString const&)
vfunc[26]:	QAbstractFileEngine::handle() const
vfunc[27]:	QAbstractFileEngine::beginEntryList(QFlags<QDir::Filter>, QStringList const&)
vfunc[28]:	QAbstractFileEngine::endEntryList()
vfunc[29]:	QAbstractFileEngine::read(char*, long long)
vfunc[30]:	QAbstractFileEngine::readLine(char*, long long)
vfunc[31]:	QAbstractFileEngine::write(char const*, long long)
vfunc[32]:	QAbstractFileEngine::extension(QAbstractFileEngine::Extension, QAbstractFileEngine::ExtensionOption const*, QAbstractFileEngine::ExtensionReturn*)
vfunc[33]:	QAbstractFileEngine::supportsExtension(QAbstractFileEngine::Extension) const

The Run Time Type Information for the QAbstractFileEngine class is described by Table 18-64

Table 18-64 typeinfo for QAbstractFileEngine

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QAbstractFileEngine

18.2.11.7 Class data for QAbstractFileEngineHandler

The virtual table for the QAbstractFileEngineHandler class is described by Table 18-65

Table 18-65 Primary vtable for QAbstractFileEngineHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractFileEngineHandler
vfunc[0]:	QAbstractFileEngineHandler::~~QAbstractFileEngineHandler()
vfunc[1]:	QAbstractFileEngineHandler::~~QAbstractFileEngineHandler()
vfunc[2]:	__cxa_pure_virtual

The Run Time Type Information for the QAbstractFileEngineHandler class is described by Table 18-66

Table 18-66 typeinfo for QAbstractFileEngineHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QAbstractFileEngineHandler

18.2.11.8 Class data for QFSFileEngine

The virtual table for the QFSFileEngine class is described by Table 18-67

Table 18-67 Primary vtable for QFSFileEngine

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFSFileEngine
vfunc[0]:	QFSFileEngine::~~QFSFileEngine()
vfunc[1]:	QFSFileEngine::~~QFSFileEngine()

vfunc[2]:	QFSFileEngine::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[3]:	QFSFileEngine::close()
vfunc[4]:	QFSFileEngine::flush()
vfunc[5]:	QFSFileEngine::size() const
vfunc[6]:	QFSFileEngine::pos() const
vfunc[7]:	QFSFileEngine::seek(long long)
vfunc[8]:	QFSFileEngine::isSequential() const
vfunc[9]:	QFSFileEngine::remove()
vfunc[10]:	QFSFileEngine::copy(QString const&)
vfunc[11]:	QFSFileEngine::rename(QString const&)
vfunc[12]:	QFSFileEngine::link(QString const&)
vfunc[13]:	QFSFileEngine::mkdir(QString const&, bool) const
vfunc[14]:	QFSFileEngine::rmdir(QString const&, bool) const
vfunc[15]:	QFSFileEngine::setSize(long long)
vfunc[16]:	QFSFileEngine::caseSensitive() const
vfunc[17]:	QFSFileEngine::isRelativePath() const
vfunc[18]:	QFSFileEngine::entryList(QFlags<QDir::Filter>, QStringList const&) const
vfunc[19]:	QFSFileEngine::fileFlags(QFlags<QAbstractFileEngine::FileFlag>) const
vfunc[20]:	QFSFileEngine::setPermissions(unsigned int)
vfunc[21]:	QFSFileEngine::fileName(QAbstractFileEngine::FileName) const
vfunc[22]:	QFSFileEngine::ownerId(QAbstractFileEngine::FileOwner) const
vfunc[23]:	QFSFileEngine::owner(QAbstractFileEngine::FileOwner) const
vfunc[24]:	QFSFileEngine::fileTime(QAbstractFileEngine::FileTime) const
vfunc[25]:	QFSFileEngine::setFileName(QString const&)
vfunc[26]:	QFSFileEngine::handle() const

vfunc[27]:	QFSFileEngine::beginEntryList(QFlags<QDir::Filter>, QStringList const&)
vfunc[28]:	QFSFileEngine::endEntryList()
vfunc[29]:	QFSFileEngine::read(char*, long long)
vfunc[30]:	QFSFileEngine::readLine(char*, long long)
vfunc[31]:	QFSFileEngine::write(char const*, long long)
vfunc[32]:	QFSFileEngine::extension(QAbstractFileEngine::Extension, QAbstractFileEngine::ExtensionOption const*, QAbstractFileEngine::ExtensionReturn*)
vfunc[33]:	QFSFileEngine::supportsExtension(QAbstractFileEngine::Extension) const

The Run Time Type Information for the QFSFileEngine class is described by Table 18-68

Table 18-68 typeinfo for QFSFileEngine

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFSFileEngine
basetype:	typeinfo for QAbstractFileEngine

18.2.11.9 Class data for QProcess

The virtual table for the QProcess class is described by Table 18-69

Table 18-69 Primary vtable for QProcess

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QProcess
vfunc[0]:	QProcess::metaObject() const
vfunc[1]:	QProcess::qt_metacast(char const*)
vfunc[2]:	QProcess::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QProcess::~QProcess()
vfunc[4]:	QProcess::~QProcess()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QProcess::isSequential() const
vfunc[13]:	QIODevice::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QProcess::close()
vfunc[15]:	QIODevice::pos() const
vfunc[16]:	QIODevice::size() const
vfunc[17]:	QIODevice::seek(long long)
vfunc[18]:	QProcess::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QProcess::bytesAvailable() const
vfunc[21]:	QProcess::bytesToWrite() const
vfunc[22]:	QProcess::canReadLine() const
vfunc[23]:	QProcess::waitForReadyRead(int)
vfunc[24]:	QProcess::waitForBytesWritten(int)
vfunc[25]:	QProcess::readData(char*, long long)
vfunc[26]:	QIODevice::readLineData(char*, long long)
vfunc[27]:	QProcess::writeData(char const*, long long)
vfunc[28]:	QProcess::setupChildProcess()

The Run Time Type Information for the QProcess class is described by Table 18-70

Table 18-70 typeinfo for QProcess

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QProcess
basetype:	typeinfo for QIODevice

18.2.11.10 Class data for QBuffer

The virtual table for the QBuffer class is described by Table 18-71

Table 18-71 Primary vtable for QBuffer

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QBuffer
vfunc[0]:	QBuffer::metaObject() const
vfunc[1]:	QBuffer::qt_metacast(char const*)
vfunc[2]:	QBuffer::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QBuffer::~QBuffer()
vfunc[4]:	QBuffer::~QBuffer()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QIODevice::isSequential() const
vfunc[13]:	QBuffer::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QBuffer::close()
vfunc[15]:	QBuffer::pos() const
vfunc[16]:	QBuffer::size() const
vfunc[17]:	QBuffer::seek(long long)
vfunc[18]:	QBuffer::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QIODevice::bytesAvailable() const
vfunc[21]:	QIODevice::bytesToWrite() const
vfunc[22]:	QBuffer::canReadLine() const
vfunc[23]:	QIODevice::waitForReadyRead(int)
vfunc[24]:	QIODevice::waitForBytesWritten(int)
vfunc[25]:	QBuffer::readData(char*, long long)

vfunc[26]:	QIODevice::readLineData(char*, long long)
vfunc[27]:	QBuffer::writeData(char const*, long long)

The Run Time Type Information for the QBuffer class is described by Table 18-72

Table 18-72 typeinfo for QBuffer

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QBuffer
basetype:	typeinfo for QIODevice

18.2.11.11 Class data for QSettings

The virtual table for the QSettings class is described by Table 18-73

Table 18-73 Primary vtable for QSettings

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSettings
vfunc[0]:	QSettings::metaObject() const
vfunc[1]:	QSettings::qt_metacast(char const*)
vfunc[2]:	QSettings::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSettings::~~QSettings()
vfunc[4]:	QSettings::~QSettings()
vfunc[5]:	QSettings::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QSettings class is described by Table 18-74

Table 18-74 typeinfo for QSettings

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSettings
basetype:	typeinfo for QObject

18.2.11.12 Class data for QSignalMapper

The virtual table for the QSignalMapper class is described by Table 18-75

Table 18-75 Primary vtable for QSignalMapper

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSignalMapper
vfunc[0]:	QSignalMapper::metaObject() const
vfunc[1]:	QSignalMapper::qt_metacast(char const*)
vfunc[2]:	QSignalMapper::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSignalMapper::~~QSignalMapper()
vfunc[4]:	QSignalMapper::~~QSignalMapper()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QSignalMapper class is described by Table 18-76

Table 18-76 typeinfo for QSignalMapper

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSignalMapper
basetype:	typeinfo for QObject

18.2.11.13 Class data for QSocketNotifier

The virtual table for the QSocketNotifier class is described by Table 18-77

Table 18-77 Primary vtable for QSocketNotifier

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSocketNotifier
vfunc[0]:	QSocketNotifier::metaObject() const
vfunc[1]:	QSocketNotifier::qt_metacast(char const*)
vfunc[2]:	QSocketNotifier::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSocketNotifier::~~QSocketNotifier()
vfunc[4]:	QSocketNotifier::~~QSocketNotifier()
vfunc[5]:	QSocketNotifier::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QSocketNotifier class is described by Table 18-78

Table 18-78 typeinfo for QSocketNotifier

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSocketNotifier
basetype:	typeinfo for QObject

18.2.11.14 Class data for QFileSystemWatcher

The virtual table for the QFileSystemWatcher class is described by Table 18-79

Table 18-79 Primary vtable for QFileSystemWatcher

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFileSystemWatcher
vfunc[0]:	QFileSystemWatcher::metaObject() const

vfunc[1]:	QFileSystemWatcher::qt_metacast(char const*)
vfunc[2]:	QFileSystemWatcher::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFileSystemWatcher::~QFileSystemWatcher()
vfunc[4]:	QFileSystemWatcher::~~QFileSystemWatcher()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

18.2.11.15 Interfaces for Qt4 Input/Output

An LSB conforming implementation shall provide the generic functions for Qt4 Input/Output specified in Table 18-80, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-80 libQtCore - Qt4 Input/Output Function Interfaces

_Z10forcepointR11QTextStream [QtCore]	_Z10noshowbaseR11QTextStream [QtCore]
_Z10scientificR11QTextStream [QtCore]	_Z11noforcesignR11QTextStream [QtCore]
_Z12noforcepointR11QTextStream [QtCore]	_Z13lowercasebaseR11QTextStream [QtCore]
_Z13uppercasebaseR11QTextStream [QtCore]	_Z15lowercasedigitsR11QTextStream [QtCore]
_Z15uppercasedigitsR11QTextStream [QtCore]	_Z2wsR11QTextStream [QtCore]
_Z3binR11QTextStream [QtCore]	_Z3bomR11QTextStream [QtCore]
_Z3decR11QTextStream [QtCore]	_Z3hexR11QTextStream [QtCore]
_Z3octR11QTextStream [QtCore]	_Z4endlR11QTextStream [QtCore]
_Z4leftR11QTextStream [QtCore]	_Z5fixedR11QTextStream [QtCore]
_Z5flushR11QTextStream [QtCore]	_Z5resetR11QTextStream [QtCore]
_Z5rightR11QTextStream [QtCore]	_Z6centerR11QTextStream [QtCore]

_Z8showbaseR11QTextStream [QtCore]	_Z9forcesignR11QTextStream [QtCore]
_ZN11QDataStream10writeBytesEPKcj [QtCore]	_ZN11QDataStream11readRawDataEPci [QtCore]
_ZN11QDataStream11resetStatusEv [QtCore]	_ZN11QDataStream11skipRawDataEi [QtCore]
_ZN11QDataStream11unsetDeviceEv [QtCore]	_ZN11QDataStream12setByteOrderENS_9ByteOrderE [QtCore]
_ZN11QDataStream12writeRawDataEPKci [QtCore]	_ZN11QDataStream9readBytesERPcRj [QtCore]
_ZN11QDataStream9setDeviceEP9QIODevice [QtCore]	_ZN11QDataStream9setStatusENS_6StatusE [QtCore]
_ZN11QDataStreamC1EP10QByteArray6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QDataStreamC1EP10QByteArrayi [QtCore]
_ZN11QDataStreamC1EP9QIODevice [QtCore]	_ZN11QDataStreamC1ERK10QByteArray [QtCore]
_ZN11QDataStreamC1Ev [QtCore]	_ZN11QDataStreamC2EP10QByteArray6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN11QDataStreamC2EP10QByteArrayi [QtCore]	_ZN11QDataStreamC2EP9QIODevice [QtCore]
_ZN11QDataStreamC2ERK10QByteArray [QtCore]	_ZN11QDataStreamC2Ev [QtCore]
_ZN11QDataStreamD0Ev [QtCore]	_ZN11QDataStreamD1Ev [QtCore]
_ZN11QDataStreamD2Ev [QtCore]	_ZN11QDataStreamlsEPKc [QtCore]
_ZN11QDataStreamlsEa [QtCore]	_ZN11QDataStreamlsEb [QtCore]
_ZN11QDataStreamlsEd [QtCore]	_ZN11QDataStreamlsEf [QtCore]
_ZN11QDataStreamlsEi [QtCore]	_ZN11QDataStreamlsEs [QtCore]
_ZN11QDataStreamlsEx [QtCore]	_ZN11QDataStreamrsERPc [QtCore]
_ZN11QDataStreamrsERa [QtCore]	_ZN11QDataStreamrsERb [QtCore]
_ZN11QDataStreamrsERd [QtCore]	_ZN11QDataStreamrsERf [QtCore]
_ZN11QDataStreamrsERi [QtCore]	_ZN11QDataStreamrsERs [QtCore]
_ZN11QDataStreamrsERx [QtCore]	_ZN11QTextStream10setPadCharE5QChar [QtCore]
_ZN11QTextStream11resetStatusEv [QtCore]	_ZN11QTextStream11setEncodingENS_8EncodingE [QtCore]
_ZN11QTextStream13setFieldWidthEi [QtCore]	_ZN11QTextStream14setIntegerBaseEi [QtCore]

_ZN11QTextStream14setNumberFlagsE6QFlagsINS_10NumberFlagEE [QtCore]	_ZN11QTextStream14skipWhiteSpaceEv [QtCore]
_ZN11QTextStream17setFieldAlignmentENS_14FieldAlignmentE [QtCore]	_ZN11QTextStream20setAutoDetectUnicodeEb [QtCore]
_ZN11QTextStream21setRealNumberNotationENS_18RealNumberNotationE [QtCore]	_ZN11QTextStream22setRealNumberPrecisionEi [QtCore]
_ZN11QTextStream24setGenerateByteOrderMarkEb [QtCore]	_ZN11QTextStream4readEx [QtCore]
_ZN11QTextStream4seekEx [QtCore]	_ZN11QTextStream5flushEv [QtCore]
_ZN11QTextStream5resetEv [QtCore]	_ZN11QTextStream7readAllEv [QtCore]
_ZN11QTextStream8readLineEx [QtCore]	_ZN11QTextStream8setCodecEP10QTextCodec [QtCore]
_ZN11QTextStream8setCodecEPKc [QtCore]	_ZN11QTextStream9setDeviceEP9QIODevice [QtCore]
_ZN11QTextStream9setStatusENS_6StatusE [QtCore]	_ZN11QTextStream9setStringEP7QString6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN11QTextStreamC1EP10QByteArray6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QTextStreamC1EP7QString6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN11QTextStreamC1EP8_IO_FILE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QTextStreamC1EP9QIODevice [QtCore]
_ZN11QTextStreamC1ERK10QByteArray6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QTextStreamC1Ev [QtCore]
_ZN11QTextStreamC2EP10QByteArray6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QTextStreamC2EP7QString6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN11QTextStreamC2EP8_IO_FILE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QTextStreamC2EP9QIODevice [QtCore]
_ZN11QTextStreamC2ERK10QByteArray6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN11QTextStreamC2Ev [QtCore]
_ZN11QTextStreamD0Ev [QtCore]	_ZN11QTextStreamD1Ev [QtCore]
_ZN11QTextStreamD2Ev [QtCore]	_ZN11QTextStreamIsE5QBool [QtCore]

_ZN11QTextStreamlsE5QChar [QtCore]	_ZN11QTextStreamlsEPKc [QtCore]
_ZN11QTextStreamlsEPKv [QtCore]	_ZN11QTextStreamlsERK10QByteAr ray [QtCore]
_ZN11QTextStreamlsERK7QString [QtCore]	_ZN11QTextStreamlsEc [QtCore]
_ZN11QTextStreamlsEd [QtCore]	_ZN11QTextStreamlsEf [QtCore]
_ZN11QTextStreamlsEi [QtCore]	_ZN11QTextStreamlsEj [QtCore]
_ZN11QTextStreamlsEl [QtCore]	_ZN11QTextStreamlsEm [QtCore]
_ZN11QTextStreamlsEs [QtCore]	_ZN11QTextStreamlsEt [QtCore]
_ZN11QTextStreamlsEx [QtCore]	_ZN11QTextStreamlsEy [QtCore]
_ZN11QTextStreamrsEPc [QtCore]	_ZN11QTextStreamrsER10QByteArr ay [QtCore]
_ZN11QTextStreamrsER5QChar [QtCore]	_ZN11QTextStreamrsER7QString [QtCore]
_ZN11QTextStreamrsERc [QtCore]	_ZN11QTextStreamrsERd [QtCore]
_ZN11QTextStreamrsERf [QtCore]	_ZN11QTextStreamrsERi [QtCore]
_ZN11QTextStreamrsERj [QtCore]	_ZN11QTextStreamrsERl [QtCore]
_ZN11QTextStreamrsERm [QtCore]	_ZN11QTextStreamrsERs [QtCore]
_ZN11QTextStreamrsERt [QtCore]	_ZN11QTextStreamrsERx [QtCore]
_ZN11QTextStreamrsERy [QtCore]	_ZN13QFSFileEngine11currentPathE RK7QString [QtCore]
_ZN13QFSFileEngine11setFileName ERK7QString [QtCore]	_ZN13QFSFileEngine12endEntryList Ev [QtCore]
_ZN13QFSFileEngine14beginEntryLi stE6QFlagsIN4QDir6FilterEERK11QS tringList [QtCore]	_ZN13QFSFileEngine14setCurrentPa thERK7QString [QtCore]
_ZN13QFSFileEngine14setPermissio nsEj [QtCore]	_ZN13QFSFileEngine4copyERK7QSt ring [QtCore]
_ZN13QFSFileEngine4linkERK7QStri ng [QtCore]	_ZN13QFSFileEngine4openE6QFlags IN9QIODevice12OpenModeFlagEE [QtCore]
_ZN13QFSFileEngine4openE6QFlags IN9QIODevice12OpenModeFlagEEP 8_IO_FILE [QtCore]	_ZN13QFSFileEngine4openE6QFlags IN9QIODevice12OpenModeFlagEEi [QtCore]
_ZN13QFSFileEngine4readEPcx [QtCore]	_ZN13QFSFileEngine4seekEx [QtCore]
_ZN13QFSFileEngine5closeEv [QtCore]	_ZN13QFSFileEngine5flushEv [QtCore]
_ZN13QFSFileEngine5writeEPKcx [QtCore]	_ZN13QFSFileEngine6drivesEv [QtCore]

_ZN13QFSFileEngine6removeEv [QtCore]	_ZN13QFSFileEngine6renameERK7QString [QtCore]
_ZN13QFSFileEngine7setSizeEx [QtCore]	_ZN13QFSFileEngine8homePathEv [QtCore]
_ZN13QFSFileEngine8readLineEPcx [QtCore]	_ZN13QFSFileEngine8rootPathEv [QtCore]
_ZN13QFSFileEngine8tempPathEv [QtCore]	_ZN13QFSFileEngine9extensionEN19QAbstractFileEngine9ExtensionEPKNS0_15ExtensionOptionEPNS0_15ExtensionReturnE [QtCore]
_ZN13QFSFileEngineC1ERK7QString [QtCore]	_ZN13QFSFileEngineC1Ev [QtCore]
_ZN13QFSFileEngineC2ERK7QString [QtCore]	_ZN13QFSFileEngineC2Ev [QtCore]
_ZN13QFSFileEngineD0Ev [QtCore]	_ZN13QFSFileEngineD1Ev [QtCore]
_ZN13QFSFileEngineD2Ev [QtCore]	_ZN13QSignalMapper10setMappingEP7QObjectP7QWidget [QtCore]
_ZN13QSignalMapper10setMappingEP7QObjectRK7QString [QtCore]	_ZN13QSignalMapper10setMappingEP7QObjectS1_ [QtCore]
_ZN13QSignalMapper10setMappingEP7QObjecti [QtCore]	_ZN13QSignalMapper11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN13QSignalMapper11qt_metacastEPKc [QtCore]	_ZN13QSignalMapper14removeMappingsEP7QObject [QtCore]
_ZN13QSignalMapper3mapEP7QObject [QtCore]	_ZN13QSignalMapper3mapEv [QtCore]
_ZN13QSignalMapper6mappedEP7QObject [QtCore]	_ZN13QSignalMapper6mappedEP7QWidget [QtCore]
_ZN13QSignalMapper6mappedERK7QString [QtCore]	_ZN13QSignalMapper6mappedEi [QtCore]
_ZN13QSignalMapperC1EP7QObject [QtCore]	_ZN13QSignalMapperC1EP7QObjectPKc [QtCore]
_ZN13QSignalMapperC2EP7QObject [QtCore]	_ZN13QSignalMapperC2EP7QObjectPKc [QtCore]
_ZN13QSignalMapperD0Ev [QtCore]	_ZN13QSignalMapperD1Ev [QtCore]
_ZN13QSignalMapperD2Ev [QtCore]	_ZN14QTemporaryFile11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN14QTemporaryFile11qt_metacastEPKc [QtCore]	_ZN14QTemporaryFile13setAutoRemoveEb [QtCore]
_ZN14QTemporaryFile15createLocalFileER5QFile [QtCore]	_ZN14QTemporaryFile15setFileTemplateERK7QString [QtCore]

_ZN14QTemporaryFile4openE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN14QTemporaryFileC1EP7QObject [QtCore]
_ZN14QTemporaryFileC1ERK7QString [QtCore]	_ZN14QTemporaryFileC1ERK7QStringP7QObject [QtCore]
_ZN14QTemporaryFileC1Ev [QtCore]	_ZN14QTemporaryFileC2EP7QObject [QtCore]
_ZN14QTemporaryFileC2ERK7QString [QtCore]	_ZN14QTemporaryFileC2ERK7QStringP7QObject [QtCore]
_ZN14QTemporaryFileC2Ev [QtCore]	_ZN14QTemporaryFileD0Ev [QtCore]
_ZN14QTemporaryFileD1Ev [QtCore]	_ZN14QTemporaryFileD2Ev [QtCore]
_ZN15QSocketNotifier10setEnabledEb [QtCore]	_ZN15QSocketNotifier11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN15QSocketNotifier11qt_metacastEPKc [QtCore]	_ZN15QSocketNotifier5eventEP6QEvent [QtCore]
_ZN15QSocketNotifier9activatedEi [QtCore]	_ZN15QSocketNotifierC1EiNS_4TypeEP7QObject [QtCore]
_ZN15QSocketNotifierC1EiNS_4TypeEP7QObjectPKc [QtCore]	_ZN15QSocketNotifierC2EiNS_4TypeEP7QObject [QtCore]
_ZN15QSocketNotifierC2EiNS_4TypeEP7QObjectPKc [QtCore]	_ZN15QSocketNotifierD0Ev [QtCore]
_ZN15QSocketNotifierD1Ev [QtCore]	_ZN15QSocketNotifierD2Ev [QtCore]
_ZN18QFileSystemWatcher10removePathERK7QString [QtXml]	_ZN18QFileSystemWatcher11fileChangedERK7QString [QtXml]
_ZN18QFileSystemWatcher11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN18QFileSystemWatcher11qt_metacastEPKc [QtXml]
_ZN18QFileSystemWatcher11removePathsERK11QStringList [QtXml]	_ZN18QFileSystemWatcher16directoryChangedERK7QString [QtXml]
_ZN18QFileSystemWatcher7addPathERK7QString [QtXml]	_ZN18QFileSystemWatcher8addPathsERK11QStringList [QtXml]
_ZN18QFileSystemWatcherC1EP7QObject [QtXml]	_ZN18QFileSystemWatcherC1ERK11QStringListP7QObject [QtXml]
_ZN18QFileSystemWatcherC2EP7QObject [QtXml]	_ZN18QFileSystemWatcherC2ERK11QStringListP7QObject [QtXml]
_ZN18QFileSystemWatcherD0Ev [QtXml]	_ZN18QFileSystemWatcherD1Ev [QtXml]
_ZN18QFileSystemWatcherD2Ev [QtXml]	_ZN19QAbstractFileEngine11setFileNameERK7QString [QtCore]

_ZN19QAbstractFileEngine12endEntryListEv [QtCore]	_ZN19QAbstractFileEngine14beginEntryListE6QFlagsIN4QDir6FilterEERK11QStringList [QtCore]
_ZN19QAbstractFileEngine14setPermissionsEj [QtCore]	_ZN19QAbstractFileEngine4copyERK7QString [QtCore]
_ZN19QAbstractFileEngine4linkERK7QString [QtCore]	_ZN19QAbstractFileEngine4openE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN19QAbstractFileEngine4readEPcx [QtCore]	_ZN19QAbstractFileEngine4seekEx [QtCore]
_ZN19QAbstractFileEngine5closeEv [QtCore]	_ZN19QAbstractFileEngine5flushEv [QtCore]
_ZN19QAbstractFileEngine5writeEPKcx [QtCore]	_ZN19QAbstractFileEngine6createERK7QString [QtCore]
_ZN19QAbstractFileEngine6removeEv [QtCore]	_ZN19QAbstractFileEngine6renameERK7QString [QtCore]
_ZN19QAbstractFileEngine7setSizeEx [QtCore]	_ZN19QAbstractFileEngine8readLineEPcx [QtCore]
_ZN19QAbstractFileEngine8setErrorEN5QFile9FileErrorERK7QString [QtCore]	_ZN19QAbstractFileEngine9extensionENS_9ExtensionEPKNS_15ExtensionOptionEPNS_15ExtensionReturnE [QtCore]
_ZN19QAbstractFileEngineC1Ev [QtCore]	_ZN19QAbstractFileEngineC2Ev [QtCore]
_ZN19QAbstractFileEngineD0Ev [QtCore]	_ZN19QAbstractFileEngineD1Ev [QtCore]
_ZN19QAbstractFileEngineD2Ev [QtCore]	_ZN26QAbstractFileEngineHandlerC1Ev [QtCore]
_ZN26QAbstractFileEngineHandlerC2Ev [QtCore]	_ZN26QAbstractFileEngineHandlerD0Ev [QtCore]
_ZN26QAbstractFileEngineHandlerD1Ev [QtCore]	_ZN26QAbstractFileEngineHandlerD2Ev [QtCore]
_ZN4QDir10setCurrentERK7QString [QtCore]	_ZN4QDir10setSortingE6QFlagsINS_8SortFlagEE [QtCore]
_ZN4QDir11currentPathEv [QtCore]	_ZN4QDir12makeAbsoluteEv [QtCore]
_ZN4QDir13setNameFilterERK7QString [QtCore]	_ZN4QDir14isRelativePathERK7QString [QtCore]
_ZN4QDir14setNameFiltersERK11QStringList [QtCore]	_ZN4QDir15setMatchAllDirsEb [QtCore]
_ZN4QDir17convertSeparatorsERK7QString [QtCore]	_ZN4QDir18toNativeSeparatorsERK7QString [QtXml]

_ZN4QDir20fromNativeSeparatorsERK7QString [QtXml]	_ZN4QDir21addResourceSearchPathERK7QString [QtCore]
_ZN4QDir21nameFiltersFromStringERK7QString [LSB]	_ZN4QDir2cdERK7QString [QtCore]
_ZN4QDir4cdUpEv [QtCore]	_ZN4QDir5matchERK11QStringListRK7QString [QtCore]
ZN4QDir5matchERK7QStringS2 [QtCore]	_ZN4QDir6drivesEv [QtCore]
_ZN4QDir6removeERK7QString [QtCore]	_ZN4QDir6renameERK7QStringS2_ [QtCore]
_ZN4QDir7setPathERK7QString [QtCore]	_ZN4QDir8homePathEv [QtCore]
_ZN4QDir8rootPathEv [QtCore]	_ZN4QDir8tempPathEv [QtCore]
_ZN4QDir9cleanPathERK7QString [QtCore]	_ZN4QDir9separatorEv [QtCore]
_ZN4QDir9setFilterE6QFlagsINS_6FilterEE [QtCore]	_ZN4QDirC1ERK7QString [QtCore]
_ZN4QDirC1ERK7QStringS2_6QFlagsINS_8SortFlagEES3_INS_6FilterEE [QtCore]	_ZN4QDirC1ERKS_ [QtCore]
_ZN4QDirC2ERK7QString [QtCore]	_ZN4QDirC2ERK7QStringS2_6QFlagsINS_8SortFlagEES3_INS_6FilterEE [QtCore]
ZN4QDirC2ERKS [QtCore]	_ZN4QDirD1Ev [QtCore]
_ZN4QDirD2Ev [QtCore]	_ZN4QDirasERK7QString [QtCore]
ZN4QDirasERKS [QtCore]	_ZN4QUrl10toPunycodeERK7QString [QtCore]
_ZN4QUrl11fromEncodedERK10QByteArray [QtCore]	_ZN4QUrl11fromEncodedERK10QByteArrayNS_11ParsingModeE [QtCore]
_ZN4QUrl11setFileNameERK7QString [QtCore]	_ZN4QUrl11setFragmentERK7QString [QtCore]
_ZN4QUrl11setPasswordERK7QString [QtCore]	_ZN4QUrl11setUserInfoERK7QString [QtCore]
_ZN4QUrl11setUserNameERK7QString [QtCore]	_ZN4QUrl12addQueryItemERK7QStringS2_ [QtCore]
_ZN4QUrl12fromPunycodeERK10QByteArray [QtCore]	_ZN4QUrl12idnWhitelistEv [QtXml]
_ZN4QUrl12setAuthorityERK7QString [QtCore]	_ZN4QUrl13fromLocalFileERK7QString [QtCore]

_ZN4QUrl13setEncodedUrlERK10QByteArray [QtCore]	_ZN4QUrl13setEncodedUrlERK10QByteArrayNS_11ParsingModeE [QtCore]
_ZN4QUrl13setQueryItemsERK5QListI7QStringS2_EE [QtCore]	_ZN4QUrl15removeQueryItemERK7QString [QtCore]
_ZN4QUrl15setEncodedQueryERK10QByteArray [QtCore]	_ZN4QUrl15setIdnWhitelistERK11QStringList [QtXml]
ZN4QUrl17toPercentEncodingERK7QStringRK10QByteArrayS5 [QtCore]	_ZN4QUrl18setQueryDelimitersEcc [QtCore]
_ZN4QUrl19fromPercentEncodingERK10QByteArray [QtCore]	_ZN4QUrl19removeAllQueryItemsERK7QString [QtCore]
_ZN4QUrl5clearEv [QtCore]	_ZN4QUrl5toAceERK7QString [QtXml]
_ZN4QUrl6detachEv [QtCore]	_ZN4QUrl6setUrlERK7QString [QtCore]
_ZN4QUrl6setUrlERK7QStringNS_11ParsingModeE [QtCore]	_ZN4QUrl7fromAceERK10QByteArray [QtXml]
_ZN4QUrl7setHostERK7QString [QtCore]	_ZN4QUrl7setPathERK7QString [QtCore]
_ZN4QUrl7setPortEi [QtCore]	_ZN4QUrl9setSchemeERK7QString [QtCore]
_ZN4QUrlC1ERK7QString [QtCore]	_ZN4QUrlC1ERK7QStringNS_11ParsingModeE [QtCore]
ZN4QUrlC1ERKS [QtCore]	_ZN4QUrlC1Ev [QtCore]
_ZN4QUrlC2ERK7QString [QtCore]	_ZN4QUrlC2ERK7QStringNS_11ParsingModeE [QtCore]
ZN4QUrlC2ERKS [QtCore]	_ZN4QUrlC2Ev [QtCore]
_ZN4QUrlD1Ev [QtCore]	_ZN4QUrlD2Ev [QtCore]
_ZN4QUrlaSERK7QString [QtCore]	_ZN4QUrlaSERKS_ [QtCore]
_ZN5QFile10decodeNameERK10QByteArray [QtCore]	_ZN5QFile10encodeNameERK7QString [QtCore]
_ZN5QFile10unsetErrorEv [QtCore]	_ZN5QFile11permissionsERK7QString [QtCore]
_ZN5QFile11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN5QFile11qt_metacastEPKc [QtCore]
_ZN5QFile11setFileNameERK7QString [QtCore]	_ZN5QFile12readLineDataEPcx [QtCore]
_ZN5QFile14setPermissionsE6QFlagsINS_10PermissionEE [QtCore]	_ZN5QFile14setPermissionsERK7QString6QFlagsINS_10PermissionEE [QtCore]

_ZN5QFile19setDecodingFunctionEPF7QStringRK10QByteArrayE [QtCore]	_ZN5QFile19setEncodingFunctionEPF10QByteArrayRK7QStringE [QtCore]
_ZN5QFile4copyERK7QString [QtCore]	_ZN5QFile4copyERK7QStringS2_ [QtCore]
_ZN5QFile4linkERK7QString [QtCore]	_ZN5QFile4linkERK7QStringS2_ [QtCore]
_ZN5QFile4openE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN5QFile4openEP8_IO_FILE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN5QFile4openEi6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN5QFile4seekEx [QtCore]
_ZN5QFile5closeEv [QtCore]	_ZN5QFile5flushEv [QtCore]
_ZN5QFile6existsERK7QString [QtCore]	_ZN5QFile6removeERK7QString [QtCore]
_ZN5QFile6removeEv [QtCore]	_ZN5QFile6renameERK7QString [QtCore]
ZN5QFile6renameERK7QStringS2 [QtCore]	_ZN5QFile6resizeERK7QStringx [QtCore]
_ZN5QFile6resizeEx [QtCore]	_ZN5QFile8readDataEPcx [QtCore]
_ZN5QFile8readLinkERK7QString [QtCore]	_ZN5QFile9writeDataEPKcx [QtCore]
_ZN5QFileC1EP7QObject [QtCore]	_ZN5QFileC1ERK7QString [QtCore]
_ZN5QFileC1ERK7QStringP7QObject [QtCore]	_ZN5QFileC1Ev [QtCore]
_ZN5QFileC2EP7QObject [QtCore]	_ZN5QFileC2ERK7QString [QtCore]
_ZN5QFileC2ERK7QStringP7QObject [QtCore]	_ZN5QFileC2Ev [QtCore]
_ZN5QFileD0Ev [QtCore]	_ZN5QFileD1Ev [QtCore]
_ZN5QFileD2Ev [QtCore]	_ZN7QBuffer11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN7QBuffer11qt_metacastEPKc [QtCore]	_ZN7QBuffer4openE6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN7QBuffer4seekEx [QtCore]	_ZN7QBuffer5closeEv [QtCore]
_ZN7QBuffer6bufferEv [QtCore]	_ZN7QBuffer7setDataERK10QByteArray [QtCore]
_ZN7QBuffer8readDataEPcx [QtCore]	_ZN7QBuffer9setBufferEP10QByteArray [QtCore]
_ZN7QBuffer9writeDataEPKcx [QtCore]	_ZN7QBufferC1EP10QByteArrayP7QObject [QtCore]

_ZN7QBufferC1EP7QObject [QtCore]	_ZN7QBufferC2EP10QByteArrayP7QObject [QtCore]
_ZN7QBufferC2EP7QObject [QtCore]	_ZN7QBufferD0Ev [QtCore]
_ZN7QBufferD1Ev [QtCore]	_ZN7QBufferD2Ev [QtCore]
_ZN8QProcess11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN8QProcess11qt_metacastEPKc [QtCore]
_ZN8QProcess12stateChangedENS_12ProcessStateE [QtCore]	_ZN8QProcess13startDetachedERK7QString [QtCore]
_ZN8QProcess13startDetachedERK7QStringRK11QStringList [QtCore]	_ZN8QProcess14setEnvironmentERK11QStringList [QtCore]
_ZN8QProcess14setReadChannelENS_14ProcessChannelE [QtCore]	_ZN8QProcess14waitForStartedEi [QtCore]
_ZN8QProcess15setProcessStateENS_12ProcessStateE [QtCore]	_ZN8QProcess15waitForFinishedEi [QtCore]
_ZN8QProcess16closeReadChannelENS_14ProcessChannelE [QtCore]	_ZN8QProcess16waitForReadyReadEi [QtCore]
_ZN8QProcess17closeWriteChannelEv [QtCore]	_ZN8QProcess17setupChildProcessEv [QtCore]
_ZN8QProcess17systemEnvironmentEv [QtCore]	_ZN8QProcess18setReadChannelModeENS_18ProcessChannelModeE [QtCore]
_ZN8QProcess19setWorkingDirectoryERK7QString [QtCore]	_ZN8QProcess19waitForBytesWrittenEi [QtCore]
_ZN8QProcess20readAllStandardErrorEv [QtCore]	_ZN8QProcess20setStandardErrorFileERK7QString6QFlagsIN9QIODevice12OpenModeFlagEE [QtXml]
_ZN8QProcess20setStandardInputFileERK7QString [QtXml]	_ZN8QProcess21readAllStandardOutputEv [QtCore]
_ZN8QProcess21setProcessChannelModeENS_18ProcessChannelModeE [QtXml]	_ZN8QProcess21setStandardOutputFileERK7QString6QFlagsIN9QIODevice12OpenModeFlagEE [QtXml]
_ZN8QProcess22readyReadStandardErrorEv [QtCore]	_ZN8QProcess23readyReadStandardOutputEv [QtCore]
ZN8QProcess24setStandardOutputProcessEPS [QtXml]	_ZN8QProcess4killEv [QtCore]
_ZN8QProcess5closeEv [QtCore]	_ZN8QProcess5errorENS_12ProcessErrorE [QtCore]
_ZN8QProcess5startERK7QString6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]	_ZN8QProcess5startERK7QStringRK11QStringList6QFlagsIN9QIODevice12OpenModeFlagEE [QtCore]
_ZN8QProcess7executeERK7QString [QtCore]	_ZN8QProcess7executeERK7QStringRK11QStringList [QtCore]

_ZN8QProcess7startedEv [QtCore]	_ZN8QProcess8finishedEi [QtCore]
_ZN8QProcess8finishedEiNS_10ExitStatusE [QtCore]	_ZN8QProcess8readDataEPcx [QtCore]
_ZN8QProcess9terminateEv [QtCore]	_ZN8QProcess9writeDataEPKcx [QtCore]
_ZN8QProcessC1EP7QObject [QtCore]	_ZN8QProcessC2EP7QObject [QtCore]
_ZN8QProcessD0Ev [QtCore]	_ZN8QProcessD1Ev [QtCore]
_ZN8QProcessD2Ev [QtCore]	_ZN9QFileInfo10setCachingEb [QtCore]
_ZN9QFileInfo12makeAbsoluteEv [QtCore]	_ZN9QFileInfo6detachEv [QtCore]
_ZN9QFileInfo7refreshEv [QtCore]	_ZN9QFileInfo7setFileERK4QDirRK7QString [QtCore]
_ZN9QFileInfo7setFileERK5QFile [QtCore]	_ZN9QFileInfo7setFileERK7QString [QtCore]
_ZN9QFileInfoC1ERK4QDirRK7QString [QtCore]	_ZN9QFileInfoC1ERK5QFile [QtCore]
_ZN9QFileInfoC1ERK7QString [QtCore]	_ZN9QFileInfoC1ERKS_ [QtCore]
_ZN9QFileInfoC1Ev [QtCore]	_ZN9QFileInfoC2ERK4QDirRK7QString [QtCore]
_ZN9QFileInfoC2ERK5QFile [QtCore]	_ZN9QFileInfoC2ERK7QString [QtCore]
ZN9QFileInfoC2ERKS [QtCore]	_ZN9QFileInfoC2Ev [QtCore]
_ZN9QFileInfoD1Ev [QtCore]	_ZN9QFileInfoD2Ev [QtCore]
ZN9QInfoaSERKS [QtCore]	_ZN9QInfoeqERKS_ [QtCore]
_ZN9QIODevice11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN9QIODevice11qt_metacastEPKc [QtCore]
_ZN9QIODevice11resetStatusEv [QtCore]	_ZN9QIODevice11setOpenModeE6QFlagsINS_12OpenModeFlagEE [QtCore]
_ZN9QIODevice12aboutToCloseEv [QtCore]	_ZN9QIODevice12bytesWrittenEx [QtCore]
_ZN9QIODevice12readLineDataEPcx [QtCore]	_ZN9QIODevice14setErrorStringERK7QString [QtCore]
_ZN9QIODevice16waitForReadyReadEi [QtCore]	_ZN9QIODevice18setTextModeEnabledEb [QtCore]
_ZN9QIODevice19waitForBytesWrittenEi [QtCore]	_ZN9QIODevice4openE6QFlagsINS_12OpenModeFlagEE [QtCore]
_ZN9QIODevice4peekEPcx [QtCore]	_ZN9QIODevice4peekEx [QtCore]

_ZN9QIODevice4readEPcx [QtCore]	_ZN9QIODevice4readEx [QtCore]
_ZN9QIODevice4seekEx [QtCore]	_ZN9QIODevice5closeEv [QtCore]
_ZN9QIODevice5resetEv [QtCore]	_ZN9QIODevice5writeEPKcx [QtCore]
_ZN9QIODevice7readAllEv [QtCore]	_ZN9QIODevice8readLineEPcx [QtCore]
_ZN9QIODevice8readLineEx [QtCore]	_ZN9QIODevice9readyReadEv [QtCore]
_ZN9QIODevice9ungetCharEc [QtCore]	_ZN9QIODeviceC1EP7QObject [QtCore]
_ZN9QIODeviceC1Ev [QtCore]	_ZN9QIODeviceC2EP7QObject [QtCore]
_ZN9QIODeviceC2Ev [QtCore]	_ZN9QIODeviceD0Ev [QtCore]
_ZN9QIODeviceD1Ev [QtCore]	_ZN9QIODeviceD2Ev [QtCore]
_ZN9QResource11searchPathsEv [QtXml]	_ZN9QResource11setFileNameERK7QString [QtXml]
_ZN9QResource13addSearchPathERK7QString [QtXml]	_ZN9QResource16registerResourceERK7QStringS2_ [QtXml]
ZN9QResource18unregisterResourceERK7QStringS2 [QtXml]	_ZN9QResource9setLocaleERK7QLocale [QtXml]
_ZN9QResourceC1ERK7QStringRK7QLocale [QtXml]	_ZN9QResourceC2ERK7QStringRK7QLocale [QtXml]
_ZN9QResourceD1Ev [QtXml]	_ZN9QResourceD2Ev [QtXml]
_ZN9QSettings10beginGroupERK7QString [QtCore]	_ZN9QSettings11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN9QSettings11qt_metacastEPKc [QtCore]	_ZN9QSettings13setArrayIndexEi [QtCore]
_ZN9QSettings14beginReadArrayERK7QString [QtCore]	_ZN9QSettings14registerFormatERK7QStringPFbR9QIODeviceR4QMapIS0_8QVariantEEPFbS4_RKS7_EN2Qt15CaseSensitivityE [QtCore]
_ZN9QSettings14setUserIniPathERK7QString [QtCore]	_ZN9QSettings15beginWriteArrayERK7QString [QtCore]
_ZN9QSettings16setSystemIniPathERK7QString [QtCore]	_ZN9QSettings19setFallbacksEnabledEb [QtCore]
_ZN9QSettings4syncEv [QtCore]	_ZN9QSettings5clearEv [QtCore]
_ZN9QSettings5eventEP6QEvent [QtCore]	_ZN9QSettings6removeERK7QString [QtCore]
_ZN9QSettings7setPathENS_6FormatENS_5ScopeERK7QString [QtCore]	_ZN9QSettings8endArrayEv [QtCore]

_ZN9QSettings8endGroupEv [QtCore]	_ZN9QSettings8setValueERK7QStringRK8QVariant [QtCore]
_ZN9QSettingsC1ENS_5ScopeERK7QStringS3_P7QObject [QtCore]	_ZN9QSettingsC1ENS_6FormatENS_5ScopeERK7QStringS4_P7QObject [QtCore]
_ZN9QSettingsC1EP7QObject [QtCore]	_ZN9QSettingsC1ERK7QStringNS_6FormatEP7QObject [QtCore]
_ZN9QSettingsC1ERK7QStringS2_P7QObject [QtCore]	_ZN9QSettingsC2ENS_5ScopeERK7QStringS3_P7QObject [QtCore]
_ZN9QSettingsC2ENS_6FormatENS_5ScopeERK7QStringS4_P7QObject [QtCore]	_ZN9QSettingsC2EP7QObject [QtCore]
_ZN9QSettingsC2ERK7QStringNS_6FormatEP7QObject [QtCore]	_ZN9QSettingsC2ERK7QStringS2_P7QObject [QtCore]
_ZN9QSettingsD0Ev [QtCore]	_ZN9QSettingsD1Ev [QtCore]
_ZN9QSettingsD2Ev [QtCore]	_ZNK11QDataStream5atEndEv [QtCore]
_ZNK11QDataStream6statusEv [QtCore]	_ZNK11QTextStream10fieldWidthEv [QtCore]
_ZNK11QTextStream11integerBaseEv [QtCore]	_ZNK11QTextStream11numberFlagsEv [QtCore]
_ZNK11QTextStream14fieldAlignmentEv [QtCore]	_ZNK11QTextStream17autoDetectUnicodeEv [QtCore]
_ZNK11QTextStream18realNumberNotationEv [QtCore]	_ZNK11QTextStream19realNumberPrecisionEv [QtCore]
_ZNK11QTextStream21generateByteOrderMarkEv [QtCore]	_ZNK11QTextStream3posEv [QtXml]
_ZNK11QTextStream5atEndEv [QtCore]	_ZNK11QTextStream5codecEv [QtCore]
_ZNK11QTextStream6deviceEv [QtCore]	_ZNK11QTextStream6statusEv [QtCore]
_ZNK11QTextStream6stringEv [QtCore]	_ZNK11QTextStream7padCharEv [QtCore]
_ZNK13QFSFileEngine12isSequentialEv [QtCore]	_ZNK13QFSFileEngine13caseSensitiveEv [QtCore]
_ZNK13QFSFileEngine14isRelativePathEv [QtCore]	_ZNK13QFSFileEngine17supportsExtensionEN19QAbstractFileEngine9ExtensionE [QtCore]
_ZNK13QFSFileEngine3posEv [QtCore]	_ZNK13QFSFileEngine4sizeEv [QtCore]
_ZNK13QFSFileEngine5mkdirERK7QStringb [QtCore]	_ZNK13QFSFileEngine5ownerEN19QAbstractFileEngine9FileOwnerE [QtCore]

_ZNK13QFSFileEngine5rmdirERK7QStringb [QtCore]	_ZNK13QFSFileEngine6handleEv [QtCore]
_ZNK13QFSFileEngine7ownerIdEN19QAbstractFileEngine9FileOwnerE [QtCore]	_ZNK13QFSFileEngine8fileNameEN19QAbstractFileEngine8FileNameE [QtCore]
_ZNK13QFSFileEngine8fileTimeEN19QAbstractFileEngine8FileTimeE [QtCore]	_ZNK13QFSFileEngine9entryListE6QFlagsIN4QDir6FilterEERK11QStringList [QtCore]
_ZNK13QFSFileEngine9fileFlagsE6QFlagsIN19QAbstractFileEngine8FileFlagEE [QtCore]	_ZNK13QSignalMapper10metaObjectEv [QtCore]
_ZNK13QSignalMapper7mappingEP7QObject [QtCore]	_ZNK13QSignalMapper7mappingEP7QWidget [QtCore]
_ZNK13QSignalMapper7mappingERK7QString [QtCore]	_ZNK13QSignalMapper7mappingEi [QtCore]
_ZNK14QTemporaryFile10autoRemoveEv [QtCore]	_ZNK14QTemporaryFile10fileEngineEv [QtCore]
_ZNK14QTemporaryFile10metaObjectEv [QtCore]	_ZNK14QTemporaryFile12fileTemplateEv [QtCore]
_ZNK14QTemporaryFile8fileNameEv [QtCore]	_ZNK15QSocketNotifier10metaObjectEv [QtCore]
_ZNK18QFileSystemWatcher10metaObjectEv [QtXml]	_ZNK18QFileSystemWatcher11directoriesEv [QtXml]
_ZNK18QFileSystemWatcher5filesEv [QtXml]	_ZNK19QAbstractFileEngine11errorStringEv [QtCore]
_ZNK19QAbstractFileEngine12isSequentialEv [QtCore]	_ZNK19QAbstractFileEngine13caseSensitiveEv [QtCore]
_ZNK19QAbstractFileEngine14isRelativePathEv [QtCore]	_ZNK19QAbstractFileEngine17supportsExtensionENS_9ExtensionE [QtCore]
_ZNK19QAbstractFileEngine3posEv [QtCore]	_ZNK19QAbstractFileEngine4sizeEv [QtCore]
_ZNK19QAbstractFileEngine5errorEv [QtCore]	_ZNK19QAbstractFileEngine5mkdirERK7QStringb [QtCore]
_ZNK19QAbstractFileEngine5ownerENS_9FileOwnerE [QtCore]	_ZNK19QAbstractFileEngine5rmdirERK7QStringb [QtCore]
_ZNK19QAbstractFileEngine6handleEv [QtCore]	_ZNK19QAbstractFileEngine7ownerIdENS_9FileOwnerE [QtCore]
_ZNK19QAbstractFileEngine8fileNameENS_8FileNameE [QtCore]	_ZNK19QAbstractFileEngine8fileTimeENS_8FileTimeE [QtCore]
_ZNK19QAbstractFileEngine9entryListE6QFlagsIN4QDir6FilterEERK11QStringList [QtCore]	_ZNK19QAbstractFileEngine9fileFlagsE6QFlagsINS_8FileFlagEE [QtCore]

_Znk4QDir10isReadableEv [QtCore]	_Znk4QDir10isRelativeEv [QtCore]
_Znk4QDir10nameFilterEv [QtCore]	_Znk4QDir11nameFiltersEv [QtCore]
_Znk4QDir12absolutePathEv [QtCore]	_Znk4QDir12matchAllDirsEv [QtCore]
_Znk4QDir13canonicalPathEv [QtCore]	_Znk4QDir13entryInfoListE6QFlagsINS_6FilterEES0_INS_8SortFlagEE [QtCore]
_Znk4QDir13entryInfoListERK11QStringList6QFlagsINS_6FilterEES3_INS_8SortFlagEE [QtCore]	_Znk4QDir16absoluteFilePathERK7QString [QtCore]
_Znk4QDir16relativeFilePathERK7QString [QtCore]	_Znk4QDir4pathEv [QtCore]
_Znk4QDir5countEv [QtCore]	_Znk4QDir5mkdirERK7QString [QtCore]
_Znk4QDir5rmdirERK7QString [QtCore]	_Znk4QDir6existsERK7QString [QtCore]
_Znk4QDir6existsEv [QtCore]	_Znk4QDir6filterEv [QtCore]
_Znk4QDir6isRootEv [QtCore]	_Znk4QDir6mkpathERK7QString [QtCore]
_Znk4QDir6rmpathERK7QString [QtCore]	_Znk4QDir7dirNameEv [QtCore]
_Znk4QDir7refreshEv [QtCore]	_Znk4QDir7sortingEv [QtCore]
_Znk4QDir8filePathERK7QString [QtCore]	_Znk4QDir9entryListE6QFlagsINS_6FilterEES0_INS_8SortFlagEE [QtCore]
_Znk4QDir9entryListERK11QStringList6QFlagsINS_6FilterEES3_INS_8SortFlagEE [QtCore]	_Znk4QDir9reqERKS_ [QtCore]
_Znk4QDirixEi [QtCore]	_Znk4QUrl10isDetachedEv [QtCore]
Znk4QUrl10isParentOfERKS [QtCore]	_Znk4QUrl10isRelativeEv [QtCore]
_Znk4QUrl10queryItemsEv [QtCore]	_Znk4QUrl11errorStringEv [QtXml]
_Znk4QUrl11hasFragmentEv [QtXml]	_Znk4QUrl11toLocalFileEv [QtCore]
_Znk4QUrl12encodedQueryEv [QtCore]	_Znk4QUrl12hasQueryItemERK7QString [QtCore]
_Znk4QUrl14queryItemValueERK7QString [QtCore]	_Znk4QUrl18allQueryItemValuesERK7QString [QtCore]

_Znk4Qurl18queryPairDelimiterEv [QtCore]	_Znk4Qurl19queryValueDelimiterEv [QtCore]
_Znk4Qurl4hostEv [QtCore]	_Znk4Qurl4pathEv [QtCore]
_Znk4Qurl4portEv [QtCore]	_Znk4Qurl4portEv [QtCore]
_Znk4Qurl6schemeEv [QtCore]	_Znk4Qurl7dirPathEv [QtCore]
_Znk4Qurl7isEmptyEv [QtCore]	_Znk4Qurl7isValidEv [QtCore]
_Znk4Qurl8fileNameEv [QtCore]	_Znk4Qurl8fragmentEv [QtCore]
_Znk4Qurl8hasQueryEv [QtCore]	_Znk4Qurl8passwordEv [QtCore]
Znk4Qurl8resolvedERKS [QtCore]	_Znk4Qurl8toStringE6QFlagsINS_16FormattingOptionEE [QtCore]
_Znk4Qurl8userInfoEv [QtCore]	_Znk4Qurl8userNameEv [QtCore]
_Znk4Qurl9authorityEv [QtCore]	_Znk4Qurl9toEncodedE6QFlagsINS_16FormattingOptionEE [QtCore]
Znk4QurlEqERKS [QtCore]	_Znk4QurlLtERKS_ [LSB]
Znk4QurlNeERKS [QtCore]	_Znk5Qfile10fileEngineEv [LSB]
_Znk5Qfile10metaObjectEv [QtCore]	_Znk5Qfile11permissionsEv [QtCore]
_Znk5Qfile12isSequentialEv [QtCore]	_Znk5Qfile3posEv [QtCore]
_Znk5Qfile4sizeEv [QtCore]	_Znk5Qfile5atEndEv [QtCore]
_Znk5Qfile5errorEv [QtCore]	_Znk5Qfile6existsEv [QtCore]
_Znk5Qfile6handleEv [QtCore]	_Znk5Qfile8fileNameEv [QtCore]
_Znk5Qfile8readLinkEv [QtCore]	_Znk7Qbuffer10metaObjectEv [QtCore]
_Znk7Qbuffer11canReadLineEv [QtCore]	_Znk7Qbuffer3posEv [QtCore]
_Znk7Qbuffer4dataEv [QtCore]	_Znk7Qbuffer4sizeEv [QtCore]
_Znk7Qbuffer5atEndEv [QtCore]	_Znk7Qbuffer6bufferEv [QtCore]
_Znk8QProcess10exitStatusEv [QtCore]	_Znk8QProcess10metaObjectEv [QtCore]
_Znk8QProcess11canReadLineEv [QtCore]	_Znk8QProcess11environmentEv [QtCore]
_Znk8QProcess11readChannelEv [QtCore]	_Znk8QProcess12bytesToWriteEv [QtCore]
_Znk8QProcess12isSequentialEv [QtCore]	_Znk8QProcess14bytesAvailableEv [QtCore]
_Znk8QProcess15readChannelModeEv [QtCore]	_Znk8QProcess16workingDirectoryEv [QtCore]

_Znk8QProcess18processChannelModeEv [QtXml]	_Znk8QProcess3pidEv [QtCore]
_Znk8QProcess5atEndEv [QtCore]	_Znk8QProcess5errorEv [QtCore]
_Znk8QProcess5stateEv [QtCore]	_Znk8QProcess8exitCodeEv [QtCore]
_Znk9QFileInfo10isReadableEv [QtCore]	_Znk9QFileInfo10isRelativeEv [QtCore]
_Znk9QFileInfo10isWritableEv [QtCore]	_Znk9QFileInfo10permissionE6QFlagsIN5QFile10PermissionEE [QtCore]
_Znk9QFileInfo11absoluteDirEv [QtCore]	_Znk9QFileInfo11permissionsEv [QtCore]
_Znk9QFileInfo12absolutePathEv [QtCore]	_Znk9QFileInfo12isExecutableEv [QtCore]
_Znk9QFileInfo12lastModifiedEv [QtCore]	_Znk9QFileInfo13canonicalPathEv [QtCore]
_Znk9QFileInfo14completeSuffixEv [QtCore]	_Znk9QFileInfo16absoluteFilePathEv [QtCore]
_Znk9QFileInfo16completeBaseNameEv [QtCore]	_Znk9QFileInfo17canonicalFilePathEv [QtCore]
_Znk9QFileInfo3dirEv [QtCore]	_Znk9QFileInfo3dirEv [QtCore]
_Znk9QFileInfo4pathEv [QtCore]	_Znk9QFileInfo4sizeEv [QtCore]
_Znk9QFileInfo5groupEv [QtCore]	_Znk9QFileInfo5isDirEv [QtCore]
_Znk9QFileInfo5ownerEv [QtCore]	_Znk9QFileInfo6existsEv [QtCore]
_Znk9QFileInfo6isFileEv [QtCore]	_Znk9QFileInfo6isRootEv [QtCore]
_Znk9QFileInfo6suffixEv [QtCore]	_Znk9QFileInfo7cachingEv [QtCore]
_Znk9QFileInfo7createdEv [QtCore]	_Znk9QFileInfo7groupIdEv [QtCore]
_Znk9QFileInfo7ownerIdEv [QtCore]	_Znk9QFileInfo8baseNameEv [QtCore]
_Znk9QFileInfo8fileNameEv [QtCore]	_Znk9QFileInfo8filePathEv [QtCore]
_Znk9QFileInfo8isHiddenEv [QtCore]	_Znk9QFileInfo8lastReadEv [QtCore]
_Znk9QFileInfo8readLinkEv [QtCore]	_Znk9QFileInfo9isSymLinkEv [QtCore]
Znk9QFileInfoeqERKS [QtCore]	_Znk9QIODevice10isReadableEv [QtCore]
_Znk9QIODevice10isWritableEv [QtCore]	_Znk9QIODevice10metaObjectEv [QtCore]
_Znk9QIODevice11canReadLineEv [QtCore]	_Znk9QIODevice11errorStringEv [QtCore]

_ZNK9QIODevice12bytesToWriteEv [QtCore]	_ZNK9QIODevice12isSequentialEv [QtCore]
_ZNK9QIODevice14bytesAvailableEv [QtCore]	_ZNK9QIODevice17isTextModeEnabledEv [QtCore]
_ZNK9QIODevice3posEv [QtCore]	_ZNK9QIODevice4sizeEv [QtCore]
_ZNK9QIODevice5atEndEv [QtCore]	_ZNK9QIODevice6isOpenEv [QtCore]
_ZNK9QIODevice6statusEv [QtCore]	_ZNK9QIODevice8openModeEv [QtCore]
_ZNK9QResource12isCompressedEv [QtXml]	_ZNK9QResource16absoluteFilePathEv [QtXml]
_ZNK9QResource4dataEv [QtXml]	_ZNK9QResource4sizeEv [QtXml]
_ZNK9QResource5isDirEv [QtXml]	_ZNK9QResource6localeEv [QtXml]
_ZNK9QResource7isValidEv [QtXml]	_ZNK9QResource8childrenEv [QtXml]
_ZNK9QResource8fileNameEv [QtXml]	_ZNK9QSettings10isWritableEv [QtCore]
_ZNK9QSettings10metaObjectEv [QtCore]	_ZNK9QSettings11childGroupsEv [QtCore]
_ZNK9QSettings16fallbacksEnabledEv [QtCore]	_ZNK9QSettings5groupEv [QtCore]
_ZNK9QSettings5valueERK7QStringRK8QVariant [QtCore]	_ZNK9QSettings6statusEv [QtCore]
_ZNK9QSettings7allKeysEv [QtCore]	_ZNK9QSettings8containsERK7QString [QtCore]
_ZNK9QSettings8fileNameEv [QtCore]	_ZNK9QSettings9childKeysEv [QtCore]
_Zls6QDebugRK4QUrl [QtCore]	_ZlsR11QDataStreamRK4QUrl [QtCore]
_ZrsR11QDataStreamR4QUrl [QtCore]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Input/Output specified in Table 18-81, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-81 libQtCore - Qt4 Input/Output Deprecated Function Interfaces

_ZN11QDataStream11unsetDeviceEv [QtCore]	_ZN4QDir17convertSeparatorsERK7QString [QtCore]

_ZN4QDirasERK7QString [QtCore]	_ZN8QProcess18setReadChannelModeENS_18ProcessChannelModeE [QtCore]
_ZN8QProcess8finishedEi [QtCore]	_ZN9QSettings14setUserIniPathERK7QString [QtCore]
_ZN9QSettings16setSystemIniPathERK7QString [QtCore]	_ZNK8QProcess15readChannelModeEv [QtCore]

18.2.12 Qt4 Core Application

18.2.12.1 Class data for QCoreApplication

The virtual table for the QCoreApplication class is described by Table 18-82

Table 18-82 Primary vtable for QCoreApplication

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCoreApplication
vfunc[0]:	QCoreApplication::metaObject() const
vfunc[1]:	QCoreApplication::qt_metacast(char const*)
vfunc[2]:	QCoreApplication::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QCoreApplication::~QCoreApplication()
vfunc[4]:	QCoreApplication::~QCoreApplication()
vfunc[5]:	QCoreApplication::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QCoreApplication::notify(QObject*, QEvent*)
vfunc[13]:	QCoreApplication::compressEvent(QEvent*, QObject*, QPostEventList*)

The Run Time Type Information for the QCoreApplication class is described by Table 18-83

Table 18-83 typeinfo for QCoreApplication

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QCoreApplication
basetype:	typeinfo for QObject

18.2.12.2 Interfaces for Qt4 Core Application

An LSB conforming implementation shall provide the generic functions for Qt4 Core Application specified in Table 18-84, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-84 libQtCore - Qt4 Core Application Function Interfaces

_ZN16QCoreApplication10enter_loopEv [QtCore]	_ZN16QCoreApplication10startingUpEv [QtCore]
_ZN16QCoreApplication10unixSignalEi [LSB]	_ZN16QCoreApplication11aboutToQuitEv [QtCore]
_ZN16QCoreApplication11closingDownEv [QtCore]	_ZN16QCoreApplication11filterEventEPvPI [QtCore]
_ZN16QCoreApplication11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN16QCoreApplication11qt_metacastEPKc [QtCore]
_ZN16QCoreApplication12libraryPathsEv [QtCore]	_ZN16QCoreApplication12setAttributeEN2Qt20ApplicationAttributeEb [QtXml]
_ZN16QCoreApplication13compressEventEP6QEventP7QObjectP14QPostEventList [QtCore]	_ZN16QCoreApplication13processEventsE6QFlagsIN10QEventLoop17ProcessEventFlagEE [QtCore]
_ZN16QCoreApplication13processEventsE6QFlagsIN10QEventLoop17ProcessEventFlagEEi [QtCore]	_ZN16QCoreApplication13testAttributeEN2Qt20ApplicationAttributeEb [QtXml]
_ZN16QCoreApplication14addLibraryPathERK7QString [QtCore]	_ZN16QCoreApplication14setEventFilterEPFbPvPIE [QtCore]
_ZN16QCoreApplication15applicationNameEv [QtCore]	_ZN16QCoreApplication15setLibraryPathsERK11QStringList [QtCore]
_ZN16QCoreApplication15watchUnixSignalEib [QtCore]	_ZN16QCoreApplication16hasPendingEventsEv [QtCore]
_ZN16QCoreApplication16organizationNameEv [QtCore]	_ZN16QCoreApplication16removeTranslatorEP11QTranslator [QtCore]
_ZN16QCoreApplication16sendPostedEventsEP7QObjecti [QtCore]	_ZN16QCoreApplication17installTranslatorEP11QTranslator [QtCore]

_ZN16QCoreApplication17removeLibraryPathERK7QString [QtCore]	_ZN16QCoreApplication18applicationDirPathEv [QtCore]
_ZN16QCoreApplication18organizationDomainEv [QtCore]	_ZN16QCoreApplication18removePostedEventsEP7QObject [QtCore]
_ZN16QCoreApplication18setApplicationNameERK7QString [QtCore]	_ZN16QCoreApplication19applicationFilePathEv [QtCore]
_ZN16QCoreApplication19setOrganizationNameERK7QString [QtCore]	_ZN16QCoreApplication21setOrganizationDomainERK7QString [QtCore]
_ZN16QCoreApplication4argcEv [QtCore]	_ZN16QCoreApplication4argvEv [QtCore]
_ZN16QCoreApplication4execEv [QtCore]	_ZN16QCoreApplication4exitEi [QtCore]
_ZN16QCoreApplication4quitEv [QtCore]	_ZN16QCoreApplication5eventEP6QEvent [QtCore]
_ZN16QCoreApplication5flushEv [QtCore]	_ZN16QCoreApplication6notifyEP7QObjectP6QEvent [QtCore]
_ZN16QCoreApplication9argumentsEv [QtCore]	_ZN16QCoreApplication9exit_loopEv [QtCore]
_ZN16QCoreApplication9loopLevelEv [QtCore]	_ZN16QCoreApplication9postEventEP7QObjectP6QEvent [QtCore]
_ZN16QCoreApplication9translateEPKcS1_S1_NS_8EncodingE [QtCore]	_ZN16QCoreApplication9translateEPKcS1_S1_NS_8EncodingEi [QtXml]
_ZN16QCoreApplicationC1ERiPPc [QtCore]	_ZN16QCoreApplicationC2ERiPPc [QtCore]
_ZN16QCoreApplicationD0Ev [QtCore]	_ZN16QCoreApplicationD1Ev [QtCore]
_ZN16QCoreApplicationD2Ev [QtCore]	_ZNK16QCoreApplication10metaObjectEv [QtCore]

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Core Application specified in Table 18-85, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-85 libQtCore - Qt4 Core Application Deprecated Function Interfaces

_ZN16QCoreApplication4argcEv [QtCore]	_ZN16QCoreApplication4argvEv [QtCore]

18.2.13 Qt4 Object Model

18.2.13.1 Class data for QObject

The virtual table for the QObject class is described by Table 18-86

Table 18-86 Primary vtable for QObject

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QObject
vfunc[0]:	QObject::metaObject() const
vfunc[1]:	QObject::qt_metacast(char const*)
vfunc[2]:	QObject::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QObject::~~QObject()
vfunc[4]:	QObject::~~QObject()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QObject class is described by Table 18-87

Table 18-87 typeinfo for QObject

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QObject

18.2.13.2 Class data for QObjectUserData

The virtual table for the QObjectUserData class is described by Table 18-88

Table 18-88 Primary vtable for QObjectUserData

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QObjectUserData
vfunc[0]:	QObjectUserData::~~QObjectUserData()
vfunc[1]:	QObjectUserData::~~QObjectUserData()

The Run Time Type Information for the QObjectUserData class is described by Table 18-89

Table 18-89 typeinfo for QObjectUserData

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QObjectUserData

18.2.13.3 Class data for QObjectCleanupHandler

The virtual table for the QObjectCleanupHandler class is described by Table 18-90

Table 18-90 Primary vtable for QObjectCleanupHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QObjectCleanupHandler
vfunc[0]:	QObjectCleanupHandler::metaObject () const
vfunc[1]:	QObjectCleanupHandler::qt_metacast (char const*)
vfunc[2]:	QObjectCleanupHandler::qt_metacall (QMetaObject::Call, int, void**)
vfunc[3]:	QObjectCleanupHandler::~~QObjectC leanupHandler()
vfunc[4]:	QObjectCleanupHandler::~~QObjectC leanupHandler()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QObjectCleanupHandler class is described by Table 18-91

Table 18-91 typeinfo for QObjectCleanupHandler

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QObjectCleanupHandler
basetype:	typeinfo for QObject

18.2.13.4 Interfaces for Qt4 Object Model

An LSB conforming implementation shall provide the generic functions for Qt4 Object Model specified in Table 18-92, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-92 libQtCore - Qt4 Object Model Function Interfaces

_ZN11QMetaObject10disconnectEPK7QObjectiS2_i [QtCore]	_ZN11QMetaObject11changeGuardEPP7QObjectS1_ [QtCore]
_ZN11QMetaObject11removeGuardEPP7QObject [QtCore]	_ZN11QMetaObject12invokeMethodEP7QObjectPKcN2Qt14ConnectionTypeE22QGenericReturnArgument16QGenericArgumentS7_S7_S7_S7_S7_S7_S7_S7_S7_S7_S7_ [QtCore]
_ZN11QMetaObject14normalizedTypeEPKc [QtXml]	_ZN11QMetaObject16checkConnectArgsEPKcS1_ [QtCore]
_ZN11QMetaObject18connectSlotsByNameEP7QObject [QtCore]	_ZN11QMetaObject19normalizedSignatureEPKc [QtCore]
_ZN11QMetaObject7connectEPK7QObjectiS2_iiPi [LSB]	_ZN11QMetaObject8activateEP7QObjectPKS_iiPPv [QtCore]
_ZN11QMetaObject8activateEP7QObjectPKS_iiPPv [QtCore]	_ZN11QMetaObject8activateEP7QObjectiPPv [QtCore]
_ZN11QMetaObject8activateEP7QObjectiPPv [QtCore]	_ZN11QMetaObject8addGuardEPP7QObject [QtCore]
_ZN13QMetaPropertyC1Ev [QtCore]	_ZN13QMetaPropertyC2Ev [QtCore]
_ZN15QObjectUserDataD0Ev [QtCore]	_ZN15QObjectUserDataD1Ev [QtCore]
_ZN15QObjectUserDataD2Ev [QtCore]	_ZN21QObjectCleanupHandler11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN21QObjectCleanupHandler11qt_metacastEPKc [QtCore]	_ZN21QObjectCleanupHandler3addEP7QObject [QtCore]
_ZN21QObjectCleanupHandler5clearEv [QtCore]	_ZN21QObjectCleanupHandler6removeEP7QObject [QtCore]
_ZN21QObjectCleanupHandlerC1Ev [QtCore]	_ZN21QObjectCleanupHandlerC2Ev [QtCore]
_ZN21QObjectCleanupHandlerD0Ev [QtCore]	_ZN21QObjectCleanupHandlerD1Ev [QtCore]
_ZN21QObjectCleanupHandlerD2Ev [QtCore]	_ZN7QObject10childEventEP11QChildEvent [QtCore]
_ZN7QObject10disconnectEPKS_PKcS1_S3_ [QtCore]	_ZN7QObject10startTimerEi [QtCore]
_ZN7QObject10timerEventEP11QTimerEvent [QtCore]	_ZN7QObject11customEventEP6QEvent [QtCore]

_ZN7QObject11deleteLaterEv [QtCore]	_ZN7QObject11eventFilterEPS_P6QEvent [QtCore]
_ZN7QObject11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]	_ZN7QObject11qt_metacastEPKc [QtCore]
_ZN7QObject11setPropertyEPKcRK8QVariant [QtCore]	_ZN7QObject11setUserDataEjP15QObjectUserData [QtCore]
_ZN7QObject12blockSignalsEb [QtCore]	_ZN7QObject12moveToThreadEP7QThread [QtCore]
_ZN7QObject13connectNotifyEPKc [QtCore]	_ZN7QObject13setObjectNameERK7QString [QtCore]
_ZN7QObject14dumpObjectInfoEv [QtCore]	_ZN7QObject14dumpObjectTreeEv [QtCore]
_ZN7QObject16disconnectNotifyEPKc [QtCore]	_ZN7QObject16registerUserDataEv [QtCore]
ZN7QObject17removeEventFilterEPS [QtCore]	_ZN7QObject18installEventFilterEPS_ [QtCore]
_ZN7QObject5eventEP6QEvent [QtCore]	_ZN7QObject7connectEPKS_PKcS1_S3_N2Qt14ConnectionTypeE [QtCore]
ZN7QObject9destroyedEPS [QtCore]	_ZN7QObject9killTimerEi [QtCore]
ZN7QObject9setParentEPS [QtCore]	_ZN7QObjectC1EPS_ [QtCore]
_ZN7QObjectC1EPS_PKc [QtCore]	_ZN7QObjectC2EPS_ [QtCore]
_ZN7QObjectC2EPS_PKc [QtCore]	_ZN7QObjectD0Ev [QtCore]
_ZN7QObjectD1Ev [QtCore]	_ZN7QObjectD2Ev [QtCore]
_ZN8QVariant10nameToTypeEPKc [QtCore]	_ZN8QVariant10typeToNameENS_4TypeE [QtCore]
_ZN8QVariant12castOrDetachENS_4TypeE [QtCore]	_ZN8QVariant4dataEv [QtCore]
_ZN8QVariant4loadER11QDataStream [QtCore]	_ZN8QVariant5clearEv [QtCore]
_ZN8QVariant6createEiPKv [QtCore]	_ZN8QVariant6detachEv [QtCore]
_ZN8QVariant7convertENS_4TypeE [QtCore]	_ZN8QVariantC1EN2Qt11GlobalColorE [QtXml]
_ZN8QVariantC1ENS_4TypeE [QtCore]	_ZN8QVariantC1EPKc [QtCore]
_ZN8QVariantC1ER11QDataStream [QtCore]	_ZN8QVariantC1ERK10QByteArray [QtCore]
_ZN8QVariantC1ERK11QStringList [QtCore]	_ZN8QVariantC1ERK13QLatin1String [QtCore]

_ZN8QVariantC1ERK4QMapI7QStringS_E [QtCore]	_ZN8QVariantC1ERK4QUrl [QtCore]
_ZN8QVariantC1ERK5QChar [QtCore]	_ZN8QVariantC1ERK5QDate [QtCore]
_ZN8QVariantC1ERK5QLine [QtCore]	_ZN8QVariantC1ERK5QListIS_E [QtCore]
_ZN8QVariantC1ERK5QRect [QtCore]	_ZN8QVariantC1ERK5QSize [QtCore]
_ZN8QVariantC1ERK5QTime [QtCore]	_ZN8QVariantC1ERK6QLineF [QtCore]
_ZN8QVariantC1ERK6QPoint [QtCore]	_ZN8QVariantC1ERK6QRectF [QtCore]
_ZN8QVariantC1ERK6QSizeF [QtCore]	_ZN8QVariantC1ERK7QLocale [QtCore]
_ZN8QVariantC1ERK7QPointF [QtCore]	_ZN8QVariantC1ERK7QRegExp [QtCore]
_ZN8QVariantC1ERK7QString [QtCore]	_ZN8QVariantC1ERK9QByteArray [QtCore]
_ZN8QVariantC1ERK9QDateTime [QtCore]	_ZN8QVariantC1ERKS_ [QtCore]
_ZN8QVariantC1Eb [QtCore]	_ZN8QVariantC1Ed [QtCore]
_ZN8QVariantC1Ei [QtCore]	_ZN8QVariantC1EiPKv [QtCore]
_ZN8QVariantC1Ej [QtCore]	_ZN8QVariantC1Ex [QtCore]
_ZN8QVariantC1Ey [QtCore]	_ZN8QVariantC2EN2Qt11GlobalColorE [QtXml]
_ZN8QVariantC2ENS_4TypeE [QtCore]	_ZN8QVariantC2EPKc [QtCore]
_ZN8QVariantC2ER11QDataStream [QtCore]	_ZN8QVariantC2ERK10QByteArray [QtCore]
_ZN8QVariantC2ERK11QStringList [QtCore]	_ZN8QVariantC2ERK13QLatin1String [QtCore]
_ZN8QVariantC2ERK4QMapI7QStringS_E [QtCore]	_ZN8QVariantC2ERK4QUrl [QtCore]
_ZN8QVariantC2ERK5QChar [QtCore]	_ZN8QVariantC2ERK5QDate [QtCore]
_ZN8QVariantC2ERK5QLine [QtCore]	_ZN8QVariantC2ERK5QListIS_E [QtCore]
_ZN8QVariantC2ERK5QRect [QtCore]	_ZN8QVariantC2ERK5QSize [QtCore]
_ZN8QVariantC2ERK5QTime [QtCore]	_ZN8QVariantC2ERK6QLineF [QtCore]

_ZN8QVariantC2ERK6QPoint [QtCore]	_ZN8QVariantC2ERK6QRectF [QtCore]
_ZN8QVariantC2ERK6QSizeF [QtCore]	_ZN8QVariantC2ERK7QLocale [QtCore]
_ZN8QVariantC2ERK7QPointF [QtCore]	_ZN8QVariantC2ERK7QRegExp [QtCore]
_ZN8QVariantC2ERK7QString [QtCore]	_ZN8QVariantC2ERK9QByteArray [QtCore]
_ZN8QVariantC2ERK9QDateTime [QtCore]	_ZN8QVariantC2ERKS_ [QtCore]
_ZN8QVariantC2Eb [QtCore]	_ZN8QVariantC2Ed [QtCore]
_ZN8QVariantC2Ei [QtCore]	_ZN8QVariantC2EiPKv [QtCore]
_ZN8QVariantC2Ej [QtCore]	_ZN8QVariantC2Ex [QtCore]
_ZN8QVariantC2Ey [QtCore]	_ZN8QVariantD1Ev [QtCore]
_ZN8QVariantD2Ev [QtCore]	_ZN8QVariantaSERKS_ [QtCore]
_ZN9QMetaType12isRegisteredEi [QtCore]	_ZN9QMetaType12registerTypeEPKcPFvPvEPFS2_PKvE [LSB]
_ZN9QMetaType23registerStreamOperatorsEPKcPFvR11QDataStreamPKvEPFvS3_PvE [QtCore]	_ZN9QMetaType4loadER11QDataStreamPKv [QtCore]
_ZN9QMetaType4saveER11QDataStreamPKv [QtCore]	_ZN9QMetaType4typeEPKc [QtCore]
_ZN9QMetaType7destroyEiPy [QtCore]	_ZN9QMetaType8typeNameEi [QtCore]
_ZN9QMetaType9constructEiPKv [QtCore]	_ZNK11QMetaMethod10attributesEv [QtCore]
_ZNK11QMetaMethod10methodTypeEv [QtCore]	_ZNK11QMetaMethod14parameterNamesEv [QtCore]
_ZNK11QMetaMethod14parameterTypesEv [QtCore]	_ZNK11QMetaMethod3tagEv [QtCore]
_ZNK11QMetaMethod6accessEv [QtCore]	_ZNK11QMetaMethod8typeNameEv [QtCore]
_ZNK11QMetaMethod9signatureEv [QtCore]	_ZNK11QMetaObject10enumeratorEi [QtCore]
_ZNK11QMetaObject11indexOfSlotEPKc [QtCore]	_ZNK11QMetaObject11methodCountEv [QtCore]
_ZNK11QMetaObject12methodOffsetEv [QtCore]	_ZNK11QMetaObject12userPropertyEv [QtXml]
_ZNK11QMetaObject13indexOfMethodEPKc [QtCore]	_ZNK11QMetaObject13indexOfSignalEPKc [QtCore]

_ZNK11QMetaObject13propertyCountEv [QtCore]	_ZNK11QMetaObject14classInfoCountEv [QtCore]
_ZNK11QMetaObject14propertyOffsetEv [QtCore]	_ZNK11QMetaObject15classInfoOffsetEv [QtCore]
_ZNK11QMetaObject15enumeratorCountEv [QtCore]	_ZNK11QMetaObject15indexOfPropertyEPKc [QtCore]
_ZNK11QMetaObject16enumeratorOffsetEv [QtCore]	_ZNK11QMetaObject16indexOfClassInfoEPKc [QtCore]
_ZNK11QMetaObject17indexOfEnumeratorEPKc [QtCore]	_ZNK11QMetaObject2trEPKcS1_ [QtCore]
_ZNK11QMetaObject2trEPKcS1_i [QtXml]	_ZNK11QMetaObject4castEP7QObject [LSB]
_ZNK11QMetaObject6methodEi [QtCore]	_ZNK11QMetaObject6trUtf8EPKcS1_ [QtCore]
_ZNK11QMetaObject6trUtf8EPKcS1_i [QtXml]	_ZNK11QMetaObject8propertyEi [QtCore]
_ZNK11QMetaObject9classInfoEi [QtCore]	_ZNK13QMetaProperty10enumeratorEv [QtCore]
_ZNK13QMetaProperty10isEditableEPK7QObject [QtCore]	_ZNK13QMetaProperty10isEnumTypeEv [QtCore]
_ZNK13QMetaProperty10isFlagTypeEv [QtCore]	_ZNK13QMetaProperty10isReadableEv [QtCore]
_ZNK13QMetaProperty10isWritableEv [QtCore]	_ZNK13QMetaProperty12hasStdCppSetEv [LSB]
_ZNK13QMetaProperty12isDesignableEPK7QObject [QtCore]	_ZNK13QMetaProperty12isResettableEv [QtCore]
_ZNK13QMetaProperty12isScriptableEPK7QObject [QtCore]	_ZNK13QMetaProperty4nameEv [QtCore]
_ZNK13QMetaProperty4readEPK7QObject [QtCore]	_ZNK13QMetaProperty4typeEv [QtCore]
_ZNK13QMetaProperty5resetEP7QObject [QtCore]	_ZNK13QMetaProperty5writeEP7QObjectRK8QVariant [QtCore]
_ZNK13QMetaProperty6isUserEPK7QObject [QtCore]	_ZNK13QMetaProperty8isStoredEPK7QObject [QtCore]
_ZNK13QMetaProperty8typeNameEv [QtCore]	_ZNK13QMetaProperty8userTypeEv [QtXml]
_ZNK14QMetaClassInfo4nameEv [QtCore]	_ZNK14QMetaClassInfo5valueEv [QtCore]
_ZNK21QObjectCleanupHandler10metaObjectEv [QtCore]	_ZNK21QObjectCleanupHandler7isEmptyEv [QtCore]
_ZNK7QObject10metaObjectEv [QtCore]	_ZNK7QObject10objectNameEv [QtCore]

_Znk7QObject20dynamicPropertyNamesEv [QtXml]	_Znk7QObject5childEPKcS1_b [QtCore]
_Znk7QObject6senderEv [QtCore]	_Znk7QObject6threadEv [QtCore]
_Znk7QObject8propertyEPKc [QtCore]	_Znk7QObject8userDataEj [QtCore]
_Znk7QObject9queryListEPKcS1_bb [LSB]	_Znk7QObject9receiversEPKc [QtCore]
_Znk8QVariant10canConvertENS_4TypeE [QtCore]	_Znk8QVariant10toBitArrayEv [QtCore]
_Znk8QVariant10toDateTimeEv [QtCore]	_Znk8QVariant10toLongLongEPb [QtCore]
_Znk8QVariant11toByteArrayEv [QtCore]	_Znk8QVariant11toULongLongEPb [QtCore]
_Znk8QVariant12toStringListEv [QtCore]	_Znk8QVariant3cmpERKS_ [QtCore]
_Znk8QVariant4saveER11QDataStream [QtCore]	_Znk8QVariant4typeEv [QtCore]
_Znk8QVariant5toIntEPb [QtCore]	_Znk8QVariant5toMapEv [QtCore]
_Znk8QVariant5toUrlEv [QtCore]	_Znk8QVariant6isNullEv [QtCore]
_Znk8QVariant6toBoolEv [QtCore]	_Znk8QVariant6toCharEv [QtCore]
_Znk8QVariant6toDateEv [QtCore]	_Znk8QVariant6toLineEv [QtCore]
_Znk8QVariant6toListEv [QtCore]	_Znk8QVariant6toRectEv [QtCore]
_Znk8QVariant6toSizeEv [QtCore]	_Znk8QVariant6toTimeEv [QtCore]
_Znk8QVariant6toUIntEPb [QtCore]	_Znk8QVariant7toLineFEv [QtCore]
_Znk8QVariant7toPointEv [QtCore]	_Znk8QVariant7toRectFEv [QtCore]
_Znk8QVariant7toSizeFEv [QtCore]	_Znk8QVariant8toDoubleEPb [QtCore]
_Znk8QVariant8toLocaleEv [QtCore]	_Znk8QVariant8toPointFev [QtCore]
_Znk8QVariant8toRegExpEv [QtCore]	_Znk8QVariant8toStringEv [QtCore]
_Znk8QVariant8typeNameEv [QtCore]	_Znk8QVariant8userTypeEv [QtCore]
_Znk8QVariant9constDataEv [QtCore]	_Znk9QMetaEnum10keyToValueEPKc [QtCore]
_Znk9QMetaEnum10valueToKeyEi [QtCore]	_Znk9QMetaEnum11keysToValueEPKc [QtCore]
_Znk9QMetaEnum11valueToKeysEi [QtCore]	_Znk9QMetaEnum3keyEi [QtCore]

_ZNK9QMetaEnum4nameEv [QtCore]	_ZNK9QMetaEnum5scopeEv [QtCore]
_ZNK9QMetaEnum5valueEi [QtCore]	_ZNK9QMetaEnum6isFlagEv [QtCore]
_ZNK9QMetaEnum8keyCountEv [QtCore]	_Zls6QDebugPK7QObject [QtCore]
_Zls6QDebugRK8QVariant [QtCore]	_ZlsR11QDataStreamRK8QVariant [QtCore]
_ZrsR11QDataStreamR8QVariant [QtCore]	

18.2.14 Qt4 Threading

18.2.14.1 Class data for QThread

The virtual table for the QThread class is described by Table 18-93

Table 18-93 Primary vtable for QThread

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QThread
vfunc[0]:	QThread::metaObject() const
vfunc[1]:	QThread::qt_metacast(char const*)
vfunc[2]:	QThread::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QThread::~QThread()
vfunc[4]:	QThread::~QThread()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual

The Run Time Type Information for the QThread class is described by Table 18-94

Table 18-94 typeinfo for QThread

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QThread
basetype:	typeinfo for QObject

18.2.14.2 Interfaces for Qt4 Threading

An LSB conforming implementation shall provide the generic functions for Qt4 Threading specified in Table 18-95, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-95 libQtCore - Qt4 Threading Function Interfaces

_ZN10QSemaphore10tryAcquireEi [QtCore]	_ZN10QSemaphore7acquireEi [QtCore]
_ZN10QSemaphore7releaseEi [QtCore]	_ZN10QSemaphoreC1Ev [QtCore]
_ZN10QSemaphoreC2Ei [QtCore]	_ZN10QSemaphoreD1Ev [QtCore]
_ZN10QSemaphoreD2Ev [QtCore]	_ZN14QReadWriteLock11lockForReadEv [QtCore]
_ZN14QReadWriteLock12lockForWriteEv [QtCore]	_ZN14QReadWriteLock14tryLockForReadEv [QtCore]
_ZN14QReadWriteLock15tryLockForWriteEv [QtCore]	_ZN14QReadWriteLock6unlockEv [QtCore]
_ZN14QReadWriteLockC1Ev [QtCore]	_ZN14QReadWriteLockC2Ev [QtCore]
_ZN14QReadWriteLockD1Ev [QtCore]	_ZN14QReadWriteLockD2Ev [QtCore]
_ZN14QWaitCondition4waitEP6QM utexm [QtCore]	_ZN14QWaitCondition7wakeAllEv [QtCore]
_ZN14QWaitCondition7wakeOneEv [QtCore]	_ZN14QWaitConditionC1Ev [QtCore]
_ZN14QWaitConditionC2Ev [QtCore]	_ZN14QWaitConditionD1Ev [QtCore]
_ZN14QWaitConditionD2Ev [QtCore]	_ZN18QThreadStorageData3setEPv [QtCore]
_ZN18QThreadStorageData6finishEPv [QtCore]	_ZN18QThreadStorageDataC1EPFvPvE [QtCore]
_ZN18QThreadStorageDataC2EPFvPvE [QtCore]	_ZN18QThreadStorageDataD1Ev [QtCore]
_ZN18QThreadStorageDataD2Ev [QtCore]	_ZN6QMutex4lockEv [QtCore]
_ZN6QMutex6unlockEv [QtCore]	_ZN6QMutex7tryLockEv [QtCore]

_ZN6QMutexC1ENS_13RecursionModeE [QtCore]	_ZN6QMutexC2ENS_13RecursionModeE [QtCore]
_ZN6QMutexD1Ev [QtCore]	_ZN6QMutexD2Ev [QtCore]
_ZN7QThread10terminatedEv [QtCore]	_ZN7QThread11qt_metacallEN11QMetaObject4CallEiPPv [QtCore]
_ZN7QThread11qt_metacastEPKc [QtCore]	_ZN7QThread11setPriorityENS_8PriorityE [QtCore]
_ZN7QThread12setStackSizeEj [QtCore]	_ZN7QThread13currentThreadEv [QtCore]
_ZN7QThread15currentThreadIdEv [QtCore]	_ZN7QThread21setTerminationEnabledEb [QtCore]
_ZN7QThread4execEv [QtCore]	_ZN7QThread4exitEi [QtCore]
_ZN7QThread4quitEv [QtCore]	_ZN7QThread4waitEm [QtCore]
_ZN7QThread5sleepEm [QtCore]	_ZN7QThread5startENS_8PriorityE [QtCore]
_ZN7QThread6msleepEm [QtCore]	_ZN7QThread6usleepEm [QtCore]
_ZN7QThread7startedEv [QtCore]	_ZN7QThread8finishedEv [QtCore]
_ZN7QThread9terminateEv [QtCore]	_ZN7QThreadC1EP7QObject [QtCore]
_ZN7QThreadC2EP7QObject [QtCore]	_ZN7QThreadD0Ev [QtCore]
_ZN7QThreadD1Ev [QtCore]	_ZN7QThreadD2Ev [QtCore]
_ZNK10QSemaphore9availableEv [QtCore]	_ZNK18QThreadStorageData3getEv [QtCore]
_ZNK7QThread10isFinishedEv [QtCore]	_ZNK7QThread10metaObjectEv [QtCore]
_ZNK7QThread8priorityEv [QtCore]	_ZNK7QThread9isRunningEv [QtCore]
_ZNK7QThread9stackSizeEv [QtCore]	

18.2.15 Qt4 2D Graphics

18.2.15.1 Interfaces for Qt4 2D Graphics

An LSB conforming implementation shall provide the generic functions for Qt4 2D Graphics specified in Table 18-96, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-96 libQtCore - Qt4 2D Graphics Function Interfaces

_ZN5QRect10moveCenterERK6QPoint [QtCore]	_ZN5QSize5scaleERKS_N2Qt15AspectRatioModeE [QtCore]

_ZN5QSize9transposeEv [QtCore]	_ZN6QSizeF5scaleERKS_N2Qt15AspectRatioModeE [QtCore]
_ZN6QSizeF9transposeEv [QtCore]	_ZNK5QRect10intersectsERKS_ [QtCore]
_ZNK5QRect10normalizedEv [QtCore]	_ZNK5QRect8containsERK6QPointb [QtCore]
_ZNK5QRect8containsERKS_b [QtCore]	_ZNK5QRectanERKS_ [QtCore]
ZNK5QRectoERKS [QtCore]	_ZNK6QLineF10unitVectorEv [QtCore]
ZNK6QLineF5angleERKS [QtCore]	_ZNK6QLineF6isNullEv [QtCore]
_ZNK6QLineF6lengthEv [QtCore]	_ZNK6QLineF9intersectERKS_P7QPointF [QtCore]
_ZNK6QPoint15manhattanLengthEv [QtCore]	_ZNK6QRectF10intersectsERKS_ [QtCore]
_ZNK6QRectF10normalizedEv [QtCore]	_ZNK6QRectF8containsERK7QPointF [QtCore]
ZNK6QRectF8containsERKS [QtCore]	_ZNK6QRectFanERKS_ [QtCore]
ZNK6QRectForERKS [QtCore]	_Zls6QDebugRK5QLine [QtCore]
_Zls6QDebugRK5QRect [QtCore]	_Zls6QDebugRK5QSize [QtCore]
_Zls6QDebugRK6QLineF [QtCore]	_Zls6QDebugRK6QPoint [QtCore]
_Zls6QDebugRK6QRectF [QtCore]	_Zls6QDebugRK6QSizeF [QtCore]
_Zls6QDebugRK7QPointF [QtCore]	_ZlsR11QDataStreamRK5QLine [QtCore]
_ZlsR11QDataStreamRK5QRect [QtCore]	_ZlsR11QDataStreamRK5QSize [QtCore]
_ZlsR11QDataStreamRK6QLineF [QtCore]	_ZlsR11QDataStreamRK6QPoint [QtCore]
_ZlsR11QDataStreamRK6QRectF [QtCore]	_ZlsR11QDataStreamRK6QSizeF [QtCore]
_ZlsR11QDataStreamRK7QPointF [QtCore]	_ZrsR11QDataStreamR5QLine [QtCore]
_ZrsR11QDataStreamR5QRect [QtCore]	_ZrsR11QDataStreamR5QSize [QtCore]
_ZrsR11QDataStreamR6QLineF [QtCore]	_ZrsR11QDataStreamR6QPoint [QtCore]
_ZrsR11QDataStreamR6QRectF [QtCore]	_ZrsR11QDataStreamR6QSizeF [QtCore]
_ZrsR11QDataStreamR7QPointF [QtCore]	

18.2.16 Qt4 Internals

18.2.16.1 Interfaces for Qt4 Internals

An LSB conforming implementation shall provide the generic functions for Qt4 Internals specified in Table 18-97, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-97 libQtCore - Qt4 Internals Function Interfaces

_ZN9QInternal12callFunctionENS_16InternalFunctionEPPv [QtXml]	_ZN9QInternal16registerCallbackENS_8CallbackEPFbPPvE [QtXml]
_ZN9QInternal17activateCallbacksENS_8CallbackEPPv [QtXml]	_ZN9QInternal18unregisterCallbackENS_8CallbackEPFbPPvE [QtXml]
_ZN9QtPrivate16QStringList_joinEPK11QStringListRK7QString [QtCore]	_ZN9QtPrivate16QStringList_sortEP11QStringList [QtCore]
_ZN9QtPrivate18QStringList_filterEPK11QStringListRK7QRegExp [QtCore]	_ZN9QtPrivate18QStringList_filterEPK11QStringListRK7QStringN2Qt15CaseSensitivityE [QtCore]
_ZN9QtPrivate19QStringList_indexOfEPK11QStringListRK7QRegExp [QtCore]	_ZN9QtPrivate20QStringList_containsEPK11QStringListRK7QStringN2Qt15CaseSensitivityE [QtCore]
_ZN9QtPrivate23QStringList_lastIndexOfEPK11QStringListRK7QRegExp [QtCore]	_ZN9QtPrivate28QStringList_replaceInStringsEP11QStringListRK7QRegExpRK7QString [QtCore]
_ZN9QtPrivate28QStringList_replaceInStringsEP11QStringListRK7QStringS4_N2Qt15CaseSensitivityE [QtCore]	

18.3 Data Definitions for libQtCore

This section defines global identifiers and their values that are associated with interfaces contained in libQtCore. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

18.3.1 QtCore/qabstracteventdispatcher.h

```
class QAbstractEventDispatcher;
typedef struct QPair<int, int> QAbstractEventDispatcher::TimerInfo;
```

```
typedef bool QAbstractEventDispatcher::EventFilter;
```

18.3.2 QtCore/qabstractfileengine.h

```
class QAbstractFileEngine;
enum QAbstractFileEngine::FileFlag {
    ExeOtherPerm = 1,
    WriteOtherPerm = 2,
    ReadOtherPerm = 4,
    ExeGroupPerm = 16,
    WriteGroupPerm = 32,
    ReadGroupPerm = 64,
    ExeUserPerm = 256,
    WriteUserPerm = 512,
    ReadUserPerm = 1024,
    ExeOwnerPerm = 4096,
    WriteOwnerPerm = 8192,
    ReadOwnerPerm = 16384,
    PermsMask = 65535,
    LinkType = 65536,
    FileType = 131072,
    DirectoryType = 262144,
    TypesMask = 983040,
    HiddenFlag = 1048576,
    LocalDiskFlag = 2097152,
    ExistsFlag = 4194304,
    RootFlag = 8388608,
    FlagsMask = 267386880,
    FileInfoAll = 268435455
};
class QFlags < QAbstractFileEngine::FileFlag >;
typedef class QFlags < QAbstractFileEngine::FileFlag >
    QAbstractFileEngine::FileFlags;
enum QAbstractFileEngine::FileName {
    DefaultName = 0,
    BaseName = 1,
    PathName = 2,
    AbsoluteName = 3,
    AbsolutePathName = 4,
    LinkName = 5,
    CanonicalName = 6,
    CanonicalPathName = 7
};
enum QAbstractFileEngine::FileOwner {
    OwnerUser = 0,
    OwnerGroup = 1
};
enum QAbstractFileEngine::FileTime {
    CreationTime = 0,
    ModificationTime = 1,
    AccessTime = 2
};
enum QAbstractFileEngine::Extension;
class QAbstractFileEngine::ExtensionOption;
class QAbstractFileEngine::ExtensionReturn;
class QAbstractFileEngineHandler;
```

18.3.3 QtCore/qabstractitemmodel.h

```
class QModelIndex;
class QPersistentModelIndex;
typedef class QList < QModelIndex > QModelIndexList;
class QAbstractItemModel;
```

```
class QAbstractTableModel;
class QAbstractListModel;
```

18.3.4 QtCore/qatomic.h

```
struct QBasicAtomic;
class QAtomic;
```

18.3.5 QtCore/qbasictimer.h

```
class QBasicTimer;
```

18.3.6 QtCore/qbitarray.h

```
class QBitArray;
class QBitRef;
```

18.3.7 QtCore/qbuffer.h

```
class QBuffer;
```

18.3.8 QtCore/qbytearray.h

```
class QByteArray;
typedef char QByteArray::iterator;
typedef char QByteArray::const_iterator;
typedef QByteArray::iterator QByteArray::Iterator;
typedef QByteArray::const_iterator QByteArray::ConstIterator;
typedef char QByteArray::const_reference;
typedef char QByteArray::reference;
struct QByteArray::Data;
class QByteRef;
```

18.3.9 QtCore/qbytearraymatcher.h

```
class QByteArrayMatcher;
```

18.3.10 QtCore/qchar.h

```
struct QLatin1Char;
class QChar;
enum QChar::SpecialCharacter {
    Null = 0,
    null = 0,
    Nbsp = 160,
    nbsp = 160,
    LineSeparator = 8232,
    ParagraphSeparator = 8233,
    ByteOrderMark = 65279,
    byteOrderMark = 65279,
    ObjectReplacementCharacter = 65532,
    replacement = 65533,
    ReplacementCharacter = 65533,
    byteOrderSwapped = 65534,
    ByteOrderSwapped = 65534
};
```



```

enum QChar::Category {
    NoCategory = 0,
    Mark_NonSpacing = 1,
    Mark_SpacingCombining = 2,
    Mark_Enclosing = 3,
    Number_DecimalDigit = 4,
    Number_Letter = 5,
    Number_Other = 6,
    Separator_Space = 7,
    Separator_Line = 8,
    Separator_Paragraph = 9,
    Other_Control = 10,
    Other_Format = 11,
    Other_Surrogate = 12,
    Other_PrivateUse = 13,
    Other_NotAssigned = 14,
    Letter_Uppercase = 15,
    Letter_Lowercase = 16,
    Letter_Titlecase = 17,
    Letter_Modifier = 18,
    Letter_Other = 19,
    Punctuation_Connector = 20,
    Punctuation_Dask = 21,
    Punctuation_Dash = 21,
    Punctuation_Open = 22,
    Punctuation_Close = 23,
    Punctuation_InitialQuote = 24,
    Punctuation_FinalQuote = 25,
    Punctuation_Other = 26,
    Symbol_Math = 27,
    Symbol_Currency = 28,
    Symbol_Modifier = 29,
    Symbol_Other = 30
};

enum QChar::Direction {
    DirL = 0,
    DirR = 1,
    DirEN = 2,
    DirES = 3,
    DirET = 4,
    DirAN = 5,
    DirCS = 6,
    DirB = 7,
    DirS = 8,
    DirWS = 9,
    DirON = 10,
    DirLRE = 11,
    DirLRO = 12,
    DirAL = 13,
    DirRLE = 14,
    DirRLO = 15,
    DirPDF = 16,
    DirNSM = 17,
    DirBN = 18
};

enum QChar::Decomposition {
    NoDecomposition = 0,
    Single = 0,
    Canonical = 1,
    Font = 2,
    NoBreak = 3,
    Initial = 4,
    Medial = 5,
    Final = 6,
    Isolated = 7,
    Circle = 8,

```

```

    Super = 9,
    Sub = 10,
    Vertical = 11,
    Wide = 12,
    Narrow = 13,
    Small = 14,
    Square = 15,
    Compat = 16,
    Fraction = 17
};
enum QChar::Joining {
    OtherJoining = 0,
    Dual = 1,
    Right = 2,
    Center = 3
};
enum QChar::CombiningClass {
    Combining_BelowLeftAttached = 200,
    Combining_BelowAttached = 202,
    Combining_BelowRightAttached = 204,
    Combining_LeftAttached = 208,
    Combining_RightAttached = 210,
    Combining_AboveLeftAttached = 212,
    Combining_AboveAttached = 214,
    Combining_AboveRightAttached = 216,
    Combining_BelowLeft = 218,
    Combining_Below = 220,
    Combining_BelowRight = 222,
    Combining_Left = 224,
    Combining_Right = 226,
    Combining_AboveLeft = 228,
    Combining_Above = 230,
    Combining_AboveRight = 232,
    Combining_DoubleBelow = 233,
    Combining_DoubleAbove = 234,
    Combining_IotaSubscript = 240
};
enum QChar::UnicodeVersion {
    Unicode_Unassigned = 0,
    Unicode_1_1 = 1,
    Unicode_2_0 = 2,
    Unicode_2_1_2 = 3,
    Unicode_3_0 = 4,
    Unicode_3_1 = 5,
    Unicode_3_2 = 6,
    Unicode_4_0 = 7
};

```

18.3.11 QtCore/qcoreapplication.h

```

class QCoreApplication;
enum QCoreApplication::Encoding {
    DefaultCodec = 0,
    UnicodeUTF8 = 1
};
typedef bool QCoreApplication::EventFilter;
typedef void QtCleanupFunction;

```

18.3.12 QtCore/qcoreevent.h

```

class QEvent;
enum QEvent::Type {
    None = 0,

```

Timer = 1,
 MouseButtonPress = 2,
 MouseButtonRelease = 3,
 MouseButtonDblClick = 4,
 MouseMove = 5,
 KeyPress = 6,
 KeyRelease = 7,
 FocusIn = 8,
 FocusOut = 9,
 Enter = 10,
 Leave = 11,
 Paint = 12,
 Move = 13,
 Resize = 14,
 Create = 15,
 Destroy = 16,
 Show = 17,
 Hide = 18,
 Close = 19,
 Quit = 20,
 ParentChange = 21,
 Reparent = 21,
 ThreadChange = 22,
 WindowActivate = 24,
 WindowDeactivate = 25,
 ShowToParent = 26,
 HideToParent = 27,
 Accel = 30,
 Wheel = 31,
 AccelAvailable = 32,
 WindowTitleChange = 33,
 CaptionChange = 33,
 WindowIconChange = 34,
 IconChange = 34,
 ApplicationWindowIconChange = 35,
 ApplicationFontChange = 36,
 ApplicationLayoutDirectionChange = 37,
 ApplicationPaletteChange = 38,
 PaletteChange = 39,
 Clipboard = 40,
 Speech = 42,
 MetaCall = 43,
 SockAct = 50,
 AccelOverride = 51,
 ShortcutOverride = 51,
 DeferredDelete = 52,
 DragEnter = 60,
 DragMove = 61,
 DragLeave = 62,
 Drop = 63,
 DragResponse = 64,
 ChildAdded = 68,
 ChildPolished = 69,
 ChildInserted = 70,
 ChildRemoved = 71,
 LayoutHint = 72,
 ShowWindowRequest = 73,
 PolishRequest = 74,
 Polish = 75,
 LayoutRequest = 76,
 UpdateRequest = 77,
 UpdateLater = 78,
 EmbeddingControl = 79,
 ActivateControl = 80,
 DeactivateControl = 81,
 ContextMenu = 82,

```

    InputMethod = 83,
    AccessibilityPrepare = 86,
    TabletMove = 87,
    LocaleChange = 88,
    LanguageChange = 89,
    LayoutDirectionChange = 90,
    Style = 91,
    TabletPress = 92,
    TabletRelease = 93,
    OkRequest = 94,
    HelpRequest = 95,
    IconDrag = 96,
    FontChange = 97,
    EnabledChange = 98,
    ActivationChange = 99,
    StyleChange = 100,
    IconTextChange = 101,
    ModifiedChange = 102,
    WindowBlocked = 103,
    WindowUnblocked = 104,
    WindowStateChange = 105,
    MouseTrackingChange = 109,
    ToolTip = 110,
    WhatsThis = 111,
    StatusTip = 112,
    ActionChanged = 113,
    ActionAdded = 114,
    ActionRemoved = 115,
    FileOpen = 116,
    Shortcut = 117,
    WhatsThisClicked = 118,
    AccessibilityHelp = 119,
    ToolBarChange = 120,
    ApplicationActivated = 121,
    ApplicationDeactivated = 122,
    QueryWhatsThis = 123,
    EnterWhatsThisMode = 124,
    LeaveWhatsThisMode = 125,
    ZOrderChange = 126,
    HoverEnter = 127,
    HoverLeave = 128,
    HoverMove = 129,
    AccessibilityDescription = 130,
    ParentAboutToChange = 131,
    WinEventAct = 132,
    AcceptDropsChange = 152,
    MenuBarUpdated = 153,
    ZeroTimerEvent = 154,
    User = 1000,
    MaxUser = 65535
};
class QTimerEvent;
class QChildEvent;
class QCustomEvent;

```

18.3.13 QtCore/qdatastream.h

```

class QDataStream;
enum QDataStream::Version {
    Qt_1_0 = 1,
    Qt_2_0 = 2,
    Qt_2_1 = 3,
    Qt_3_0 = 4,
    Qt_3_1 = 5,
    Qt_3_3 = 6,

```

```

    Qt_4_0 = 7,
    Qt_4_1 = 7
};
enum QDataStream::ByteOrder {
    BigEndian = 0,
    LittleEndian = 1
};
enum QDataStream::Status {
    Ok = 0,
    ReadPastEnd = 1,
    ReadCorruptData = 2
};

```

18.3.14 QtCore/qdatettime.h

```

class QDate;
class QTime;
class QDateTime;

```

18.3.15 QtCore/qdebug.h

```

class QDebug;
struct QDebug::Stream;

```

18.3.16 QtCore/qdir.h

```

class QDir;
enum QDir::Filter {
    DefaultFilter = -1,
    NoFilter = -1,
    Dirs = 1,
    Files = 2,
    Drives = 4,
    AllEntries = 7,
    All = 7,
    NoSymLinks = 8,
    TypeMask = 15,
    Readable = 16,
    Writable = 32,
    Executable = 64,
    PermissionMask = 112,
    RWEMask = 112,
    Modified = 128,
    Hidden = 256,
    System = 512,
    AccessMask = 1008,
    AllDirs = 1024,
    CaseSensitive = 2048,
    NoDotAndDotDot = 4096
};
class QFlags < QDir::Filter >;
typedef class QFlags < QDir::Filter > QDir::Filters;
typedef QDir::Filters QDir::FilterSpec;
enum QDir::SortFlag {
    DefaultSort = -1,
    NoSort = -1,
    Name = 0,
    Time = 1,
    Size = 2,
    Unsorted = 3,
    SortByMask = 3,
    DirsFirst = 4,

```

```

    Reversed = 8,
    IgnoreCase = 16,
    DirsLast = 32,
    LocaleAware = 64,
    Type = 128
};
class QFlags < QDir::SortFlag >;
typedef class QFlags < QDir::SortFlag > QDir::SortFlags;
typedef QDir::SortFlags QDir::SortSpec;

```

18.3.17 QtCore/qeventloop.h

```

class QEventLoop;
enum QEventLoop::ProcessEventsFlag {
    AllEvents = 0,
    ExcludeUserInputEvents = 1,
    ExcludeUserInput = 1,
    ExcludeSocketNotifiers = 2,
    WaitForMoreEvents = 4,
    WaitForMore = 4,
    X11ExcludeTimers = 8,
    DeferredDeletion = 16
};
class QFlags < QEventLoop::ProcessEventsFlag >;
typedef class QFlags < QEventLoop::ProcessEventsFlag >
    QEventLoop::ProcessEventsFlags;

```

18.3.18 QtCore/qfactoryinterface.h

```

struct QFactoryInterface;

```

18.3.19 QtCore/qfile.h

```

class QFile;
enum QFile::FileError {
    NoError = 0,
    ReadError = 1,
    WriteError = 2,
    FatalError = 3,
    ResourceError = 4,
    OpenError = 5,
    AbortError = 6,
    TimeoutError = 7,
    UnspecifiedError = 8,
    RemoveError = 9,
    RenameError = 10,
    PositionError = 11,
    ResizeError = 12,
    PermissionsError = 13,
    CopyError = 14,
    ConnectError = 30
};
enum QFile::Permission {
    ExeOther = 1,
    WriteOther = 2,
    ReadOther = 4,
    ExeGroup = 16,
    WriteGroup = 32,
    ReadGroup = 64,
    ExeUser = 256,
    WriteUser = 512,
    ReadUser = 1024,

```

```

    ExeOwner = 4096,
    WriteOwner = 8192,
    ReadOwner = 16384
};
class QFlags < QFile::Permission >;
typedef class QFlags < QFile::Permission > QFile::Permissions;
typedef class QByteArrayQFile::EncoderFn;
typedef class QStringQFile::DecoderFn;
typedef enum QFile::Permission {
    ExeOther = 1,
    WriteOther = 2,
    ReadOther = 4,
    ExeGroup = 16,
    WriteGroup = 32,
    ReadGroup = 64,
    ExeUser = 256,
    WriteUser = 512,
    ReadUser = 1024,
    ExeOwner = 4096,
    WriteOwner = 8192,
    ReadOwner = 16384
} QFile::PermissionSpec;

```

18.3.20 QtCore/qfileinfo.h

```

class QFileInfo;
enum QFileInfo::Permission {
    ExeOther = 1,
    WriteOther = 2,
    ReadOther = 4,
    ExeGroup = 16,
    WriteGroup = 32,
    ReadGroup = 64,
    ExeUser = 256,
    WriteUser = 512,
    ReadUser = 1024,
    ExeOwner = 4096,
    WriteOwner = 8192,
    ReadOwner = 16384
};
class QFlags < QFileInfo::Permission >;
typedef class QFlags < QFileInfo::Permission >
QFileInfo::PermissionSpec;
typedef class QList < QFileInfo > QFileInfoList;

```

18.3.21 QtCore/qfsfileengine.h

```

class QFSFileEngine;

```

18.3.22 QtCore/qglobal.h

```

typedef signed char quint8;
typedef unsigned char quint8;
typedef short quint16;
typedef unsigned short quint16;
typedef int quint32;
typedef unsigned int quint32;
typedef long long int quint64;
typedef unsigned long long int quint64;
typedef quint64 qlonglong;
typedef quint64 qulonglong;
typedef unsigned char uchar;

```

```

typedef int QNoImplicitBoolCast;
typedef double qreal;
typedef quint8 Q_INT8;
typedef quint8 Q_UINT8;
typedef quint16 Q_INT16;
typedef quint16 Q_UINT16;
typedef quint32 Q_INT32;
typedef quint32 Q_UINT32;
typedef quint64 Q_INT64;
typedef quint64 Q_UINT64;
typedef quint64 Q_LONG;
typedef quint64 Q_ULONG;
typedef long int Q_LONG;
typedef unsigned long int Q_ULONG;
class QSysInfo;
enum QSysInfo::Endian {
    BigEndian = 0,
    LittleEndian = 1,
    ByteOrder = 1
};
enum QtMsgType {
    QtDebugMsg = 0,
    QtWarningMsg = 1,
    QtCriticalMsg = 2,
    QtSystemMsg = 2,
    QtFatalMsg = 3
};
typedef void QtMsgHandler;
class QBool;
class QFlag;
enum QtValidLicenseForCoreModule {
    LicensedCore = 1
};
enum QtValidLicenseForGuiModule {
    LicensedGui = 1
};
enum QtValidLicenseForNetworkModule {
    LicensedNetwork = 1
};
enum QtValidLicenseForOpenGLModule {
    LicensedOpenGL = 1
};
enum QtValidLicenseForSqlModule {
    LicensedSql = 1
};
enum QtValidLicenseForXmlModule {
    LicensedXml = 1
};
enum QtValidLicenseForQt3SupportLightModule {
    LicensedQt3SupportLight = 1
};
enum QtValidLicenseForQt3SupportModule {
    LicensedQt3Support = 1
};
enum QtValidLicenseForSvgModule {
    LicensedSvg = 1
};
enum QtValidLicenseForActiveQtModule {
    LicensedActiveQt = 1
};
};

```

18.3.23 QtCore/qhash.h

```

struct QHashData;
struct QHashData::Node;

```



```
struct QHashDummyValue;
```

18.3.24 QtCore/qiodevice.h

```
class QIODevice;
enum QIODevice::OpenModeFlag {
    NotOpen = 0,
    ReadOnly = 1,
    WriteOnly = 2,
    ReadWrite = 3,
    Append = 4,
    Truncate = 8,
    Text = 16,
    Unbuffered = 32
};
class QFlags < QIODevice::OpenModeFlag >;
typedef class QFlags < QIODevice::OpenModeFlag >
QIODevice::OpenMode;
typedef quint64 QIODevice::Offset;
typedef int QIODevice::Status;
```

18.3.25 QtCore/qlibrary.h

```
class QLibrary;
```

18.3.26 QtCore/qlibraryinfo.h

```
class QLibraryInfo;
enum QLibraryInfo::LibraryLocation {
    PrefixPath = 0,
    DocumentationPath = 1,
    HeadersPath = 2,
    LibrariesPath = 3,
    BinariesPath = 4,
    PluginsPath = 5,
    DataPath = 6,
    TranslationsPath = 7,
    SettingsPath = 8,
    DemosPath = 9,
    ExamplesPath = 10
};
```

18.3.27 QtCore/qline.h

```
class QLine;
class QLineF;
enum QLineF::IntersectType {
    NoIntersection = 0,
    BoundedIntersection = 1,
    UnboundedIntersection = 2
};
```

18.3.28 QtCore/qlinkedlist.h

```
struct QLinkedListData;
```

18.3.29 QtCore/qlist.h

```

class QList < QByteArray >;
class QList < QFileInfo >;
class QList < QVariant >;
class QList < QUrl >;
class QList < QModelIndex >;
class QList < QHostAddress >;
class QList < int >;
class QList < QObject * >;
class QList < QWidget * >;
class QList < QPair < QString, QString > >;
struct QListData;
struct QListData::Data;

```

18.3.30 QtCore/qlocale.h

```

class QLocale;
enum QLocale::Language {
    C = 1,
    Abkhazian = 2,
    Afan = 3,
    Afar = 4,
    Afrikaans = 5,
    Albanian = 6,
    Amharic = 7,
    Arabic = 8,
    Armenian = 9,
    Assamese = 10,
    Aymara = 11,
    Azerbaijani = 12,
    Bashkir = 13,
    Basque = 14,
    Bengali = 15,
    Bhutani = 16,
    Bihari = 17,
    Bislama = 18,
    Breton = 19,
    Bulgarian = 20,
    Burmese = 21,
    Byelorussian = 22,
    Cambodian = 23,
    Catalan = 24,
    Chinese = 25,
    Corsican = 26,
    Croatian = 27,
    Czech = 28,
    Danish = 29,
    Dutch = 30,
    English = 31,
    Esperanto = 32,
    Estonian = 33,
    Faroese = 34,
    FijiLanguage = 35,
    Finnish = 36,
    French = 37,
    Frisian = 38,
    Gaelic = 39,
    Galician = 40,
    Georgian = 41,
    German = 42,
    Greek = 43,
    Greenlandic = 44,
    Guarani = 45,
    Gujarati = 46,
    Hausa = 47,
    Hebrew = 48,

```

Hindi = 49,
 Hungarian = 50,
 Icelandic = 51,
 Indonesian = 52,
 Interlingua = 53,
 Interlingue = 54,
 Inuktitut = 55,
 Inupiak = 56,
 Irish = 57,
 Italian = 58,
 Japanese = 59,
 Javanese = 60,
 Kannada = 61,
 Kashmiri = 62,
 Kazakh = 63,
 Kinyarwanda = 64,
 Kirghiz = 65,
 Korean = 66,
 Kurdish = 67,
 Kurundi = 68,
 Laothian = 69,
 Latin = 70,
 Latvian = 71,
 Lingala = 72,
 Lithuanian = 73,
 Macedonian = 74,
 Malagasy = 75,
 Malay = 76,
 Malayalam = 77,
 Maltese = 78,
 Maori = 79,
 Marathi = 80,
 Moldavian = 81,
 Mongolian = 82,
 NauruLanguage = 83,
 Nepali = 84,
 Norwegian = 85,
 Occitan = 86,
 Oriya = 87,
 Pashto = 88,
 Persian = 89,
 Polish = 90,
 Portuguese = 91,
 Punjabi = 92,
 Quechua = 93,
 RhaetoRomance = 94,
 Romanian = 95,
 Russian = 96,
 Samoan = 97,
 Sangho = 98,
 Sanskrit = 99,
 Serbian = 100,
 SerboCroatian = 101,
 Sesotho = 102,
 Setswana = 103,
 Shona = 104,
 Sindhi = 105,
 Singhalese = 106,
 Siswati = 107,
 Slovak = 108,
 Slovenian = 109,
 Somali = 110,
 Spanish = 111,
 Sundanese = 112,
 Swahili = 113,
 Swedish = 114,

```

Tagalog = 115,
Tajik = 116,
Tamil = 117,
Tatar = 118,
Telugu = 119,
Thai = 120,
Tibetan = 121,
Tigrinya = 122,
TongaLanguage = 123,
Tsonga = 124,
Turkish = 125,
Turkmen = 126,
Twi = 127,
Uigur = 128,
Ukrainian = 129,
Urdu = 130,
Uzbek = 131,
Vietnamese = 132,
Volapuk = 133,
Welsh = 134,
Wolof = 135,
Xhosa = 136,
Yiddish = 137,
Yoruba = 138,
Zhuang = 139,
Zulu = 140,
Nynorsk = 141,
Bosnian = 142,
Divehi = 143,
Manx = 144,
Cornish = 145,
LastLanguage = 145
};
enum QLocale::Country {
    AnyCountry = 0,
    Afghanistan = 1,
    Albania = 2,
    Algeria = 3,
    AmericanSamoa = 4,
    Andorra = 5,
    Angola = 6,
    Anguilla = 7,
    Antarctica = 8,
    AntiguaAndBarbuda = 9,
    Argentina = 10,
    Armenia = 11,
    Aruba = 12,
    Australia = 13,
    Austria = 14,
    Azerbaijan = 15,
    Bahamas = 16,
    Bahrain = 17,
    Bangladesh = 18,
    Barbados = 19,
    Belarus = 20,
    Belgium = 21,
    Belize = 22,
    Benin = 23,
    Bermuda = 24,
    Bhutan = 25,
    Bolivia = 26,
    BosniaAndHerzegovina = 27,
    Botswana = 28,
    BouvetIsland = 29,
    Brazil = 30,
    BritishIndianOceanTerritory = 31,

```

BruneiDarussalam = 32,
 Bulgaria = 33,
 BurkinaFaso = 34,
 Burundi = 35,
 Cambodia = 36,
 Cameroon = 37,
 Canada = 38,
 CapeVerde = 39,
 CaymanIslands = 40,
 CentralAfricanRepublic = 41,
 Chad = 42,
 Chile = 43,
 China = 44,
 ChristmasIsland = 45,
 CocosIslands = 46,
 Colombia = 47,
 Comoros = 48,
 DemocraticRepublicOfCongo = 49,
 PeoplesRepublicOfCongo = 50,
 CookIslands = 51,
 CostaRica = 52,
 IvoryCoast = 53,
 Croatia = 54,
 Cuba = 55,
 Cyprus = 56,
 CzechRepublic = 57,
 Denmark = 58,
 Djibouti = 59,
 Dominica = 60,
 DominicanRepublic = 61,
 EastTimor = 62,
 Ecuador = 63,
 Egypt = 64,
 ElSalvador = 65,
 EquatorialGuinea = 66,
 Eritrea = 67,
 Estonia = 68,
 Ethiopia = 69,
 FalklandIslands = 70,
 FaroeIslands = 71,
 FijiCountry = 72,
 Finland = 73,
 France = 74,
 MetropolitanFrance = 75,
 FrenchGuiana = 76,
 FrenchPolynesia = 77,
 FrenchSouthernTerritories = 78,
 Gabon = 79,
 Gambia = 80,
 Georgia = 81,
 Germany = 82,
 Ghana = 83,
 Gibraltar = 84,
 Greece = 85,
 Greenland = 86,
 Grenada = 87,
 Guadeloupe = 88,
 Guam = 89,
 Guatemala = 90,
 Guinea = 91,
 GuineaBissau = 92,
 Guyana = 93,
 Haiti = 94,
 HeardAndMcDonaldIslands = 95,
 Honduras = 96,
 HongKong = 97,

Hungary = 98,
 Iceland = 99,
 India = 100,
 Indonesia = 101,
 Iran = 102,
 Iraq = 103,
 Ireland = 104,
 Israel = 105,
 Italy = 106,
 Jamaica = 107,
 Japan = 108,
 Jordan = 109,
 Kazakhstan = 110,
 Kenya = 111,
 Kiribati = 112,
 DemocraticRepublicOfKorea = 113,
 RepublicOfKorea = 114,
 Kuwait = 115,
 Kyrgyzstan = 116,
 Lao = 117,
 Latvia = 118,
 Lebanon = 119,
 Lesotho = 120,
 Liberia = 121,
 LibyanArabJamahiriya = 122,
 Liechtenstein = 123,
 Lithuania = 124,
 Luxembourg = 125,
 Macau = 126,
 Macedonia = 127,
 Madagascar = 128,
 Malawi = 129,
 Malaysia = 130,
 Maldives = 131,
 Mali = 132,
 Malta = 133,
 MarshallIslands = 134,
 Martinique = 135,
 Mauritania = 136,
 Mauritius = 137,
 Mayotte = 138,
 Mexico = 139,
 Micronesia = 140,
 Moldova = 141,
 Monaco = 142,
 Mongolia = 143,
 Montserrat = 144,
 Morocco = 145,
 Mozambique = 146,
 Myanmar = 147,
 Namibia = 148,
 NauruCountry = 149,
 Nepal = 150,
 Netherlands = 151,
 NetherlandsAntilles = 152,
 NewCaledonia = 153,
 NewZealand = 154,
 Nicaragua = 155,
 Niger = 156,
 Nigeria = 157,
 Niue = 158,
 NorfolkIsland = 159,
 NorthernMarianaIslands = 160,
 Norway = 161,
 Oman = 162,
 Pakistan = 163,

Palau = 164,
 PalestinianTerritory = 165,
 Panama = 166,
 PapuaNewGuinea = 167,
 Paraguay = 168,
 Peru = 169,
 Philippines = 170,
 Pitcairn = 171,
 Poland = 172,
 Portugal = 173,
 PuertoRico = 174,
 Qatar = 175,
 Reunion = 176,
 Romania = 177,
 RussianFederation = 178,
 Rwanda = 179,
 SaintKittsAndNevis = 180,
 StLucia = 181,
 StVincentAndTheGrenadines = 182,
 Samoa = 183,
 SanMarino = 184,
 SaoTomeAndPrincipe = 185,
 SaudiArabia = 186,
 Senegal = 187,
 Seychelles = 188,
 SierraLeone = 189,
 Singapore = 190,
 Slovakia = 191,
 Slovenia = 192,
 SolomonIslands = 193,
 Somalia = 194,
 SouthAfrica = 195,
 SouthGeorgiaAndTheSouthSandwichIslands = 196,
 Spain = 197,
 SriLanka = 198,
 StHelena = 199,
 StPierreAndMiquelon = 200,
 Sudan = 201,
 Suriname = 202,
 SvalbardAndJanMayenIslands = 203,
 Swaziland = 204,
 Sweden = 205,
 Switzerland = 206,
 SyrianArabRepublic = 207,
 Taiwan = 208,
 Tajikistan = 209,
 Tanzania = 210,
 Thailand = 211,
 Togo = 212,
 Tokelau = 213,
 TongaCountry = 214,
 TrinidadAndTobago = 215,
 Tunisia = 216,
 Turkey = 217,
 Turkmenistan = 218,
 TurksAndCaicosIslands = 219,
 Tuvalu = 220,
 Uganda = 221,
 Ukraine = 222,
 UnitedArabEmirates = 223,
 UnitedKingdom = 224,
 UnitedStates = 225,
 UnitedStatesMinorOutlyingIslands = 226,
 Uruguay = 227,
 Uzbekistan = 228,
 Vanuatu = 229,

```

    VaticanCityState = 230,
    Venezuela = 231,
    VietNam = 232,
    BritishVirginIslands = 233,
    USVirginIslands = 234,
    WallisAndFutunaIslands = 235,
    WesternSahara = 236,
    Yemen = 237,
    Yugoslavia = 238,
    Zambia = 239,
    Zimbabwe = 240,
    SerbiaAndMontenegro = 241,
    LastCountry = 241
};
enum QLocale::FormatType {
    LongFormat = 0,
    ShortFormat = 1
};

```

18.3.31 QtCore/qmap.h

```

class QMap < int, QVariant >;
class QMap < QString, QVariant >;
struct QMapData;
struct QMapData::Node;

```

18.3.32 QtCore/qmetaobject.h

```

class QMetaMethod;
enum QMetaMethod::Access {
    Private = 0,
    Protected = 1,
    Public = 2
};
enum QMetaMethod::MethodType {
    Method = 0,
    Signal = 1,
    Slot = 2
};
enum QMetaMethod::Attributes {
    Compatibility = 1,
    Cloned = 2,
    Scriptable = 4
};
class QMetaEnum;
class QMetaProperty;
class QMetaClassInfo;

```

18.3.33 QtCore/qmetatype.h

```

class QMetaType;
enum QMetaType::Type {
    Void = 0,
    Bool = 1,
    Int = 2,
    UInt = 3,
    Double = 6,
    QChar = 7,
    QString = 10,
    QByteArray = 12,
    VoidStar = 128,
    Long = 129,

```



```

    Short = 130,
    Char = 131,
    ULong = 132,
    UShort = 133,
    UChar = 134,
    Float = 135,
    QObjectStar = 136,
    QWidgetStar = 137,
    User = 256
};
typedef void QMetaType::Destructor;
typedef void QMetaType::Constructor;
typedef void QMetaType::SaveOperator;
typedef void QMetaType::LoadOperator;

```

18.3.34 QtCore/qmimedata.h

```
class QMimeData;
```

18.3.35 QtCore/qmutex.h

```

class QMutex;
enum QMutex::RecursionMode {
    NonRecursive = 0,
    Recursive = 1
};
class QMutexLocker;

```

18.3.36 QtCore/qnamespace.h

```

enum Qt::GlobalColor {
    color0 = 0,
    color1 = 1,
    black = 2,
    white = 3,
    darkGray = 4,
    gray = 5,
    lightGray = 6,
    red = 7,
    green = 8,
    blue = 9,
    cyan = 10,
    magenta = 11,
    yellow = 12,
    darkRed = 13,
    darkGreen = 14,
    darkBlue = 15,
    darkCyan = 16,
    darkMagenta = 17,
    darkYellow = 18,
    transparent = 19
};
enum Qt::KeyboardModifier {
    KeyboardModifierMask = -33554432,
    NoModifier = 0,
    ShiftModifier = 33554432,
    ControlModifier = 67108864,
    AltModifier = 134217728,
    MetaModifier = 268435456,
    KeypadModifier = 536870912
};
typedef class QFlags < Qt::KeyboardModifier > Qt::KeyboardModifiers;

```

```

enum Qt::Modifier {
    MODIFIER_MASK = -33554432,
    UNICODE_ACCEL = 0,
    SHIFT = 33554432,
    CTRL = 67108864,
    ALT = 134217728,
    META = 268435456
};

enum Qt::MouseButton {
    NoButton = 0,
    LeftButton = 1,
    RightButton = 2,
    MidButton = 4,
    XButton1 = 8,
    XButton2 = 16,
    MouseButtonMask = 255
};

class QFlags < Qt::MouseButton >;
typedef class QFlags < Qt::MouseButton > Qt::MouseButtons;
enum Qt::ButtonState_enum {
    KeyButtonMask = -33554432,
    ShiftButton = 33554432,
    ControlButton = 67108864,
    AltButton = 134217728,
    MetaButton = 268435456,
    Keypad = 536870912
};

typedef int Qt::ButtonState;
enum Qt::Orientation {
    Horizontal = 1,
    Vertical = 2
};

class QFlags < Qt::Orientation >;
typedef class QFlags < Qt::Orientation > Qt::Orientations;
enum Qt::FocusPolicy {
    NoFocus = 0,
    TabFocus = 1,
    ClickFocus = 2,
    StrongFocus = 11,
    WheelFocus = 15
};

enum Qt::SortOrder {
    AscendingOrder = 0,
    Ascending = 0,
    DescendingOrder = 1,
    Descending = 1
};

enum Qt::AlignmentFlag {
    AlignLeft = 1,
    AlignAuto = 1,
    AlignLeading = 1,
    AlignRight = 2,
    AlignTrailing = 2,
    AlignHCenter = 4,
    AlignJustify = 8,
    AlignAbsolute = 16,
    AlignHorizontal_Mask = 31,
    AlignTop = 32,
    AlignBottom = 64,
    AlignVCenter = 128,
    AlignCenter = 132,
    AlignVertical_Mask = 224
};

class QFlags < Qt::AlignmentFlag >;
typedef class QFlags < Qt::AlignmentFlag > Qt::Alignment;
enum Qt::TextFlag {

```

```

TextSingleLine = 256,
SingleLine = 256,
DontClip = 512,
TextDontClip = 512,
ExpandTabs = 1024,
TextExpandTabs = 1024,
ShowPrefix = 2048,
TextShowMnemonic = 2048,
TextWordWrap = 4096,
WordBreak = 4096,
BreakAnywhere = 8192,
TextWrapAnywhere = 8192,
DontPrint = 16384,
TextDontPrint = 16384,
TextHideMnemonic = 32768,
NoAccel = 32768,
TextIncludeTrailingSpaces = 134217728,
IncludeTrailingSpaces = 134217728
};

typedef enum Qt::TextFlag {
    TextSingleLine = 256,
    SingleLine = 256,
    DontClip = 512,
    TextDontClip = 512,
    ExpandTabs = 1024,
    TextExpandTabs = 1024,
    ShowPrefix = 2048,
    TextShowMnemonic = 2048,
    TextWordWrap = 4096,
    WordBreak = 4096,
    BreakAnywhere = 8192,
    TextWrapAnywhere = 8192,
    DontPrint = 16384,
    TextDontPrint = 16384,
    TextHideMnemonic = 32768,
    NoAccel = 32768,
    TextIncludeTrailingSpaces = 134217728,
    IncludeTrailingSpaces = 134217728
} Qt::TextFlags;

enum Qt::TextElideMode {
    ElideLeft = 0,
    ElideRight = 1,
    ElideMiddle = 2
};

enum Qt::WindowType {
    Widget = 0,
    WStyle_Customize = 0,
    WStyle_NormalBorder = 0,
    WPaintDesktop = 0,
    WPaintClever = 0,
    WRepaintNoErase = 0,
    WResizeNoErase = 0,
    WMacNoSheet = 0,
    WNoAutoErase = 0,
    WType_TopLevel = 1,
    Window = 1,
    Dialog = 3,
    WType_Dialog = 3,
    WStyle_Dialog = 3,
    WMacSheet = 5,
    Sheet = 5,
    WMacDrawer = 7,
    Drawer = 7,
    Popup = 9,
    WType_Popup = 9,
    WStyle_Tool = 11,

```

```

Tool = 11,
ToolTip = 13,
SplashScreen = 15,
WStyle_Splash = 15,
Desktop = 17,
WType_Desktop = 17,
SubWindow = 18,
WType_Mask = 255,
WindowType_Mask = 255,
WStyle_DialogBorder = 256,
MSWindowsFixedSizeDialogHint = 256,
MSWindowsOwnDC = 512,
WWinOwnDC = 512,
X11BypassWindowManagerHint = 1024,
WX11BypassWM = 1024,
FramelessWindowHint = 2048,
WStyle_NoBorder = 2048,
WStyle_NoBorderEx = 2048,
WStyle_Title = 4096,
WindowTitleHint = 4096,
WindowSystemMenuHint = 8192,
WStyle_SysMenu = 8192,
WStyle_Minimize = 16384,
WindowMinimizeButtonHint = 16384,
WStyle_Maximize = 32768,
WindowMaximizeButtonHint = 32768,
WStyle_MinMax = 49152,
WindowMinMaxButtonsHint = 49152,
WStyle_ContextHelp = 65536,
WindowContextHelpButtonHint = 65536,
WindowShadeButtonHint = 131072,
WStyle_StaysOnTop = 262144,
WindowStaysOnTopHint = 262144,
WMouseNoMask = 524288,
WDestructiveClose = 1048576,
WNorthWestGravity = 2097152,
WStaticContents = 2097152,
WGroupLeader = 4194304,
WShowModal = 8388608,
WType_Modal = 8388611,
WNoMousePropagation = 16777216
};
class QFlags < Qt::WindowType >;
typedef class QFlags < Qt::WindowType > Qt::WindowFlags;
enum Qt::WindowState {
    WindowNoState = 0,
    WindowMinimized = 1,
    WindowMaximized = 2,
    WindowFullScreen = 4,
    WindowActive = 8
};
class QFlags < Qt::WindowState >;
typedef class QFlags < Qt::WindowState > Qt::WindowStates;
enum Qt::WidgetAttribute {
    WA_Disabled = 0,
    WA_UnderMouse = 1,
    WA_MouseTracking = 2,
    WA_ContentsPropagated = 3,
    WA_OpaquePaintEvent = 4,
    WA_NoBackground = 4,
    WA_StaticContents = 5,
    WA_LaidOut = 7,
    WA_PaintOnScreen = 8,
    WA_NoSystemBackground = 9,
    WA_UpdatesDisabled = 10,
    WA_Mapped = 11,

```

```

WA_MacNoClickThrough = 12,
WA_PaintOutsidePaintEvent = 13,
WA_InputMethodEnabled = 14,
WA_WState_Visible = 15,
WA_WState_Hidden = 16,
WA_ForceDisabled = 32,
WA_KeyCompression = 33,
WA_PendingMoveEvent = 34,
WA_PendingResizeEvent = 35,
WA_SetPalette = 36,
WA_SetFont = 37,
WA_SetCursor = 38,
WA_NoChildEventsFromChildren = 39,
WA_WindowModified = 41,
WA_Resize = 42,
WA_Moved = 43,
WA_PendingUpdate = 44,
WA_InvalidSize = 45,
WA_MacMetalStyle = 46,
WA_CustomWhatsThis = 47,
WA_LayoutOnEntireRect = 48,
WA_OutsideWSRange = 49,
WA_GrabbedShortcut = 50,
WA_TransparentForMouseEvents = 51,
WA_PaintUnclipped = 52,
WA_SetWindowIcon = 53,
WA_NoMouseReplay = 54,
WA_DeleteOnClose = 55,
WA_RightToLeft = 56,
WA_SetLayoutDirection = 57,
WA_NoChildEventsForParent = 58,
WA_ForceUpdatesDisabled = 59,
WA_WState_Created = 60,
WA_WState_CompressKeys = 61,
WA_WState_InPaintEvent = 62,
WA_WState_Reparented = 63,
WA_WState_ConfigPending = 64,
WA_WState_Polished = 66,
WA_WState_DND = 67,
WA_WState_OwnSizePolicy = 68,
WA_WState_ExplicitShowHide = 69,
WA_ShowModal = 70,
WA_MouseNoMask = 71,
WA_GroupLeader = 72,
WA_NoMousePropagation = 73,
WA_Hover = 74,
WA_InputMethodTransparent = 75,
WA_QuitOnClose = 76,
WA_KeyboardFocusChange = 77,
WA_AcceptDrops = 78,
WA_DropSiteRegistered = 79,
WA_ForceAcceptDrops = 79,
WA_WindowPropagation = 80,
WA_NoX11EventCompression = 81,
WA_TintedBackground = 82,
WA_AttributeCount = 83
};

enum Qt::ImageConversionFlag {
    AutoColor = 0,
    AutoDither = 0,
    ThresholdAlphaDither = 0,
    DiffuseDither = 0,
    MonoOnly = 2,
    ColorMode_Mask = 3,
    ColorOnly = 3,
    OrderedAlphaDither = 4,

```

```

        DiffuseAlphaDither = 8,
        NoAlpha = 12,
        AlphaDither_Mask = 12,
        OrderedDither = 16,
        ThresholdDither = 32,
        Dither_Mask = 48,
        PreferDither = 64,
        AvoidDither = 128,
        DitherMode_Mask = 192
    };
    typedef      class      QFlags      <      Qt::ImageConversionFlag      >
    Qt::ImageConversionFlags;
    enum Qt::BGMode {
        TransparentMode = 0,
        OpaqueMode = 1
    };
    enum Qt::PaintUnit {
        PixelUnit = 0,
        LoMetricUnit = 1,
        HiMetricUnit = 2,
        LoEnglishUnit = 3,
        HiEnglishUnit = 4,
        TwipsUnit = 5
    };
    enum Qt::GUIStyle {
        MacStyle = 0,
        WindowsStyle = 1,
        Win3Style = 2,
        PMStyle = 3,
        MotifStyle = 4
    };
    enum Qt::Key {
        Key_Space = 32,
        Key_Any = 32,
        Key_Exclam = 33,
        Key_QuoteDbl = 34,
        Key_NumberSign = 35,
        Key_Dollar = 36,
        Key_Percent = 37,
        Key_Ampersand = 38,
        Key_Apostrophe = 39,
        Key_ParenLeft = 40,
        Key_ParenRight = 41,
        Key_Asterisk = 42,
        Key_Plus = 43,
        Key_Comma = 44,
        Key_Minus = 45,
        Key_Period = 46,
        Key_Slash = 47,
        Key_0 = 48,
        Key_1 = 49,
        Key_2 = 50,
        Key_3 = 51,
        Key_4 = 52,
        Key_5 = 53,
        Key_6 = 54,
        Key_7 = 55,
        Key_8 = 56,
        Key_9 = 57,
        Key_Colon = 58,
        Key_Semicolon = 59,
        Key_Less = 60,
        Key_Equal = 61,
        Key_Greater = 62,
        Key_Question = 63,
        Key_At = 64,

```

Key_A = 65,
 Key_B = 66,
 Key_C = 67,
 Key_D = 68,
 Key_E = 69,
 Key_F = 70,
 Key_G = 71,
 Key_H = 72,
 Key_I = 73,
 Key_J = 74,
 Key_K = 75,
 Key_L = 76,
 Key_M = 77,
 Key_N = 78,
 Key_O = 79,
 Key_P = 80,
 Key_Q = 81,
 Key_R = 82,
 Key_S = 83,
 Key_T = 84,
 Key_U = 85,
 Key_V = 86,
 Key_W = 87,
 Key_X = 88,
 Key_Y = 89,
 Key_Z = 90,
 Key_BracketLeft = 91,
 Key_Backslash = 92,
 Key_BracketRight = 93,
 Key_AsciiCircum = 94,
 Key_Underscore = 95,
 Key_QuoteLeft = 96,
 Key_BraceLeft = 123,
 Key_Bar = 124,
 Key_BraceRight = 125,
 Key_AsciiTilde = 126,
 Key_nobreakspace = 160,
 Key_exclamdown = 161,
 Key_cent = 162,
 Key_sterling = 163,
 Key_currency = 164,
 Key_yen = 165,
 Key_brokenbar = 166,
 Key_section = 167,
 Key_diaeresis = 168,
 Key_copyright = 169,
 Key_ordfeminine = 170,
 Key_guillemotleft = 171,
 Key_notsign = 172,
 Key_hyphen = 173,
 Key_registered = 174,
 Key_macron = 175,
 Key_degree = 176,
 Key_plusminus = 177,
 Key_twosuperior = 178,
 Key_threesuperior = 179,
 Key_acute = 180,
 Key_mu = 181,
 Key_paragraph = 182,
 Key_periodcentered = 183,
 Key_cedilla = 184,
 Key_onesuperior = 185,
 Key_masculine = 186,
 Key_guillemotright = 187,
 Key_onequarter = 188,
 Key_onehalf = 189,

Key_threequarters = 190,
 Key_questiondown = 191,
 Key_agrave = 192,
 Key_Agrave = 192,
 Key_aacute = 193,
 Key_Aacute = 193,
 Key_Acircumflex = 194,
 Key_acircumflex = 194,
 Key_Atilde = 195,
 Key_atilde = 195,
 Key_Adiaeresis = 196,
 Key_adiaeresis = 196,
 Key_Aring = 197,
 Key_aring = 197,
 Key_AE = 198,
 Key_ae = 198,
 Key_Ccedilla = 199,
 Key_ccedilla = 199,
 Key_egrave = 200,
 Key_Egrave = 200,
 Key_Eacute = 201,
 Key_eacute = 201,
 Key_Ecircumflex = 202,
 Key_ecircumflex = 202,
 Key_Ediaeresis = 203,
 Key_ediaeresis = 203,
 Key_igrave = 204,
 Key_Igrave = 204,
 Key_Iacute = 205,
 Key_iacute = 205,
 Key_icircumflex = 206,
 Key_Icircumflex = 206,
 Key_idiaeresis = 207,
 Key_Idiaeresis = 207,
 Key_ETH = 208,
 Key_eth = 208,
 Key_Ntilde = 209,
 Key_ntilde = 209,
 Key_Ograve = 210,
 Key_ograve = 210,
 Key_Oacute = 211,
 Key_oacute = 211,
 Key_ocircumflex = 212,
 Key_Ocircumflex = 212,
 Key_otilde = 213,
 Key_otilde = 213,
 Key_odiaeresis = 214,
 Key_Odiaeresis = 214,
 Key_multiply = 215,
 Key_Ooblique = 216,
 Key_oslash = 216,
 Key_ugrave = 217,
 Key_Ugrave = 217,
 Key_uacute = 218,
 Key_Uacute = 218,
 Key_Ucircumflex = 219,
 Key_ucircumflex = 219,
 Key_Udiaeresis = 220,
 Key_udiaeresis = 220,
 Key_yacute = 221,
 Key_Yacute = 221,
 Key_THORN = 222,
 Key_thorn = 222,
 Key_ssharp = 223,
 Key_division = 247,
 Key_ydiaeresis = 255,

Key_Escape = 16777216,
 Key_Tab = 16777217,
 Key_Backtab = 16777218,
 Key_BackTab = 16777218,
 Key_BackSpace = 16777219,
 Key_Backspace = 16777219,
 Key_Return = 16777220,
 Key_Enter = 16777221,
 Key_Insert = 16777222,
 Key_Delete = 16777223,
 Key_Pause = 16777224,
 Key_Print = 16777225,
 Key_SysReq = 16777226,
 Key_Clear = 16777227,
 Key_Home = 16777232,
 Key_End = 16777233,
 Key_Left = 16777234,
 Key_Up = 16777235,
 Key_Right = 16777236,
 Key_Down = 16777237,
 Key_Prior = 16777238,
 Key_PageUp = 16777238,
 Key_PageDown = 16777239,
 Key_Next = 16777239,
 Key_Shift = 16777248,
 Key_Control = 16777249,
 Key_Meta = 16777250,
 Key_Alt = 16777251,
 Key_CapsLock = 16777252,
 Key_NumLock = 16777253,
 Key_ScrollLock = 16777254,
 Key_F1 = 16777264,
 Key_F2 = 16777265,
 Key_F3 = 16777266,
 Key_F4 = 16777267,
 Key_F5 = 16777268,
 Key_F6 = 16777269,
 Key_F7 = 16777270,
 Key_F8 = 16777271,
 Key_F9 = 16777272,
 Key_F10 = 16777273,
 Key_F11 = 16777274,
 Key_F12 = 16777275,
 Key_F13 = 16777276,
 Key_F14 = 16777277,
 Key_F15 = 16777278,
 Key_F16 = 16777279,
 Key_F17 = 16777280,
 Key_F18 = 16777281,
 Key_F19 = 16777282,
 Key_F20 = 16777283,
 Key_F21 = 16777284,
 Key_F22 = 16777285,
 Key_F23 = 16777286,
 Key_F24 = 16777287,
 Key_F25 = 16777288,
 Key_F26 = 16777289,
 Key_F27 = 16777290,
 Key_F28 = 16777291,
 Key_F29 = 16777292,
 Key_F30 = 16777293,
 Key_F31 = 16777294,
 Key_F32 = 16777295,
 Key_F33 = 16777296,
 Key_F34 = 16777297,
 Key_F35 = 16777298,

Key_Super_L = 16777299,
 Key_Super_R = 16777300,
 Key_Menu = 16777301,
 Key_Hyper_L = 16777302,
 Key_Hyper_R = 16777303,
 Key_Help = 16777304,
 Key_Direction_L = 16777305,
 Key_Direction_R = 16777312,
 Key_Back = 16777313,
 Key_Forward = 16777314,
 Key_Stop = 16777315,
 Key_Refresh = 16777316,
 Key_VolumeDown = 16777328,
 Key_VolumeMute = 16777329,
 Key_VolumeUp = 16777330,
 Key_BassBoost = 16777331,
 Key_BassUp = 16777332,
 Key_BassDown = 16777333,
 Key_TrebleUp = 16777334,
 Key_TrebleDown = 16777335,
 Key_MediaPlay = 16777344,
 Key_MediaStop = 16777345,
 Key_MediaPrevious = 16777346,
 Key_MediaPrev = 16777346,
 Key_MediaNext = 16777347,
 Key_MediaRecord = 16777348,
 Key_HomePage = 16777360,
 Key_Favorites = 16777361,
 Key_Search = 16777362,
 Key_Standby = 16777363,
 Key_OpenUrl = 16777364,
 Key_LaunchMail = 16777376,
 Key_LaunchMedia = 16777377,
 Key_Launch0 = 16777378,
 Key_Launch1 = 16777379,
 Key_Launch2 = 16777380,
 Key_Launch3 = 16777381,
 Key_Launch4 = 16777382,
 Key_Launch5 = 16777383,
 Key_Launch6 = 16777384,
 Key_Launch7 = 16777385,
 Key_Launch8 = 16777386,
 Key_Launch9 = 16777387,
 Key_LaunchA = 16777388,
 Key_LaunchB = 16777389,
 Key_LaunchC = 16777390,
 Key_LaunchD = 16777391,
 Key_LaunchE = 16777392,
 Key_LaunchF = 16777393,
 Key_AltGr = 16781571,
 Key_Multi_key = 16781600,
 Key_Kanji = 16781601,
 Key_Muhenkan = 16781602,
 Key_Henkan = 16781603,
 Key_Romaji = 16781604,
 Key_Hiragana = 16781605,
 Key_Katakana = 16781606,
 Key_Hiragana_Katakana = 16781607,
 Key_Zenkaku = 16781608,
 Key_Hankaku = 16781609,
 Key_Zenkaku_Hankaku = 16781610,
 Key_Touroku = 16781611,
 Key_Massyo = 16781612,
 Key_Kana_Lock = 16781613,
 Key_Kana_Shift = 16781614,
 Key_Eisu_Shift = 16781615,

```

Key_Eisu_toggle = 16781616,
Key_Hangul = 16781617,
Key_Hangul_Start = 16781618,
Key_Hangul_End = 16781619,
Key_Hangul_Hanja = 16781620,
Key_Hangul_Jamo = 16781621,
Key_Hangul_Romaja = 16781622,
Key_Codeinput = 16781623,
Key_Hangul_Jeonja = 16781624,
Key_Hangul_Banja = 16781625,
Key_Hangul_PreHanja = 16781626,
Key_Hangul_PostHanja = 16781627,
Key_SingleCandidate = 16781628,
Key_MultipleCandidate = 16781629,
Key_PreviousCandidate = 16781630,
Key_Hangul_Special = 16781631,
Key_Mode_switch = 16781694,
Key_Dead_Grave = 16781904,
Key_Dead_Acute = 16781905,
Key_Dead_Circumflex = 16781906,
Key_Dead_Tilde = 16781907,
Key_Dead_Macron = 16781908,
Key_Dead_Breve = 16781909,
Key_Dead_Abovedot = 16781910,
Key_Dead_Diaeresis = 16781911,
Key_Dead_Abovering = 16781912,
Key_Dead_Doubleacute = 16781913,
Key_Dead_Caron = 16781914,
Key_Dead_Cedilla = 16781915,
Key_Dead_Ogonek = 16781916,
Key_Dead_Iota = 16781917,
Key_Dead_Voiced_Sound = 16781918,
Key_Dead_Semivoiced_Sound = 16781919,
Key_Dead_Belowdot = 16781920,
Key_Dead_Hook = 16781921,
Key_Dead_Horn = 16781922,
Key_MediaLast = 16842751,
Key_Select = 16842752,
Key_Yes = 16842753,
Key_No = 16842754,
Key_Context1 = 17825792,
Key_Context2 = 17825793,
Key_Context3 = 17825794,
Key_Context4 = 17825795,
Key_Call = 17825796,
Key_Hangup = 17825797,
Key_Flip = 17825798,
Key_unknown = 33554431
};

enum Qt::ArrowType {
    NoArrow = 0,
    UpArrow = 1,
    DownArrow = 2,
    LeftArrow = 3,
    RightArrow = 4
};

enum Qt::PenStyle {
    NoPen = 0,
    SolidLine = 1,
    DashLine = 2,
    DotLine = 3,
    DashDotLine = 4,
    DashDotDotLine = 5,
    CustomDashLine = 6,
    MPenStyle = 15
};

```

```

enum Qt::PenCapStyle {
    FlatCap = 0,
    SquareCap = 16,
    RoundCap = 32,
    MPenCapStyle = 48
};
enum Qt::PenJoinStyle {
    MiterJoin = 0,
    BevelJoin = 64,
    RoundJoin = 128,
    MPenJoinStyle = 192
};
enum Qt::BrushStyle {
    NoBrush = 0,
    SolidPattern = 1,
    Dense1Pattern = 2,
    Dense2Pattern = 3,
    Dense3Pattern = 4,
    Dense4Pattern = 5,
    Dense5Pattern = 6,
    Dense6Pattern = 7,
    Dense7Pattern = 8,
    HorPattern = 9,
    VerPattern = 10,
    CrossPattern = 11,
    BDiagPattern = 12,
    FDiagPattern = 13,
    DiagCrossPattern = 14,
    LinearGradientPattern = 15,
    RadialGradientPattern = 16,
    ConicalGradientPattern = 17,
    TexturePattern = 24,
    CustomPattern = 24
};
enum Qt::UIEffect {
    UI_General = 0,
    UI_AnimateMenu = 1,
    UI_FadeMenu = 2,
    UI_AnimateCombo = 3,
    UI_AnimateTooltip = 4,
    UI_FadeTooltip = 5,
    UI_AnimateToolBox = 6
};
enum Qt::CursorShape {
    ArrowCursor = 0,
    arrowCursor = 0,
    UpArrowCursor = 1,
    upArrowCursor = 1,
    CrossCursor = 2,
    crossCursor = 2,
    waitCursor = 3,
    WaitCursor = 3,
    ibeamCursor = 4,
    IBeamCursor = 4,
    SizeVerCursor = 5,
    sizeVerCursor = 5,
    sizeHorCursor = 6,
    SizeHorCursor = 6,
    SizeBDiagCursor = 7,
    sizeBDiagCursor = 7,
    sizeFDiagCursor = 8,
    SizeFDiagCursor = 8,
    SizeAllCursor = 9,
    sizeAllCursor = 9,
    BlankCursor = 10,
    blankCursor = 10,

```

```

        splitVCursor = 11,
        SplitVCursor = 11,
        splitHCursor = 12,
        SplitHCursor = 12,
        PointingHandCursor = 13,
        pointingHandCursor = 13,
        ForbiddenCursor = 14,
        forbiddenCursor = 14,
        whatsThisCursor = 15,
        WhatsThisCursor = 15,
        BusyCursor = 16,
        LastCursor = 16,
        BitmapCursor = 24
    };
    enum Qt::TextFormat {
        PlainText = 0,
        RichText = 1,
        AutoText = 2,
        LogText = 3
    };
    enum Qt::AspectRatioMode {
        IgnoreAspectRatio = 0,
        ScaleFree = 0,
        KeepAspectRatio = 1,
        ScaleMin = 1,
        KeepAspectRatioByExpanding = 2,
        ScaleMax = 2
    };
    typedef enum Qt::AspectRatioMode {
        IgnoreAspectRatio = 0,
        ScaleFree = 0,
        KeepAspectRatio = 1,
        ScaleMin = 1,
        KeepAspectRatioByExpanding = 2,
        ScaleMax = 2
    } Qt::ScaleMode;
    enum Qt::AnchorAttribute {
        AnchorName = 0,
        AnchorHref = 1
    };
    enum Qt::DockWidgetArea {
        LeftDockWidgetArea = 1,
        RightDockWidgetArea = 2,
        TopDockWidgetArea = 4,
        BottomDockWidgetArea = 8,
        DockWidgetArea_Mask = 15,
        AllDockWidgetAreas = 15
    };
    typedef class QFlags < Qt::DockWidgetArea > Qt::DockWidgetAreas;
    enum Qt::ToolBarArea {
        LeftToolBarArea = 1,
        RightToolBarArea = 2,
        TopToolBarArea = 4,
        BottomToolBarArea = 8,
        ToolBarArea_Mask = 15,
        AllToolBarAreas = 15
    };
    typedef class QFlags < Qt::ToolBarArea > Qt::ToolBarAreas;
    enum Qt::Dock {
        DockUnmanaged = 0,
        Unmanaged = 0,
        TornOff = 1,
        DockTornOff = 1,
        Top = 2,
        DockTop = 2,
        DockBottom = 3,

```

```

        Bottom = 3,
        DockRight = 4,
        Right = 4,
        DockLeft = 5,
        Left = 5,
        DockMinimized = 6,
        Minimized = 6
    };
typedef enum Qt::Dock {
    DockUnmanaged = 0,
    Unmanaged = 0,
    TornOff = 1,
    DockTornOff = 1,
    Top = 2,
    DockTop = 2,
    DockBottom = 3,
    Bottom = 3,
    DockRight = 4,
    Right = 4,
    DockLeft = 5,
    Left = 5,
    DockMinimized = 6,
    Minimized = 6
} Qt::ToolBarDock;
enum Qt::DateFormat {
    TextDate = 0,
    ISODate = 1,
    LocalDate = 2
};
enum Qt::TimeSpec {
    LocalTime = 0,
    UTC = 1
};
enum Qt::DayOfWeek {
    Monday = 1,
    Tuesday = 2,
    Wednesday = 3,
    Thursday = 4,
    Friday = 5,
    Saturday = 6,
    Sunday = 7
};
enum Qt::ScrollBarPolicy {
    ScrollBarAsNeeded = 0,
    ScrollBarAlwaysOff = 1,
    ScrollBarAlwaysOn = 2
};
enum Qt::BackgroundMode {
    FixedColor = 0,
    FixedPixmap = 1,
    NoBackground = 2,
    PaletteForeground = 3,
    PaletteButton = 4,
    PaletteLight = 5,
    PaletteMidlight = 6,
    PaletteDark = 7,
    PaletteMid = 8,
    PaletteText = 9,
    PaletteBrightText = 10,
    PaletteBase = 11,
    PaletteBackground = 12,
    PaletteShadow = 13,
    PaletteHighlight = 14,
    PaletteHighlightedText = 15,
    PaletteButtonText = 16,
    PaletteLink = 17,

```

```

        PaletteLinkVisited = 18,
        X11ParentRelative = 19
    };
    enum Qt::CaseSensitivity {
        CaseInsensitive = 0,
        CaseSensitive = 1
    };
    enum Qt::Corner {
        TopLeftCorner = 0,
        TopLeft = 0,
        TopRightCorner = 1,
        TopRight = 1,
        BottomLeftCorner = 2,
        BottomLeft = 2,
        BottomRightCorner = 3,
        BottomRight = 3
    };
    enum Qt::ConnectionType {
        AutoConnection = 0,
        DirectConnection = 1,
        QueuedConnection = 2,
        AutoCompatConnection = 3
    };
    enum Qt::ShortcutContext {
        WidgetShortcut = 0,
        WindowShortcut = 1,
        ApplicationShortcut = 2
    };
    enum Qt::FillRule {
        OddEvenFill = 0,
        WindingFill = 1
    };
    enum Qt::ClipOperation {
        NoClip = 0,
        ReplaceClip = 1,
        IntersectClip = 2,
        UniteClip = 3
    };
    enum Qt::TransformationMode {
        FastTransformation = 0,
        SmoothTransformation = 1
    };
    enum Qt::FocusReason {
        MouseFocusReason = 0,
        TabFocusReason = 1,
        BacktabFocusReason = 2,
        ActiveWindowFocusReason = 3,
        PopupFocusReason = 4,
        ShortcutFocusReason = 5,
        MenuBarFocusReason = 6,
        OtherFocusReason = 7,
        NoFocusReason = 8
    };
    enum Qt::ContextMenuPolicy {
        NoContextMenu = 0,
        DefaultContextMenu = 1,
        ActionsContextMenu = 2,
        CustomContextMenu = 3
    };
    enum Qt::InputMethodQuery {
        ImMicroFocus = 0,
        ImFont = 1,
        ImCursorPosition = 2,
        ImSurroundingText = 3,
        ImCurrentSelection = 4
    };
};

```

```

enum Qt::ToolButtonStyle {
    ToolButtonIconOnly = 0,
    ToolButtonTextOnly = 1,
    ToolButtonTextBesideIcon = 2,
    ToolButtonTextUnderIcon = 3
};
enum Qt::LayoutDirection {
    LeftToRight = 0,
    RightToLeft = 1
};
enum Qt::DropAction {
    IgnoreAction = 0,
    CopyAction = 1,
    MoveAction = 2,
    LinkAction = 4,
    ActionMask = 255,
    TargetMoveAction = 32770
};
typedef class QFlags < Qt::DropAction > Qt::DropActions;
enum Qt::CheckState {
    Unchecked = 0,
    PartiallyChecked = 1,
    Checked = 2
};
enum Qt::ItemDataRole {
    DisplayRole = 0,
    DecorationRole = 1,
    EditRole = 2,
    ToolTipRole = 3,
    StatusTipRole = 4,
    WhatsThisRole = 5,
    FontRole = 6,
    TextAlignmentRole = 7,
    BackgroundColorRole = 8,
    TextColorRole = 9,
    CheckStateRole = 10,
    AccessibleTextRole = 11,
    AccessibleDescriptionRole = 12,
    SizeHintRole = 13,
    UserRole = 32
};
enum Qt::ItemFlag {
    ItemIsSelectable = 1,
    ItemIsEditable = 2,
    ItemIsDragEnabled = 4,
    ItemIsDropEnabled = 8,
    ItemIsUserCheckable = 16,
    ItemIsEnabled = 32,
    ItemIsTristate = 64
};
class QFlags < Qt::ItemFlag >;
typedef class QFlags < Qt::ItemFlag > Qt::ItemFlags;
enum Qt::MatchFlag {
    MatchExactly = 0,
    MatchContains = 1,
    MatchStartsWith = 2,
    MatchEndsWith = 3,
    MatchRegExp = 4,
    MatchWildcard = 5,
    MatchCaseSensitive = 16,
    MatchWrap = 32,
    MatchRecursive = 64
};
typedef class QFlags < Qt::MatchFlag > Qt::MatchFlags;
typedef unsigned long int Qt::HANDLE;
typedef Qt::WindowFlags Qt::WFlags;

```



```

enum Qt::WindowModality {
    NonModal = 0,
    WindowModal = 1,
    ApplicationModal = 2
};
class QInternal;
enum QInternal::PaintDeviceFlags {
    UnknownDevice = 0,
    Widget = 1,
    Pixmap = 2,
    Image = 3,
    Printer = 4,
    Picture = 5,
    Pbuffer = 6
};
enum QInternal::RelayoutType {
    RelayoutNormal = 0,
    RelayoutDragging = 1,
    RelayoutDropped = 2
};
typedef qint32 QCOORD;

```

18.3.37 QtCore/qobject.h

```

typedef class QList < QObject * >QObjectList;
class QObjectData;
class QObject;
class QObjectUserData;

```

18.3.38 QtCore/qobjectcleanuphandler.h

```

class QObjectCleanupHandler;

```

18.3.39 QtCore/qobjectdefs.h

```

class QGenericArgument;
class QGenericReturnArgument;
struct QMetaObject;
enum QMetaObject::Call {
    InvokeMetaMethod = 0,
    ReadProperty = 1,
    WriteProperty = 2,
    ResetProperty = 3,
    QueryPropertyDesignable = 4,
    QueryPropertyScriptable = 5,
    QueryPropertyStored = 6,
    QueryPropertyEditable = 7,
    QueryPropertyUser = 8
};

```

18.3.40 QtCore/qpair.h

```

struct QPair <int, int >;

```

18.3.41 QtCore/qplugin.h

```

typedef class QObjectQtPluginInstanceFunction;

```

18.3.42 QtCore/qpluginloader.h

```
class QPluginLoader;
```

18.3.43 QtCore/qpoint.h

```
class QPoint;
class QPointF;
```

18.3.44 QtCore/qprocess.h

```
typedef quint64 Q_PID;
class QProcess;
enum QProcess::ProcessError {
    FailedToStart = 0,
    Crashed = 1,
    Timedout = 2,
    ReadError = 3,
    WriteError = 4,
    UnknownError = 5
};
enum QProcess::ProcessState {
    NotRunning = 0,
    Starting = 1,
    Running = 2
};
enum QProcess::ProcessChannel {
    StandardOutput = 0,
    StandardError = 1
};
enum QProcess::ProcessChannelMode {
    SeparateChannels = 0,
    MergedChannels = 1,
    ForwardedChannels = 2
};
enum QProcess::ExitStatus {
    NormalExit = 0,
    CrashExit = 1
};
```

18.3.45 QtCore/qreadwritelock.h

```
typedef enum QtValidLicenseForCoreModule QtCoreModule;
class QReadWriteLock;
class QReadLocker;
class QWriteLocker;
```

18.3.46 QtCore/qrect.h

```
class QRect;
class QRectF;
```

18.3.47 QtCore/qregex.h

```
class QRegExp;
enum QRegExp::PatternSyntax {
    RegExp = 0,
    Wildcard = 1,
```

```

    FixedString = 2
};
enum QRegExp::CaretMode {
    CaretAtZero = 0,
    CaretAtOffset = 1,
    CaretWontMatch = 2
};

```

18.3.48 QtCore/qsemaphore.h

```
class QSemaphore;
```

18.3.49 QtCore/qsettings.h

```

class QSettings;
enum QSettings::Status {
    NoError = 0,
    AccessError = 1,
    FormatError = 2
};
enum QSettings::Format {
    NativeFormat = 0,
    IniFormat = 1,
    InvalidFormat = 16,
    CustomFormat1 = 17,
    CustomFormat2 = 18,
    CustomFormat3 = 19,
    CustomFormat4 = 20,
    CustomFormat5 = 21,
    CustomFormat6 = 22,
    CustomFormat7 = 23,
    CustomFormat8 = 24,
    CustomFormat9 = 25,
    CustomFormat10 = 26,
    CustomFormat11 = 27,
    CustomFormat12 = 28,
    CustomFormat13 = 29,
    CustomFormat14 = 30,
    CustomFormat15 = 31,
    CustomFormat16 = 32
};
enum QSettings::Scope {
    UserScope = 0,
    User = 0,
    SystemScope = 1,
    Global = 1
};
typedef class QMap < QString, QVariant > QSettings::SettingsMap;
typedef bool QSettings::ReadFunc;
typedef bool QSettings::WriteFunc;
enum QSettings::System {
    Unix = 0,
    Windows = 1,
    Mac = 2
};

```

18.3.50 QtCore/qshareddata.h

```
class QSharedData;
```

18.3.51 QtCore/qsignalmapper.h

```
class QSignalMapper;
```

18.3.52 QtCore/qsize.h

```
class QSize;
class QSizeF;
```

18.3.53 QtCore/qsocketnotifier.h

```
class QSocketNotifier;
enum QSocketNotifier::Type {
    Read = 0,
    Write = 1,
    Exception = 2
};
```

18.3.54 QtCore/qstring.h

```
class QString;
enum QString::SectionFlag {
    SectionDefault = 0,
    SectionSkipEmpty = 1,
    SectionIncludeLeadingSep = 2,
    SectionIncludeTrailingSep = 4,
    SectionCaseInsensitiveSeps = 8
};
class QFlags < QString::SectionFlag >;
typedef class QFlags < QString::SectionFlag > QString::SectionFlags;
enum QString::SplitBehavior {
    KeepEmptyParts = 0,
    SkipEmptyParts = 1
};
enum QString::NormalizationForm {
    NormalizationForm_D = 0,
    NormalizationForm_C = 1,
    NormalizationForm_KD = 2,
    NormalizationForm_KC = 3
};
typedef class QCharQString::iterator;
typedef class QCharQString::const_iterator;
typedef QString::iterator QString::Iterator;
typedef QString::const_iterator QString::ConstIterator;
struct QString::Null;
struct QString::Data;
class QLatin1String;
class QCharRef;
class QConstString;
```

18.3.55 QtCore/qstringlist.h

```
class QStringList;
```

18.3.56 QtCore/qstringmatcher.h

```
class QStringMatcher;
```

18.3.57 QtCore/qttemporaryfile.h

```
class QTemporaryFile;
```

18.3.58 QtCore/qttextcodec.h

```
class QTextCodec;
enum QTextCodec::ConversionFlag {
    ConvertInvalidToNull = -2147483648,
    DefaultConversion = 0,
    IgnoreHeader = 1
};
class QFlags < QTextCodec::ConversionFlag >;
typedef class QFlags < QTextCodec::ConversionFlag >
    QTextCodec::ConversionFlags;
struct QTextCodec::ConverterState;
class QTextEncoder;
class QTextDecoder;
```

18.3.59 QtCore/qttextcodecplugin.h

```
struct QTextCodecFactoryInterface;
class QTextCodecPlugin;
```

18.3.60 QtCore/qttextstream.h

```
class QTextStream;
enum QTextStream::RealNumberNotation {
    SmartNotation = 0,
    FixedNotation = 1,
    ScientificNotation = 2
};
enum QTextStream::FieldAlignment {
    AlignLeft = 0,
    AlignRight = 1,
    AlignCenter = 2,
    AlignAccountingStyle = 3
};
enum QTextStream::Status {
    Ok = 0,
    ReadPastEnd = 1,
    ReadCorruptData = 2
};
enum QTextStream::NumberFlag {
    ShowBase = 1,
    ForcePoint = 2,
    ForceSign = 4,
    UppercaseBase = 8,
    UppercaseDigits = 16
};
class QFlags < QTextStream::NumberFlag >;
typedef class QFlags < QTextStream::NumberFlag >
    QTextStream::NumberFlags;
enum QTextStream::Encoding {
    Locale = 0,
    Latin1 = 1,
    Unicode = 2,
    UnicodeNetworkOrder = 3,
    UnicodeReverse = 4,
    RawUnicode = 5,
```

```

        UnicodeUTF8 = 6
    };
typedef class QTextStreamQTextStreamFunction;
typedef void QTSMFI;
typedef void QTSMFC;
class QTextStreamManipulator;
typedef class QTextStreamQTS;
class QTextIStream;
class QTextOStream;

```

18.3.61 QtCore/qthread.h

```

class QThread;
enum QThread::Priority {
    IdlePriority = 0,
    LowestPriority = 1,
    LowPriority = 2,
    NormalPriority = 3,
    HighPriority = 4,
    HighestPriority = 5,
    TimeCriticalPriority = 6,
    InheritPriority = 7
};

```

18.3.62 QtCore/qthreadstorage.h

```

class QThreadStorageData;

```

18.3.63 QtCore/qtimer.h

```

class QTimer;

```

18.3.64 QtCore/qtranslator.h

```

class QTranslator;

```

18.3.65 QtCore/qurl.h

```

class QUrl;
enum QUrl::ParsingMode {
    TolerantMode = 0,
    StrictMode = 1
};
enum QUrl::FormattingOption {
    None = 0,
    RemoveScheme = 1,
    RemovePassword = 2,
    RemoveUserInfo = 6,
    RemovePort = 8,
    RemoveAuthority = 30,
    RemovePath = 32,
    RemoveQuery = 64,
    RemoveFragment = 128,
    StripTrailingSlash = 65536
};
class QFlags < QUrl::FormattingOption >;
typedef class QFlags < QUrl::FormattingOption >
    QUrl::FormattingOptions;

```

18.3.66 QtCore/quuid.h

```

struct QUuid;
enum QUuid::Variant {
    VarUnknown = -1,
    NCS = 0,
    DCE = 2,
    Microsoft = 6,
    Reserved = 7
};
enum QUuid::Version {
    VerUnknown = -1,
    Time = 1,
    EmbeddedPOSIX = 2,
    Name = 3,
    Random = 4
};

```

18.3.67 QtCore/qvariant.h

```

class QVariant;
enum QVariant::Type {
    LastType = -1,
    Invalid = 0,
    Bool = 1,
    Int = 2,
    UInt = 3,
    LongLong = 4,
    ULongLong = 5,
    Double = 6,
    Char = 7,
    Map = 8,
    List = 9,
    String = 10,
    StringList = 11,
    ByteArray = 12,
    CString = 12,
    BitArray = 13,
    Date = 14,
    Time = 15,
    DateTime = 16,
    Url = 17,
    Locale = 18,
    Rect = 19,
    RectF = 20,
    Size = 21,
    SizeF = 22,
    Line = 23,
    LineF = 24,
    Point = 25,
    PointF = 26,
    RegExp = 27,
    ColorGroup = 63,
    Font = 64,
    QPixmap = 65,
    Brush = 66,
    Color = 67,
    Palette = 68,
    IconSet = 69,
    Icon = 69,
    Image = 70,
    Polygon = 71,
    PointArray = 71,
};

```

```

    Region = 72,
    Bitmap = 73,
    Cursor = 74,
    SizePolicy = 75,
    KeySequence = 76,
    Pen = 77,
    TextLength = 78,
    TextFormat = 79,
    UserType = 127
};
struct QVariant::PrivateShared;
struct QVariant::Private;
union _ZN8QVariant7Private4DataE;
typedef void QVariant::f_construct;
typedef void QVariant::f_clear;
typedef bool QVariant::f_null;
typedef void QVariant::f_load;
typedef void QVariant::f_save;
typedef bool QVariant::f_compare;
typedef bool QVariant::f_convert;
typedef bool QVariant::f_canConvert;
typedef void QVariant::f_debugStream;
struct QVariant::Handler;
typedef class QList < QVariant > QVariantList;
typedef class QMap < QString, QVariant > QVariantMap;
class QVariantComparisonHelper;

```

18.3.68 QtCore/qvector.h

```

class QVector < QVariant >;
struct QVectorData;

```

18.3.69 QtCore/qwaitcondition.h

```

class QWaitCondition;

```

18.4 Interface Definitions for libQtCore

The interfaces defined on the following pages are included in libQtCore and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 18.2 shall behave as described in the referenced base document.

_Z5qFreePv

Name

qFree — free the memory space

Synopsis

```

#include <QtCore/qglobal.h>
void qFree (void *ptr);

```

Description

The qFree() function has the same behavior as the free() function described in POSIX 1003.1-2008 (ISO/IEC 9945-2009).

_Z7qgetenvPKc**Name**

qgetenv — get value of an environment variable

Synopsis

```
#include <QtCore/qglobal.h>
QByteArray qgetenv (const char *varName);
```

Description

The `qgetenv()` function has the same behavior as the `getenv()` function described in POSIX 1003.1-2008 (ISO/IEC 9945-2009) except that the result is represented as `QByteArray`.

_ZN11QMetaObject7connectEPK7QObjectiS2_iiPi**Name**

`QMetaObject::connect` — connect signals

Synopsis

```
#include <QtCore/qmetaobject.h>
bool QMetaObject::connect(const QObject * sender, int signal_index,
const QObject * receiver, int method_index, int type, int * types);
```

Description

The `QMetaObject::connect()` function shall connect the signal with `signal_index` index from the sender to the receiver's slot.

The `type` specifies connection type and should have one of the `Qt::ConnectionType` values.

`types` is a 0-terminated vector of meta types for queued connections.

If `signal_index` is -1, then the function shall effectively connect all signals from the sender to the receiver's slot.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN16QCoreApplication10unixSignalEi**Name**

`QCoreApplication::unixSignal` — emit a signal whenever a Unix signal is received

Synopsis

```
#include <QtCore/qcoreapplication.h>
```

```
void QCoreApplication::unixSignal (int number);
```

Description

The `QCoreApplication::unixSignal()` is a Qt signal function.

The signal is emitted whenever Unix signal is received by the application. The Unix signal received is specified by its number *number*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN4QDir21nameFiltersFromStringERK7QString

Name

`QDir::nameFiltersFromString` — retrieve a list of name filters from the given

Synopsis

```
#include <QtCore/qdir.h>
static QStringList QDir::nameFiltersFromString (const QString
&nameFilter);
```

Description

The `QDir::nameFiltersFromString()` function shall return a list of name filters from the given string *nameFilter*.

If there is more than one filter in the string, then each pair of filters should be separated by a space or by a semicolon.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN7QString4freeEPNS_4DataE

Name

`QString::free` — free the memory space

Synopsis

```
#include <QtCore/qstring.h>
static void QString::free (Data *d);
```

Description

The `QString::free()` function shall free the memory area allocated for the *d* object.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN9QHashData12allocateNodeEv**Name**

`QHashData::allocateNode` — allocate memory for the node

Synopsis

```
#include <QtCore/qhash.h>
void * QHashData::allocateNode(void);
```

Description

The `QHashData::allocateNode()` function shall allocate the memory for the object's node.

The call to this function is equivalent to the call for the `qMalloc()` function with the argument equal to the `nodeSize` property of the object.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN9QHashData8freeNodeEPv**Name**

`QHashData::freeNode` — free the memory occupied by the node

Synopsis

```
#include <QtCore/qhash.h>
void QHashData::freeNode(void * node);
```

Description

The `QHashData::freeNode()` function shall free the memory occupied by the *node*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN9QMetaType12registerTypeEPKcPFvPvEPFS2_PKvE**Name**

`QMetaType::registerType` — register a user type for marshalling

Synopsis

```
#include <QtCore/qmetatype.h>
static int QMetaType::registerType(const char * typeName, Destructor
destructor, Constructor constructor);
```

Description

The `QMetaType::registerType()` function shall register a user type for marshalling, with a `typeName` name, `destructor` destructor, and a `constructor` constructor.

The function shall return the type's handle on success, or -1 if the type could not be registered.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK11QMetaObject4castEP7QObject

Name

`QMetaObject::cast` — check if an object inherits from the meta-object

Synopsis

```
#include <QtCore/qobjectdefs.h>
QObject * QMetaObject::cast (QObject *obj);
```

Description

The `QMetaObject::cast()` function shall check if the object pointed by `obj` inherits from the current meta-object. If yes, the function shall return `obj`. Otherwise it shall return 0.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK13QMetaProperty12hasStdCppSetEv

Name

`QMetaProperty::hasStdCppSet` — check if a property follows "name" / "setName" pattern

Synopsis

```
#include <QtCore/qmetaobject.h>
bool QMetaProperty::hasStdCppSet(void);
```

Description

The `QMetaProperty::hasStdCppSet()` function shall return true if the property has a C++ setter function that follows Qt's standard "name" / "setName" pattern. This function can be queried in order to avoid expensive `QObject::setProperty()` calls. All properties in Qt should follow the pattern mentioned.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK21QPersistentModelIndex10internalIdEv**Name**

`QPersistentModelIndex::internalId` — get the id used to associate the index with the internal data structure

Synopsis

```
#include <QtCore/qabstractitemmodel.h>
qint64 QPersistentModelIndex::internalId (void);
```

Description

The `QPersistentModelIndex::internalId()` function shall return the internal integer identifier used by the model to associate the index with the internal data structure.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK21QPersistentModelIndex15internalPointerEv**Name**

`QPersistentModelIndex::internalPointer` — get the pointer used to associate the index with the internal data structure

Synopsis

```
#include <QtCore/qabstractitemmodel.h>
void * QPersistentModelIndex::internalPointer (void);
```

Description

The `QPersistentModelIndex::internalPointer()` function shall return the internal pointer used by the model to associate the index with the internal data structure.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

ZNK4QUrlItERKS**Name**

`QUrl::operator<` — compare URLs

Synopsis

```
#include <QtCore/qurl.h>
bool QUrl::operator< (const QUrl &url);
```

Description

This operator shall return true if the URL represented by its caller is "less than" the given URL.

This operator provides a means of ordering URLs. If two URLs are not equal then one of them should be "less than" the other. No exact comparison algorithm is specified.

This operator is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK5QFile10fileEngineEv**Name**

`QFile::fileEngine` — get the `QIOEngine` for this `QFile` object

Synopsis

```
#include <QtCore/qfile.h>
virtual QAbstractFileEngine * QFile::fileEngine (void);
```

Description

The `QFile::fileEngine()` function shall return the `QIOEngine` for this `QFile` object.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

QUuid::operator >**Name**

`QUuid::operator >` — UUID greater-than comparison operator

Synopsis

```
public class QUuid {
    bool operator >(const QUuid & other);
}
```

Description

`QUuid::operator >` has the same specification as in QtCore 4.2.0, except for differences noted below.

Return Value

Returns true if this QUuid has the same variant field as the other QUuid and is lexicographically after the other QUuid. If the other QUuid has a different variant field, the return value is unspecified.

QUuid::operator <**Name**

`QUuid::operator <` — UUID less-than comparison operator

Synopsis

```
public class QUuid {
    bool operator <(const QUuid & other);
}
```

Description

`QUuid::operator <` has the same specification as in QtCore 4.2.0, except for differences noted below.

Return Value

Returns true if this QUuid has the same variant field as the other QUuid and is lexicographically before the other QUuid. If the other QUuid has a different variant field, the return value is unspecified.

_ZNK7QObject9queryListEPKcS1_bb**Name**

`QObject::queryList` — search the children and optionally grandchildren of the object

Synopsis

```
#include <QtCore/qobject.h>
QObjectList QObject::queryList(const char * inheritsClass = 0 , const
char * objName = 0, bool regexpMatch = true, bool recursiveSearch =
true);
```

Description

The `QObject::queryList()` function shall search the children and optionally grandchildren of this object.

The function shall return a list of those objects that are named or that match *objName* and inherit *inheritsClass*. If *inheritsClass* is 0 (the default), all classes match. If *objName* is 0 (the default), all object names match.

If *regexpMatch* is true (the default), *objName* is a regular expression that the objects's names must match. The syntax is that of a `QRegExp`. If *regexpMatch* is false, *objName* is a string and object names must match it exactly.

Note that *inheritsClass* uses single inheritance from `QObject`, the way `inherits()` does. According to `inherits()`, `QWidget` inherits `QObject` but not `QPaintDevice`. This does not quite match reality, but is the best that can be done on the wide variety of compilers Qt supports.

Finally, if *recursiveSearch* is true (the default), the `QObject::queryList()` shall search n-th-generation as well as first-generation children.

Warning: Delete the obtained list as soon you have finished using it. The list contains pointers that may become invalid at almost any time without notice (as soon as the user closes a window you may have dangling pointers, for example).

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

See Also

`child()`, `children()`, `parent()`, `inherits()`, `objectName()`, `QRegExp()`

18.5 Interfaces for libQtGui

Table 18-98 defines the library name and shared object name for the `libQtGui` library

Table 18-98 libQtGui Definition

Library:	libQtGui
SONAME:	libQtGui.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI

[LSB] This Specification

[QtGui] QtGui 4.2.0

[QtXml] QtXml 4.2.0

18.5.1 Qt4 GUI Functions

18.5.1.1 Interfaces for Qt4 GUI Functions

An LSB conforming implementation shall provide the generic functions for Qt4 GUI Functions specified in Table 18-99, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-99 libQtGui - Qt4 GUI Functions Function Interfaces

_Z10qDrawArrowP8QPainterN2Qt9ArrowTypeENS1_8GUIStyleEbiiiiRK8QPaletteb [QtGui]	_Z13qDrawWinPanelP8QPainterRK5QRectRK8QPalettebPK6QBrush [QtGui]
_Z6bitBltp12QPaintDeviceRK6QPoinPKS_RK5QRectb [QtGui]	_Z6bitBltp12QPaintDeviceiiPK6QImageiiii [QtGui]
_Z6bitBltp12QPaintDeviceiiPKS_iiii [QtGui]	_Z6bitBltp6QImageiiPKS_iiii6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]
_Z7copyBltp7QPixmapiiPKS_iiii [QtGui]	_Z9qDrawItemP8QPainterN2Qt8GUIStyleEiiiiRK8QPalettebPK7QPixmapRK7QStringiPK6QColor [QtGui]
_Z9qItemRectP8QPainterN2Qt8GUIStyleEiiiiPK7QPixmapRK7QStringi [QtGui]	_ZN10QTextBlock8iteratorommEv [QtGui]
_ZN10QTextBlock8iteratorppEv [QtGui]	_ZN10QTextFrame8iteratoraSERKS0_ [QtGui]
_ZN10QTextFrame8iteratorommEv [QtGui]	_ZN10QTextFrame8iteratorppEv [QtGui]
_ZN20QStyleOptionSizeGripC1Ei [QtXml]	_ZN20QStyleOptionSizeGripC1Ev [QtXml]
_ZN20QStyleOptionSizeGripC2Ei [QtXml]	_ZN20QStyleOptionSizeGripC2Ev [QtXml]
_ZN2Qt12codecForHtmlERK10QByteArray [LSB]	_ZN2Qt15mightBeRichTextERK7QString [QtGui]
_ZN2Qt20convertFromPlainTextERK7QStringNS_14WhiteSpaceModeE [QtGui]	_ZN2Qt6escapeERK7QString [QtGui]
_ZNK10QTextBlock8iterator8fragmentEv [QtGui]	_ZNK10QTextFrame8iterator12currentBlockEv [QtGui]
_ZNK10QTextFrame8iterator12currentFrameEv [QtGui]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 GUI Functions specified in Table 18-100, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-100 libQtGui - Qt4 GUI Functions Deprecated Function Interfaces

_Z10qDrawArrowP8QPainterN2Qt9ArrowTypeENS1_8GUIStyleEbiiiiRK8QPaletteb [QtGui]	_Z9qDrawItemP8QPainterN2Qt8GUIStyleEiiiiRK8QPalettebPK7QPixmapRK7QStringiPK6QColor [QtGui]
_Z9qItemRectP8QPainterN2Qt8GUIStyleEiiiiibPK7QPixmapRK7QStringi [QtGui]	

An LSB conforming implementation shall provide the generic data interfaces for Qt4 GUI Functions specified in Table 18-101, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-101 libQtGui - Qt4 GUI Functions Data Interfaces

_ZN10QBoxLayout16staticMetaObjectE [QtGui]	_ZN10QClipboard16staticMetaObjectE [QtGui]
_ZN10QCompleter16staticMetaObjectE [QtXml]	_ZN10QLCDNumber16staticMetaObjectE [QtGui]
_ZN10QScrollBar16staticMetaObjectE [QtGui]	_ZN10QStatusBar16staticMetaObjectE [QtGui]
_ZN10QTabWidget16staticMetaObjectE [QtGui]	_ZN10QTableView16staticMetaObjectE [QtGui]
_ZN10QTextFrame16staticMetaObjectE [QtGui]	_ZN10QTextTable16staticMetaObjectE [QtGui]
_ZN10QUndoGroup16staticMetaObjectE [QtXml]	_ZN10QUndoStack16staticMetaObjectE [QtXml]
_ZN10QValidator16staticMetaObjectE [QtGui]	_ZN10QWorkspace16staticMetaObjectE [QtGui]
_ZN11QDockWidget16staticMetaObjectE [QtGui]	_ZN11QFileDialog16staticMetaObjectE [QtGui]
_ZN11QFocusFrame16staticMetaObjectE [QtGui]	_ZN11QFontDialog16staticMetaObjectE [QtGui]
_ZN11QGridLayout16staticMetaObjectE [QtGui]	_ZN11QHBoxLayout16staticMetaObjectE [QtGui]
_ZN11QHeaderView16staticMetaObjectE [QtGui]	_ZN11QListWidget16staticMetaObjectE [QtGui]
_ZN11QMainWindow16staticMetaObjectE [QtGui]	_ZN11QMessageBox16staticMetaObjectE [QtGui]
_ZN11QMotifStyle16staticMetaObjectE [QtGui]	_ZN11QProxyModel16staticMetaObjectE [QtGui]
_ZN11QPushButton16staticMetaObjectE [QtGui]	_ZN11QRubberBand16staticMetaObjectE [QtGui]
_ZN11QScrollArea16staticMetaObjectE [QtGui]	_ZN11QSizePolicy16staticMetaObjectE [QtGui]

_ZN11QTextFormat16staticMetaObjectE [QtGui]	_ZN11QTextObject16staticMetaObjectE [QtGui]
_ZN11QToolButton16staticMetaObjectE [QtGui]	_ZN11QTreeWidget16staticMetaObjectE [QtGui]
_ZN11QVBoxLayout16staticMetaObjectE [QtGui]	_ZN12QActionGroup16staticMetaObjectE [QtGui]
_ZN12QApplication16staticMetaObjectE [QtGui]	_ZN12QButtonGroup16staticMetaObjectE [QtGui]
_ZN12QColorDialog16staticMetaObjectE [QtGui]	_ZN12QCommonStyle16staticMetaObjectE [QtGui]
_ZN12QInputDialog16staticMetaObjectE [QtGui]	_ZN12QPrintDialog16staticMetaObjectE [QtGui]
_ZN12QProgressBar16staticMetaObjectE [QtGui]	_ZN12QRadioButton16staticMetaObjectE [QtGui]
_ZN12QStylePlugin16staticMetaObjectE [QtGui]	_ZN12QTableWidget16staticMetaObjectE [QtGui]
_ZN12QTextBrowser16staticMetaObjectE [QtGui]	_ZN13QDateTimeEdit16staticMetaObjectE [QtGui]
_ZN13QErrorMessage16staticMetaObjectE [QtGui]	_ZN13QFontComboBox16staticMetaObjectE [QtXml]
_ZN13QFontDatabase16staticMetaObjectE [QtXml]	_ZN13QGraphicsView16staticMetaObjectE [QtXml]
_ZN13QInputContext16staticMetaObjectE [QtGui]	_ZN13QIntValidator16staticMetaObjectE [QtGui]
_ZN13QItemDelegate16staticMetaObjectE [QtGui]	_ZN13QSplashScreen16staticMetaObjectE [QtGui]
_ZN13QTextDocument16staticMetaObjectE [QtGui]	_ZN13QWidgetAction16staticMetaObjectE [QtXml]
_ZN13QWindowsStyle16staticMetaObjectE [QtGui]	_ZN14QDesktopWidget16staticMetaObjectE [QtGui]
_ZN14QDoubleSpinBox16staticMetaObjectE [QtGui]	_ZN14QGraphicsScene16staticMetaObjectE [QtXml]
_ZN14QImageIOPlugin16staticMetaObjectE [QtGui]	_ZN14QStackedLayout16staticMetaObjectE [QtGui]
_ZN14QStackedWidget16staticMetaObjectE [QtGui]	_ZN15QAbstractButton16staticMetaObjectE [QtGui]
_ZN15QAbstractSlider16staticMetaObjectE [QtGui]	_ZN15QCalendarWidget16staticMetaObjectE [QtXml]
_ZN15QPlastiqueStyle16staticMetaObjectE [QtGui]	_ZN15QProgressDialog16staticMetaObjectE [QtGui]
_ZN15QSessionManager16staticMetaObjectE [QtGui]	_ZN15QSplitterHandle16staticMetaObjectE [QtGui]

_ZN15QSystemTrayIcon16staticMetaObjectE [QtXml]	_ZN15QTextBlockGroup16staticMetaObjectE [QtGui]
_ZN16QAbstractSpinBox16staticMetaObjectE [QtGui]	_ZN16QCleanlooksStyle16staticMetaObjectE [QtXml]
_ZN16QDialogButtonBox16staticMetaObjectE [QtXml]	_ZN16QDoubleValidator16staticMetaObjectE [QtGui]
_ZN16QPageSetupDialog16staticMetaObjectE [QtXml]	_ZN16QRegExpValidator16staticMetaObjectE [QtGui]
_ZN16QStringListModel16staticMetaObjectE [QtGui]	_ZN17QAbstractItemView16staticMetaObjectE [QtGui]
_ZN17QAccessiblePlugin16staticMetaObjectE [QtGui]	_ZN17QDataWidgetMapper16staticMetaObjectE [QtXml]
_ZN17QGraphicsTextItem16staticMetaObjectE [QtXml]	_ZN17QIconEnginePlugin16staticMetaObjectE [QtGui]
_ZN18QStandardItemModel16staticMetaObjectE [QtGui]	_ZN18QSyntaxHighlighter16staticMetaObjectE [QtGui]
_ZN19QAbstractProxyModel16staticMetaObjectE [QtGui]	_ZN19QAbstractScrollArea16staticMetaObjectE [QtGui]
_ZN19QInputContextPlugin16staticMetaObjectE [QtGui]	_ZN19QItemSelectionModel16staticMetaObjectE [QtGui]
_ZN20QAbstractPrintDialog16staticMetaObjectE [QtXml]	_ZN20QPictureFormatPlugin16staticMetaObjectE [QtGui]
_ZN21QAbstractItemDelegate16staticMetaObjectE [QtGui]	_ZN21QSortFilterProxyModel16staticMetaObjectE [QtGui]
_ZN22QGraphicsItemAnimation16staticMetaObjectE [QtXml]	_ZN23QAccessibleBridgePlugin16staticMetaObjectE [QtGui]
_ZN24QAbstractPageSetupDialog16staticMetaObjectE [QtXml]	_ZN27QAbstractTextDocumentLayout16staticMetaObjectE [QtGui]
_ZN5QDialog16staticMetaObjectE [QtGui]	_ZN5QDrag16staticMetaObjectE [QtGui]
_ZN5QMenu16staticMetaObjectE [QtGui]	_ZN6QFrame16staticMetaObjectE [QtGui]
_ZN6QLabel16staticMetaObjectE [QtGui]	_ZN6QMovie16staticMetaObjectE [QtGui]
_ZN6QSound16staticMetaObjectE [QtGui]	_ZN6QStyle16staticMetaObjectE [QtGui]
_ZN7QAction16staticMetaObjectE [QtGui]	_ZN7QDialog16staticMetaObjectE [QtGui]
_ZN7QLayout16staticMetaObjectE [QtGui]	_ZN7QSlider16staticMetaObjectE [QtGui]
_ZN7QTabBar16staticMetaObjectE [QtGui]	_ZN7QWidget16staticMetaObjectE [QtGui]

_ZN8QMenuBar16staticMetaObjectE [QtGui]	_ZN8QPainter16staticMetaObjectE [QtXml]
_ZN8QPalette16staticMetaObjectE [QtGui]	_ZN8QSpinBox16staticMetaObjectE [QtGui]
_ZN8QToolBar16staticMetaObjectE [QtGui]	_ZN8QToolBox16staticMetaObjectE [QtGui]
_ZN9QCDEStyle16staticMetaObjectE [QtGui]	_ZN9QCheckBox16staticMetaObjectE [QtGui]
_ZN9QComboBox16staticMetaObjectE [QtGui]	_ZN9QDateEdit16staticMetaObjectE [QtGui]
_ZN9QDirModel16staticMetaObjectE [QtGui]	_ZN9QGradient16staticMetaObjectE [QtXml]
_ZN9QGroupBox16staticMetaObjectE [QtGui]	_ZN9QLineEdit16staticMetaObjectE [QtGui]
_ZN9QListView16staticMetaObjectE [QtGui]	_ZN9QMenuItem16staticMetaObjectE [QtXml]
_ZN9QShortcut16staticMetaObjectE [QtGui]	_ZN9QSizeGrip16staticMetaObjectE [QtGui]
_ZN9QSplitter16staticMetaObjectE [QtGui]	_ZN9QTextEdit16staticMetaObjectE [QtGui]
_ZN9QTextList16staticMetaObjectE [QtGui]	_ZN9QTimeEdit16staticMetaObjectE [QtGui]
_ZN9QTreeView16staticMetaObjectE [QtGui]	_ZN9QUndoView16staticMetaObjectE [QtXml]
_ZTI10QBoxLayout [CXXABI-1.86]	_ZTI10QClipboard [CXXABI-1.86]
_ZTI10QCompleter [CXXABI-1.86]	_ZTI10QDropEvent [CXXABI-1.86]
_ZTI10QHelpEvent [CXXABI-1.86]	_ZTI10QHideEvent [CXXABI-1.86]
_ZTI10QLCDNumber [CXXABI-1.86]	_ZTI10QMoveEvent [CXXABI-1.86]
_ZTI10QScrollBar [CXXABI-1.86]	_ZTI10QShowEvent [CXXABI-1.86]
_ZTI10QStatusBar [CXXABI-1.86]	_ZTI10QTabWidget [CXXABI-1.86]
_ZTI10QTableView [CXXABI-1.86]	_ZTI10QTextFrame [CXXABI-1.86]
_ZTI10QTextTable [CXXABI-1.86]	_ZTI10QUndoGroup [CXXABI-1.86]
_ZTI10QUndoStack [CXXABI-1.86]	_ZTI10QValidator [CXXABI-1.86]
_ZTI10QWorkspace [CXXABI-1.86]	_ZTI11QAccessible [CXXABI-1.86]
_ZTI11QCloseEvent [CXXABI-1.86]	_ZTI11QDockWidget [CXXABI-1.86]
_ZTI11QFileDialog [CXXABI-1.86]	_ZTI11QFocusEvent [CXXABI-1.86]
_ZTI11QFocusFrame [CXXABI-1.86]	_ZTI11QFontDialog [CXXABI-1.86]
_ZTI11QGridLayout [CXXABI-1.86]	_ZTI11QHBoxLayout [CXXABI-1.86]
_ZTI11QHeaderView [CXXABI-1.86]	_ZTI11QHoverEvent [CXXABI-1.86]

_ZTI11QIconEngine [CXXABI-1.86]	_ZTI11QInputEvent [CXXABI-1.86]
_ZTI11QLayoutItem [CXXABI-1.86]	_ZTI11QListWidget [CXXABI-1.86]
_ZTI11QMainWindow [CXXABI-1.86]	_ZTI11QMessageBox [CXXABI-1.86]
_ZTI11QMimeSource [CXXABI-1.86]	_ZTI11QMotifStyle [CXXABI-1.86]
_ZTI11QMouseEvent [CXXABI-1.86]	_ZTI11QPaintEvent [CXXABI-1.86]
_ZTI11QProxyModel [CXXABI-1.86]	_ZTI11QPushButton [CXXABI-1.86]
_ZTI11QRubberBand [CXXABI-1.86]	_ZTI11QScrollArea [CXXABI-1.86]
_ZTI11QSpacerItem [CXXABI-1.86]	_ZTI11QStrokerOps [CXXABI-1.86]
_ZTI11QTextObject [CXXABI-1.86]	_ZTI11QToolButton [CXXABI-1.86]
_ZTI11QTreeWidget [CXXABI-1.86]	_ZTI11QVBoxLayout [CXXABI-1.86]
_ZTI11QWheelEvent [CXXABI-1.86]	_ZTI11QWidgetItem [CXXABI-1.86]
_ZTI12QActionEvent [CXXABI-1.86]	_ZTI12QActionGroup [CXXABI-1.86]
_ZTI12QApplication [CXXABI-1.86]	_ZTI12QButtonGroup [CXXABI-1.86]
_ZTI12QColorDialog [CXXABI-1.86]	_ZTI12QCommonStyle [CXXABI-1.86]
_ZTI12QDashStroker [CXXABI-1.86]	_ZTI12QInputDialog [CXXABI-1.86]
_ZTI12QPaintDevice [CXXABI-1.86]	_ZTI12QPaintEngine [CXXABI-1.86]
_ZTI12QPrintDialog [CXXABI-1.86]	_ZTI12QPrintEngine [CXXABI-1.86]
_ZTI12QProgressBar [CXXABI-1.86]	_ZTI12QRadioButton [CXXABI-1.86]
_ZTI12QResizeEvent [CXXABI-1.86]	_ZTI12QStylePlugin [CXXABI-1.86]
_ZTI12QTableWidget [CXXABI-1.86]	_ZTI12QTabletEvent [CXXABI-1.86]
_ZTI12QTextBrowser [CXXABI-1.86]	_ZTI12QUndoCommand [CXXABI-1.86]
_ZTI13QDateTimeEdit [CXXABI-1.86]	_ZTI13QErrorMessage [CXXABI-1.86]
_ZTI13QFontComboBox [CXXABI-1.86]	_ZTI13QFontEngineFT [CXXABI-1.86]
_ZTI13QGraphicsItem [CXXABI-1.86]	_ZTI13QGraphicsView [CXXABI-1.86]
_ZTI13QInputContext [CXXABI-1.86]	_ZTI13QIntValidator [CXXABI-1.86]
_ZTI13QItemDelegate [CXXABI-1.86]	_ZTI13QSplashScreen [CXXABI-1.86]
_ZTI13QStandardItem [CXXABI-1.86]	_ZTI13QTextDocument [CXXABI-1.86]
_ZTI13QWidgetAction [CXXABI-1.86]	_ZTI13QWindowsStyle [CXXABI-1.86]
_ZTI14QDesktopWidget [CXXABI-1.86]	_ZTI14QDoubleSpinBox [CXXABI-1.86]

_ZTI14QDragMoveEvent [CXXABI-1.86]	_ZTI14QFileOpenEvent [CXXABI-1.86]
_ZTI14QGraphicsScene [CXXABI-1.86]	_ZTI14QIconDragEvent [CXXABI-1.86]
_ZTI14QImageIOPlugin [CXXABI-1.86]	_ZTI14QLayoutPrivate [CXXABI-1.86]
_ZTI14QShortcutEvent [CXXABI-1.86]	_ZTI14QStackedLayout [CXXABI-1.86]
_ZTI14QStackedWidget [CXXABI-1.86]	_ZTI14QWidgetPrivate [CXXABI-1.86]
_ZTI15QAbstractButton [CXXABI-1.86]	_ZTI15QAbstractSlider [CXXABI-1.86]
_ZTI15QCalendarWidget [CXXABI-1.86]	_ZTI15QClipboardEvent [CXXABI-1.86]
_ZTI15QDragEnterEvent [CXXABI-1.86]	_ZTI15QDragLeaveEvent [CXXABI-1.86]
_ZTI15QImageIOHandler [CXXABI-1.86]	_ZTI15QListWidgetItem [CXXABI-1.86]
_ZTI15QPlastiqueStyle [CXXABI-1.86]	_ZTI15QProgressDialog [CXXABI-1.86]
_ZTI15QSessionManager [CXXABI-1.86]	_ZTI15QSplitterHandle [CXXABI-1.86]
_ZTI15QStatusTipEvent [CXXABI-1.86]	_ZTI15QSystemTrayIcon [CXXABI-1.86]
_ZTI15QTextBlockGroup [CXXABI-1.86]	_ZTI15QTreeWidgetItem [CXXABI-1.86]
_ZTI15QX11EmbedWidget [CXXABI-1.86]	_ZTI16QAbstractSpinBox [CXXABI-1.86]
_ZTI16QCleanlooksStyle [CXXABI-1.86]	_ZTI16QDialogButtonBox [CXXABI-1.86]
_ZTI16QDoubleValidator [CXXABI-1.86]	_ZTI16QPageSetupDialog [CXXABI-1.86]
_ZTI16QRegExpValidator [CXXABI-1.86]	_ZTI16QStringListModel [CXXABI-1.86]
_ZTI16QTableWidgetItem [CXXABI-1.86]	_ZTI17QAbstractItemView [CXXABI-1.86]
_ZTI17QAccessibleObject [CXXABI-1.86]	_ZTI17QAccessiblePlugin [CXXABI-1.86]
_ZTI17QAccessibleWidget [CXXABI-1.86]	_ZTI17QContextMenuEvent [CXXABI-1.86]
_ZTI17QDataWidgetMapper [CXXABI-1.86]	_ZTI17QFactoryInterface [CXXABI-1.86]

_ZTI17QFileIconProvider [CXXABI-1.86]	_ZTI17QGraphicsLineItem [CXXABI-1.86]
_ZTI17QGraphicsPathItem [CXXABI-1.86]	_ZTI17QGraphicsRectItem [CXXABI-1.86]
_ZTI17QGraphicsTextItem [CXXABI-1.86]	_ZTI17QIconEnginePlugin [CXXABI-1.86]
_ZTI17QInputMethodEvent [CXXABI-1.86]	_ZTI18QDragResponseEvent [CXXABI-1.86]
_ZTI18QGraphicsItemGroup [CXXABI-1.86]	_ZTI18QItemEditorFactory [CXXABI-1.86]
_ZTI18QStandardItemModel [CXXABI-1.86]	_ZTI18QSyntaxHighlighter [CXXABI-1.86]
_ZTI18QTextBlockUserData [CXXABI-1.86]	_ZTI18QX11EmbedContainer [CXXABI-1.86]
_ZTI19QAbstractProxyModel [CXXABI-1.86]	_ZTI19QAbstractScrollArea [CXXABI-1.86]
_ZTI19QApplicationPrivate [CXXABI-1.86]	_ZTI19QGraphicsPixmapItem [CXXABI-1.86]
_ZTI19QGraphicsSceneEvent [CXXABI-1.86]	_ZTI19QInputContextPlugin [CXXABI-1.86]
_ZTI19QItemSelectionModel [CXXABI-1.86]	_ZTI19QToolBarChangeEvent [CXXABI-1.86]
_ZTI20QAbstractPrintDialog [CXXABI-1.86]	_ZTI20QAccessibleInterface [CXXABI-1.86]
_ZTI20QGraphicsEllipseItem [CXXABI-1.86]	_ZTI20QGraphicsPolygonItem [CXXABI-1.86]
_ZTI20QMenuBarUpdatedEvent [CXXABI-1.86]	_ZTI20QPictureFormatPlugin [CXXABI-1.86]
_ZTI20QTextFrameLayoutData [CXXABI-1.86]	_ZTI20QTextObjectInterface [CXXABI-1.86]
_ZTI20QWidgetResizeHandler [CXXABI-1.86]	_ZTI21QAbstractItemDelegate [CXXABI-1.86]
_ZTI21QSortFilterProxyModel [CXXABI-1.86]	_ZTI22QAccessibleApplication [CXXABI-1.86]
_ZTI22QGraphicsItemAnimation [CXXABI-1.86]	_ZTI22QStyleFactoryInterface [CXXABI-1.86]
_ZTI22QWhatsThisClickedEvent [CXXABI-1.86]	_ZTI23QAccessibleBridgePlugin [CXXABI-1.86]
_ZTI23QGraphicsSceneHelpEvent [CXXABI-1.86]	_ZTI23QGraphicsSimpleTextItem [CXXABI-1.86]
_ZTI23QPictureFormatInterface [CXXABI-1.86]	_ZTI23QWindowStateChangeEvent [CXXABI-1.86]

_ZTI24QAbstractPageSetupDialog [CXXABI-1.86]	_ZTI24QGraphicsSceneHoverEvent [CXXABI-1.86]
_ZTI24QGraphicsSceneMouseEvent [CXXABI-1.86]	_ZTI24QGraphicsSceneWheelEvent [CXXABI-1.86]
_ZTI26QAbstractGraphicsShapeItem [CXXABI-1.86]	_ZTI27QAbstractTextDocumentLayout [CXXABI-1.86]
_ZTI27QAccessibleFactoryInterface [CXXABI-1.86]	_ZTI27QGraphicsSceneDragDropEvent [CXXABI-1.86]
_ZTI27QIconEngineFactoryInterface [CXXABI-1.86]	_ZTI29QInputContextFactoryInterface [CXXABI-1.86]
_ZTI30QGraphicsSceneContextMenuEvent [CXXABI-1.86]	_ZTI31QImageIOHandlerFactoryInterface [CXXABI-1.86]
_ZTI33QAccessibleBridgeFactoryInterface [CXXABI-1.86]	_ZTI5QDial [CXXABI-1.86]
_ZTI5QDrag [CXXABI-1.86]	_ZTI5QMenu [CXXABI-1.86]
_ZTI6QFrame [CXXABI-1.86]	_ZTI6QImage [CXXABI-1.86]
_ZTI6QLabel [CXXABI-1.86]	_ZTI6QMovie [CXXABI-1.86]
_ZTI6QSound [CXXABI-1.86]	_ZTI6QStyle [CXXABI-1.86]
_ZTI7QAction [CXXABI-1.86]	_ZTI7QBitmap [CXXABI-1.86]
_ZTI7QDialog [CXXABI-1.86]	_ZTI7QLayout [CXXABI-1.86]
_ZTI7QPixmap [CXXABI-1.86]	_ZTI7QSlider [CXXABI-1.86]
_ZTI7QTabBar [CXXABI-1.86]	_ZTI7QWidget [CXXABI-1.86]
_ZTI8QMenuBar [CXXABI-1.86]	_ZTI8QPicture [CXXABI-1.86]
_ZTI8QPrinter [CXXABI-1.86]	_ZTI8QSpinBox [CXXABI-1.86]
_ZTI8QStroker [CXXABI-1.86]	_ZTI8QToolBar [CXXABI-1.86]
_ZTI8QToolBox [CXXABI-1.86]	_ZTI9QCDEStyle [CXXABI-1.86]
_ZTI9QCheckBox [CXXABI-1.86]	_ZTI9QComboBox [CXXABI-1.86]
_ZTI9QDateEdit [CXXABI-1.86]	_ZTI9QDirModel [CXXABI-1.86]
_ZTI9QGroupBox [CXXABI-1.86]	_ZTI9QKeyEvent [CXXABI-1.86]
_ZTI9QLineEdit [CXXABI-1.86]	_ZTI9QListView [CXXABI-1.86]
_ZTI9QMenuItem [CXXABI-1.86]	_ZTI9QShortcut [CXXABI-1.86]
_ZTI9QSizeGrip [CXXABI-1.86]	_ZTI9QSplitter [CXXABI-1.86]
_ZTI9QTextEdit [CXXABI-1.86]	_ZTI9QTextList [CXXABI-1.86]
_ZTI9QTimeEdit [CXXABI-1.86]	_ZTI9QTreeView [CXXABI-1.86]
_ZTI9QUndoView [CXXABI-1.86]	_ZTV10QBoxLayout [CXXABI-1.86]
_ZTV10QClipboard [CXXABI-1.86]	_ZTV10QCompleter [CXXABI-1.86]
_ZTV10QDropEvent [CXXABI-1.86]	_ZTV10QHelpEvent [CXXABI-1.86]

_ZTV10QHideEvent [CXXABI-1.86]	_ZTV10QLCDNumber [CXXABI-1.86]
_ZTV10QMoveEvent [CXXABI-1.86]	_ZTV10QScrollBar [CXXABI-1.86]
_ZTV10QShowEvent [CXXABI-1.86]	_ZTV10QStatusBar [CXXABI-1.86]
_ZTV10QTabWidget [CXXABI-1.86]	_ZTV10QTableView [CXXABI-1.86]
_ZTV10QTextFrame [CXXABI-1.86]	_ZTV10QTextTable [CXXABI-1.86]
_ZTV10QUndoGroup [CXXABI-1.86]	_ZTV10QUndoStack [CXXABI-1.86]
_ZTV10QValidator [CXXABI-1.86]	_ZTV10QWorkspace [CXXABI-1.86]
_ZTV11QCloseEvent [CXXABI-1.86]	_ZTV11QDockWidget [CXXABI-1.86]
_ZTV11QFileDialog [CXXABI-1.86]	_ZTV11QFocusEvent [CXXABI-1.86]
_ZTV11QFocusFrame [CXXABI-1.86]	_ZTV11QFontDialog [CXXABI-1.86]
_ZTV11QGridLayout [CXXABI-1.86]	_ZTV11QHBoxLayout [CXXABI-1.86]
_ZTV11QHeaderView [CXXABI-1.86]	_ZTV11QHoverEvent [CXXABI-1.86]
_ZTV11QIconEngine [CXXABI-1.86]	_ZTV11QInputEvent [CXXABI-1.86]
_ZTV11QLayoutItem [CXXABI-1.86]	_ZTV11QListWidget [CXXABI-1.86]
_ZTV11QMainWindow [CXXABI-1.86]	_ZTV11QMessageBox [CXXABI-1.86]
_ZTV11QMimeSource [CXXABI-1.86]	_ZTV11QMotifStyle [CXXABI-1.86]
_ZTV11QMouseEvent [CXXABI-1.86]	_ZTV11QPaintEvent [CXXABI-1.86]
_ZTV11QProxyModel [CXXABI-1.86]	_ZTV11QPushButton [CXXABI-1.86]
_ZTV11QRubberBand [CXXABI-1.86]	_ZTV11QScrollArea [CXXABI-1.86]
_ZTV11QSpacerItem [CXXABI-1.86]	_ZTV11QStrokerOps [CXXABI-1.86]
_ZTV11QTextObject [CXXABI-1.86]	_ZTV11QToolButton [CXXABI-1.86]
_ZTV11QTreeWidget [CXXABI-1.86]	_ZTV11QVBoxLayout [CXXABI-1.86]
_ZTV11QWheelEvent [CXXABI-1.86]	_ZTV11QWidgetItem [CXXABI-1.86]
_ZTV12QActionEvent [CXXABI-1.86]	_ZTV12QActionGroup [CXXABI-1.86]
_ZTV12QApplication [CXXABI-1.86]	_ZTV12QButtonGroup [CXXABI-1.86]
_ZTV12QColorDialog [CXXABI-1.86]	_ZTV12QCommonStyle [CXXABI-1.86]
_ZTV12QDashStroker [CXXABI-1.86]	_ZTV12QInputDialog [CXXABI-1.86]
_ZTV12QPaintDevice [CXXABI-1.86]	_ZTV12QPaintEngine [CXXABI-1.86]
_ZTV12QPrintDialog [CXXABI-1.86]	_ZTV12QPrintEngine [CXXABI-1.86]

_ZTV12QProgressBar [CXXABI-1.86]	_ZTV12QRadioButton [CXXABI-1.86]
_ZTV12QResizeEvent [CXXABI-1.86]	_ZTV12QStylePlugin [CXXABI-1.86]
_ZTV12QTableWidget [CXXABI-1.86]	_ZTV12QTabletEvent [CXXABI-1.86]
_ZTV12QTextBrowser [CXXABI-1.86]	_ZTV12QUndoCommand [CXXABI-1.86]
_ZTV13QDateTimeEdit [CXXABI-1.86]	_ZTV13QErrorMessage [CXXABI-1.86]
_ZTV13QFontEngineFT [CXXABI-1.86]	_ZTV13QGraphicsItem [CXXABI-1.86]
_ZTV13QInputContext [CXXABI-1.86]	_ZTV13QIntValidator [CXXABI-1.86]
_ZTV13QItemDelegate [CXXABI-1.86]	_ZTV13QSplashScreen [CXXABI-1.86]
_ZTV13QStandardItem [CXXABI-1.86]	_ZTV13QTextDocument [CXXABI-1.86]
_ZTV13QWidgetAction [CXXABI-1.86]	_ZTV13QWindowsStyle [CXXABI-1.86]
_ZTV14QDesktopWidget [CXXABI-1.86]	_ZTV14QDoubleSpinBox [CXXABI-1.86]
_ZTV14QDragMoveEvent [CXXABI-1.86]	_ZTV14QFileOpenEvent [CXXABI-1.86]
_ZTV14QGraphicsScene [CXXABI-1.86]	_ZTV14QIconDragEvent [CXXABI-1.86]
_ZTV14QImageIOPlugin [CXXABI-1.86]	_ZTV14QLayoutPrivate [CXXABI-1.86]
_ZTV14QShortcutEvent [CXXABI-1.86]	_ZTV14QStackedLayout [CXXABI-1.86]
_ZTV14QStackedWidget [CXXABI-1.86]	_ZTV14QWidgetPrivate [CXXABI-1.86]
_ZTV15QAbstractButton [CXXABI-1.86]	_ZTV15QAbstractSlider [CXXABI-1.86]
_ZTV15QClipboardEvent [CXXABI-1.86]	_ZTV15QDragEnterEvent [CXXABI-1.86]
_ZTV15QDragLeaveEvent [CXXABI-1.86]	_ZTV15QImageIOHandler [CXXABI-1.86]
_ZTV15QListWidgetItem [CXXABI-1.86]	_ZTV15QPlastiqueStyle [CXXABI-1.86]
_ZTV15QProgressDialog [CXXABI-1.86]	_ZTV15QSessionManager [CXXABI-1.86]

_ZTV15QSplitterHandle [CXXABI-1.86]	_ZTV15QStatusTipEvent [CXXABI-1.86]
_ZTV15QSystemTrayIcon [CXXABI-1.86]	_ZTV15QTextBlockGroup [CXXABI-1.86]
_ZTV15QTreeWidgetItem [CXXABI-1.86]	_ZTV15QX11EmbedWidget [CXXABI-1.86]
_ZTV16QAbstractSpinBox [CXXABI-1.86]	_ZTV16QCleanlooksStyle [CXXABI-1.86]
_ZTV16QDoubleValidator [CXXABI-1.86]	_ZTV16QPageSetupDialog [CXXABI-1.86]
_ZTV16QRegExpValidator [CXXABI-1.86]	_ZTV16QStringListModel [CXXABI-1.86]
_ZTV16QTableWidgetItem [CXXABI-1.86]	_ZTV17QAbstractItemView [CXXABI-1.86]
_ZTV17QAccessibleObject [CXXABI-1.86]	_ZTV17QAccessiblePlugin [CXXABI-1.86]
_ZTV17QAccessibleWidget [CXXABI-1.86]	_ZTV17QContextMenuEvent [CXXABI-1.86]
_ZTV17QDataWidgetMapper [CXXABI-1.86]	_ZTV17QFactoryInterface [CXXABI-1.86]
_ZTV17QFileIconProvider [CXXABI-1.86]	_ZTV17QGraphicsLineItem [CXXABI-1.86]
_ZTV17QGraphicsPathItem [CXXABI-1.86]	_ZTV17QGraphicsRectItem [CXXABI-1.86]
_ZTV17QIconEnginePlugin [CXXABI-1.86]	_ZTV17QInputMethodEvent [CXXABI-1.86]
_ZTV18QDragResponseEvent [CXXABI-1.86]	_ZTV18QGraphicsItemGroup [CXXABI-1.86]
_ZTV18QItemEditorFactory [CXXABI-1.86]	_ZTV18QStandardItemModel [CXXABI-1.86]
_ZTV18QSyntaxHighlighter [CXXABI-1.86]	_ZTV18QTextBlockUserData [CXXABI-1.86]
_ZTV18QX11EmbedContainer [CXXABI-1.86]	_ZTV19QAbstractProxyModel [CXXABI-1.86]
_ZTV19QAbstractScrollArea [CXXABI-1.86]	_ZTV19QApplicationPrivate [CXXABI-1.86]
_ZTV19QGraphicsPixmapItem [CXXABI-1.86]	_ZTV19QGraphicsSceneEvent [CXXABI-1.86]
_ZTV19QInputContextPlugin [CXXABI-1.86]	_ZTV19QItemSelectionModel [CXXABI-1.86]
_ZTV19QToolBarChangeEvent [CXXABI-1.86]	_ZTV20QAbstractPrintDialog [CXXABI-1.86]

_ZTV20QAccessibleInterface [CXXABI-1.86]	_ZTV20QGraphicsEllipseItem [CXXABI-1.86]
_ZTV20QGraphicsPolygonItem [CXXABI-1.86]	_ZTV20QMenuBarUpdatedEvent [CXXABI-1.86]
_ZTV20QPictureFormatPlugin [CXXABI-1.86]	_ZTV20QTextFrameLayoutData [CXXABI-1.86]
_ZTV20QTextObjectInterface [CXXABI-1.86]	_ZTV20QWidgetResizeHandler [CXXABI-1.86]
_ZTV21QAbstractItemDelegate [CXXABI-1.86]	_ZTV21QSortFilterProxyModel [CXXABI-1.86]
_ZTV22QAccessibleApplication [CXXABI-1.86]	_ZTV22QGraphicsItemAnimation [CXXABI-1.86]
_ZTV22QStyleFactoryInterface [CXXABI-1.86]	_ZTV22QWhatsThisClickedEvent [CXXABI-1.86]
_ZTV23QAccessibleBridgePlugin [CXXABI-1.86]	_ZTV23QGraphicsSceneHelpEvent [CXXABI-1.86]
_ZTV23QGraphicsSimpleTextItem [CXXABI-1.86]	_ZTV23QPictureFormatInterface [CXXABI-1.86]
_ZTV23QWindowStateChangeEvent [CXXABI-1.86]	_ZTV24QAbstractPageSetupDialog [CXXABI-1.86]
_ZTV24QGraphicsSceneHoverEvent [CXXABI-1.86]	_ZTV24QGraphicsSceneMouseEvent [CXXABI-1.86]
_ZTV24QGraphicsSceneWheelEvent [CXXABI-1.86]	_ZTV26QAbstractGraphicsShapeItem [CXXABI-1.86]
_ZTV27QAbstractTextDocumentLayout [CXXABI-1.86]	_ZTV27QAccessibleFactoryInterface [CXXABI-1.86]
_ZTV27QGraphicsSceneDragDropEvent [CXXABI-1.86]	_ZTV27QIconEngineFactoryInterface [CXXABI-1.86]
_ZTV29QInputContextFactoryInterface [CXXABI-1.86]	_ZTV30QGraphicsSceneContextMenuEvent [CXXABI-1.86]
_ZTV31QImageIOHandlerFactoryInterface [CXXABI-1.86]	_ZTV33QAccessibleBridgeFactoryInterface [CXXABI-1.86]
_ZTV5QDial [CXXABI-1.86]	_ZTV5QDrag [CXXABI-1.86]
_ZTV5QMenu [CXXABI-1.86]	_ZTV6QFrame [CXXABI-1.86]
_ZTV6QImage [CXXABI-1.86]	_ZTV6QLabel [CXXABI-1.86]
_ZTV6QMovie [CXXABI-1.86]	_ZTV6QSound [CXXABI-1.86]
_ZTV6QStyle [CXXABI-1.86]	_ZTV7QAction [CXXABI-1.86]
_ZTV7QBitmap [CXXABI-1.86]	_ZTV7QDialog [CXXABI-1.86]
_ZTV7QLayout [CXXABI-1.86]	_ZTV7QPixmap [CXXABI-1.86]
_ZTV7QSlider [CXXABI-1.86]	_ZTV7QTabBar [CXXABI-1.86]

_ZTV7QWidget [CXXABI-1.86]	_ZTV8QMenuBar [CXXABI-1.86]
_ZTV8QPicture [CXXABI-1.86]	_ZTV8QPrinter [CXXABI-1.86]
_ZTV8QSpinBox [CXXABI-1.86]	_ZTV8QStroker [CXXABI-1.86]
_ZTV8QToolBar [CXXABI-1.86]	_ZTV8QToolBox [CXXABI-1.86]
_ZTV9QCDEStyle [CXXABI-1.86]	_ZTV9QCheckBox [CXXABI-1.86]
_ZTV9QComboBox [CXXABI-1.86]	_ZTV9QDateEdit [CXXABI-1.86]
_ZTV9QDirModel [CXXABI-1.86]	_ZTV9QGroupBox [CXXABI-1.86]
_ZTV9QKeyEvent [CXXABI-1.86]	_ZTV9QLineEdit [CXXABI-1.86]
_ZTV9QListView [CXXABI-1.86]	_ZTV9QMenuItem [CXXABI-1.86]
_ZTV9QShortcut [CXXABI-1.86]	_ZTV9QSizeGrip [CXXABI-1.86]
_ZTV9QSplitter [CXXABI-1.86]	_ZTV9QTextEdit [CXXABI-1.86]
_ZTV9QTextList [CXXABI-1.86]	_ZTV9QTimeEdit [CXXABI-1.86]
_ZTV9QTreeView [CXXABI-1.86]	

18.5.2 Qt4 Organizers

18.5.2.1 Class data for QButtonGroup

The virtual table for the QButtonGroup class is described by Table 18-102

Table 18-102 Primary vtable for QButtonGroup

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QButtonGroup
vfunc[0]:	QButtonGroup::metaObject() const
vfunc[1]:	QButtonGroup::qt_metacast(char const*)
vfunc[2]:	QButtonGroup::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QButtonGroup::~QButtonGroup()
vfunc[4]:	QButtonGroup::~QButtonGroup()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QButtonGroup class is described by Table 18-103

Table 18-103 typeinfo for QButtonGroup

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QButtonGroup
basetype:	typeinfo for QObject

18.5.2.2 Class data for QGroupBox

The virtual table for the QGroupBox class is described by Table 18-104

Table 18-104 Primary vtable for QGroupBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGroupBox
vfunc[0]:	QGroupBox::metaObject() const
vfunc[1]:	QGroupBox::qt_metacast(char const*)
vfunc[2]:	QGroupBox::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QGroupBox::~~QGroupBox()
vfunc[4]:	QGroupBox::~~QGroupBox()
vfunc[5]:	QGroupBox::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QGroupBox::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const

vfunc[15]:	QGroupBox::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QGroupBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QGroupBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QGroupBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QGroupBox::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QGroupBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QGroupBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)

vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QGroupBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QGroupBox class is described by Table 18-105

Table 18-105 typeinfo for QGroupBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QGroupBox
basetype:	typeinfo for QWidget

18.5.2.3 Class data for QSplitter

The virtual table for the QSplitter class is described by Table 18-106

Table 18-106 Primary vtable for QSplitter

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSplitter
vfunc[0]:	QSplitter::metaObject() const

vfunc[1]:	QSplitter::qt_metacast(char const*)
vfunc[2]:	QSplitter::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSplitter::~QSplitter()
vfunc[4]:	QSplitter::~~QSplitter()
vfunc[5]:	QSplitter::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QSplitter::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QSplitter::sizeHint() const
vfunc[15]:	QSplitter::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)

vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QFrame::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QSplitter::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QSplitter::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QSplitter::createHandle()

The Run Time Type Information for the QSplitter class is described by Table 18-107

Table 18-107 typeinfo for QSplitter

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSplitter
basetype:	typeinfo for QFrame

18.5.2.4 Class data for QSplitterHandle

The virtual table for the QSplitterHandle class is described by Table 18-108

Table 18-108 Primary vtable for QSplitterHandle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSplitterHandle
vfunc[0]:	QSplitterHandle::metaObject() const
vfunc[1]:	QSplitterHandle::qt_metacast(char const*)
vfunc[2]:	QSplitterHandle::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QSplitterHandle::~~QSplitterHandle()
vfunc[4]:	NULL or QSplitterHandle::~~QSplitterHandle()
vfunc[5]:	QSplitterHandle::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QSplitterHandle::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const

vfunc[18]:	QSplitterHandle::mousePressEvent(QMouseEvent*)
vfunc[19]:	QSplitterHandle::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QSplitterHandle::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QSplitterHandle::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)

vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QSplitterHandle class is described by Table 18-109

Table 18-109 typeinfo for QSplitterHandle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSplitterHandle
basetype:	typeinfo for QWidget

18.5.2.5 Interfaces for Qt4 Organizers

An LSB conforming implementation shall provide the generic functions for Qt4 Organizers specified in Table 18-110, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-110 libQtGui - Qt4 Organizers Function Interfaces

_ZN12QButtonGroup11qt_metacalleN11QMetaObject4CallEiPPv [QtGui]	_ZN12QButtonGroup11qt_metacastEPKc [QtGui]
_ZN12QButtonGroup12removeButtonEP15QAbstractButton [QtGui]	_ZN12QButtonGroup12setExclusiveEb [QtGui]
_ZN12QButtonGroup13buttonClickedEP15QAbstractButton [QtGui]	_ZN12QButtonGroup13buttonClickedEi [QtGui]
_ZN12QButtonGroup13buttonPressedEP15QAbstractButton [QtXml]	_ZN12QButtonGroup13buttonPressedEi [QtXml]

_ZN12QButtonGroup14buttonReleasedEP15QAbstractButton [QtXml]	_ZN12QButtonGroup14buttonReleasedEi [QtXml]
_ZN12QButtonGroup5setIdEP15QAbstractButtoni [QtGui]	_ZN12QButtonGroup9addButtonEP15QAbstractButton [QtGui]
_ZN12QButtonGroup9addButtonEP15QAbstractButtoni [QtGui]	_ZN12QButtonGroupC1EP7QObject [QtGui]
_ZN12QButtonGroupC2EP7QObject [QtGui]	_ZN12QButtonGroupD0Ev [QtGui]
_ZN12QButtonGroupD1Ev [QtGui]	_ZN12QButtonGroupD2Ev [QtGui]
_ZN15QSplitterHandle10paintEventEP11QPaintEvent [QtGui]	_ZN15QSplitterHandle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN15QSplitterHandle11qt_metacastEPKc [QtGui]	_ZN15QSplitterHandle12moveSplitterEi [QtGui]
_ZN15QSplitterHandle14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN15QSplitterHandle14setOrientationEN2Qt11OrientationE [QtGui]
_ZN15QSplitterHandle15mousePressEventEP11QMouseEvent [QtGui]	_ZN15QSplitterHandle17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN15QSplitterHandle20closestLegalPositionEi [QtGui]	_ZN15QSplitterHandle5eventEP6QEvent [QtGui]
_ZN15QSplitterHandleC1EN2Qt11OrientationEP9QSplitter [QtGui]	_ZN15QSplitterHandleC2EN2Qt11OrientationEP9QSplitter [QtGui]
_ZN9QGroupBox10childEventEP11QChildEvent [QtGui]	_ZN9QGroupBox10paintEventEP11QPaintEvent [QtGui]
_ZN9QGroupBox10setCheckedEb [QtGui]	_ZN9QGroupBox11changeEventEP6QEvent [QtGui]
_ZN9QGroupBox11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN9QGroupBox11qt_metacastEPKc [QtGui]
_ZN9QGroupBox11resizeEventEP12QResizeEvent [QtGui]	_ZN9QGroupBox12focusInEventEP11QFocusEvent [QtGui]
_ZN9QGroupBox12setAlignmentEi [QtGui]	_ZN9QGroupBox12setCheckableEb [QtGui]
_ZN9QGroupBox14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN9QGroupBox15mousePressEventEP11QMouseEvent [QtGui]
_ZN9QGroupBox17mouseReleaseEventEP11QMouseEvent [QtGui]	_ZN9QGroupBox5eventEP6QEvent [QtGui]
_ZN9QGroupBox7clickedEb [QtXml]	_ZN9QGroupBox7setFlatEb [QtGui]
_ZN9QGroupBox7toggledEb [QtGui]	_ZN9QGroupBox8setTitleERK7QString [QtGui]
_ZN9QGroupBoxC1EP7QWidget [QtGui]	_ZN9QGroupBoxC1EP7QWidgetPKc [QtGui]

_ZN9QGroupBoxC1ERK7QStringP7QWidget [QtGui]	_ZN9QGroupBoxC1ERK7QStringP7QWidgetPKc [QtGui]
_ZN9QGroupBoxC2EP7QWidget [QtGui]	_ZN9QGroupBoxC2EP7QWidgetPKc [QtGui]
_ZN9QGroupBoxC2ERK7QStringP7QWidget [QtGui]	_ZN9QGroupBoxC2ERK7QStringP7QWidgetPKc [QtGui]
_ZN9QGroupBoxD0Ev [QtGui]	_ZN9QGroupBoxD1Ev [QtGui]
_ZN9QGroupBoxD2Ev [QtGui]	_ZN9QSplitter10childEventEP11QChildEvent [QtGui]
_ZN9QSplitter11changeEventEP6QEvent [QtGui]	_ZN9QSplitter11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QSplitter11qt_metacastEPKc [QtGui]	_ZN9QSplitter11resizeEventEP12QRResizeEvent [QtGui]
_ZN9QSplitter12createHandleEv [QtGui]	_ZN9QSplitter12insertWidgetEiP7QWidget [QtGui]
_ZN9QSplitter12moveSplitterEii [QtGui]	_ZN9QSplitter12restoreStateERK10QByteArray [QtGui]
_ZN9QSplitter13setResizeModeEP7QWidgetNS_10ResizeModeE [QtGui]	_ZN9QSplitter13setRubberBandEi [QtGui]
_ZN9QSplitter13splitterMovedEii [QtGui]	_ZN9QSplitter14setCollapsibleEib [QtGui]
_ZN9QSplitter14setHandleWidthEi [QtGui]	_ZN9QSplitter14setOrientationEN2Qt11OrientationE [QtGui]
_ZN9QSplitter15setOpaqueResizeEb [QtGui]	_ZN9QSplitter16setStretchFactorEii [QtGui]
_ZN9QSplitter20closestLegalPositionEii [QtGui]	_ZN9QSplitter22setChildrenCollapsibleEb [QtGui]
_ZN9QSplitter5eventEP6QEvent [QtGui]	_ZN9QSplitter7refreshEv [QtGui]
_ZN9QSplitter8setSizesERK5QListIiE [QtGui]	_ZN9QSplitter9addWidgetEP7QWidget [QtGui]
_ZN9QSplitterC1EN2Qt11OrientationEP7QWidget [QtGui]	_ZN9QSplitterC1EN2Qt11OrientationEP7QWidgetPKc [QtGui]
_ZN9QSplitterC1EP7QWidget [QtGui]	_ZN9QSplitterC1EP7QWidgetPKc [QtGui]
_ZN9QSplitterC2EN2Qt11OrientationEP7QWidget [QtGui]	_ZN9QSplitterC2EN2Qt11OrientationEP7QWidgetPKc [QtGui]
_ZN9QSplitterC2EP7QWidget [QtGui]	_ZN9QSplitterC2EP7QWidgetPKc [QtGui]
_ZN9QSplitterD0Ev [QtGui]	_ZN9QSplitterD1Ev [QtGui]
_ZN9QSplitterD2Ev [QtGui]	_ZNK12QButtonGroup10metaObjectEv [QtGui]

_Znk12QButtonGroup13checkedBut tonEv [QtGui]	_Znk12QButtonGroup2idEP15QAbs tractButton [QtGui]
_Znk12QButtonGroup6buttonEi [QtGui]	_Znk12QButtonGroup7buttonsEv [QtGui]
_Znk12QButtonGroup9checkedIdEv [QtGui]	_Znk12QButtonGroup9exclusiveEv [QtGui]
_Znk15QSplitterHandle10metaObje ctEv [QtGui]	_Znk15QSplitterHandle11orientatio nEv [QtGui]
_Znk15QSplitterHandle12opaqueRe sizeEv [QtGui]	_Znk15QSplitterHandle8sizeHintEv [QtGui]
_Znk15QSplitterHandle8splitterEv [QtGui]	_Znk9QGroupBox10metaObjectEv [QtGui]
_Znk9QGroupBox11isCheckableEv [QtGui]	_Znk9QGroupBox15minimumSizeH intEv [QtGui]
_Znk9QGroupBox5titleEv [QtGui]	_Znk9QGroupBox6isFlatEv [QtGui]
_Znk9QGroupBox9alignmentEv [QtGui]	_Znk9QGroupBox9isCheckedEv [QtGui]
_Znk9QSplitter10metaObjectEv [QtGui]	_Znk9QSplitter11handleWidthEv [QtGui]
_Znk9QSplitter11orientationEv [QtGui]	_Znk9QSplitter12opaqueResizeEv [QtGui]
_Znk9QSplitter13isCollapsibleEi [QtGui]	_Znk9QSplitter15minimumSizeHint Ev [QtGui]
_Znk9QSplitter19childrenCollapsibl eEv [QtGui]	_Znk9QSplitter5countEv [QtGui]
_Znk9QSplitter5sizesEv [QtGui]	_Znk9QSplitter6handleEi [QtGui]
_Znk9QSplitter6widgetEi [QtGui]	_Znk9QSplitter7indexOfEP7QWidg et [QtGui]
Znk9QSplitter8getRangeEiPiS0 [QtGui]	_Znk9QSplitter8sizeHintEv [QtGui]
_Znk9QSplitter9saveStateEv [QtGui]	_ZlsR11QTextStreamRK9QSplitter [QtGui]
_ZrsR11QTextStreamR9QSplitter [QtGui]	

18.5.3 Qt4 Events

18.5.3.1 Class data for QInputEvent

The virtual table for the QInputEvent class is described by Table 18-111

Table 18-111 Primary vtable for QInputEvent

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QInputEvent
vfunc[0]:	QInputEvent::~~QInputEvent()
vfunc[1]:	QInputEvent::~~QInputEvent()

The Run Time Type Information for the QInputEvent class is described by Table 18-112

Table 18-112 typeinfo for QInputEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QInputEvent
basetype:	typeinfo for QEvent

18.5.3.2 Class data for QMouseEvent

The virtual table for the QMouseEvent class is described by Table 18-113

Table 18-113 Primary vtable for QMouseEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMouseEvent
vfunc[0]:	QMouseEvent::~~QMouseEvent()
vfunc[1]:	QMouseEvent::~~QMouseEvent()

The Run Time Type Information for the QMouseEvent class is described by Table 18-114

Table 18-114 typeinfo for QMouseEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMouseEvent
basetype:	typeinfo for QInputEvent

18.5.3.3 Class data for QHoverEvent

The virtual table for the QHoverEvent class is described by Table 18-115

Table 18-115 Primary vtable for QHoverEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHoverEvent
vfunc[0]:	QHoverEvent::~~QHoverEvent()

vfunc[1]:	QHoverEvent::~~QHoverEvent()

The Run Time Type Information for the QHoverEvent class is described by Table 18-116

Table 18-116 typeinfo for QHoverEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHoverEvent
basetype:	typeinfo for QEvent

18.5.3.4 Class data for QWheelEvent

The virtual table for the QWheelEvent class is described by Table 18-117

Table 18-117 Primary vtable for QWheelEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWheelEvent
vfunc[0]:	QWheelEvent::~~QWheelEvent()
vfunc[1]:	QWheelEvent::~~QWheelEvent()

The Run Time Type Information for the QWheelEvent class is described by Table 18-118

Table 18-118 typeinfo for QWheelEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWheelEvent
basetype:	typeinfo for QInputEvent

18.5.3.5 Class data for QTabletEvent

The virtual table for the QTabletEvent class is described by Table 18-119

Table 18-119 Primary vtable for QTabletEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTabletEvent
vfunc[0]:	QTabletEvent::~~QTabletEvent()
vfunc[1]:	QTabletEvent::~~QTabletEvent()

The Run Time Type Information for the QTabletEvent class is described by Table 18-120

Table 18-120 typeinfo for QTabletEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTabletEvent
basetype:	typeinfo for QInputEvent

18.5.3.6 Class data for QKeyEvent

The virtual table for the QKeyEvent class is described by Table 18-121

Table 18-121 Primary vtable for QKeyEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QKeyEvent
vfunc[0]:	QKeyEvent::~~QKeyEvent()
vfunc[1]:	QKeyEvent::~~QKeyEvent()

The Run Time Type Information for the QKeyEvent class is described by Table 18-122

Table 18-122 typeinfo for QKeyEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QKeyEvent
basetype:	typeinfo for QInputEvent

18.5.3.7 Class data for QFocusEvent

The virtual table for the QFocusEvent class is described by Table 18-123

Table 18-123 Primary vtable for QFocusEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFocusEvent
vfunc[0]:	QFocusEvent::~~QFocusEvent()
vfunc[1]:	QFocusEvent::~~QFocusEvent()

The Run Time Type Information for the QFocusEvent class is described by Table 18-124

Table 18-124 typeinfo for QFocusEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFocusEvent

basetype:	typeinfo for QEvent

18.5.3.8 Class data for QPaintEvent

The virtual table for the QPaintEvent class is described by Table 18-125

Table 18-125 Primary vtable for QPaintEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPaintEvent
vfunc[0]:	QPaintEvent::~~QPaintEvent()
vfunc[1]:	QPaintEvent::~~QPaintEvent()

The Run Time Type Information for the QPaintEvent class is described by Table 18-126

Table 18-126 typeinfo for QPaintEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPaintEvent
basetype:	typeinfo for QEvent

18.5.3.9 Class data for QMoveEvent

The virtual table for the QMoveEvent class is described by Table 18-127

Table 18-127 Primary vtable for QMoveEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMoveEvent
vfunc[0]:	QMoveEvent::~~QMoveEvent()
vfunc[1]:	QMoveEvent::~~QMoveEvent()

The Run Time Type Information for the QMoveEvent class is described by Table 18-128

Table 18-128 typeinfo for QMoveEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMoveEvent
basetype:	typeinfo for QEvent

18.5.3.10 Class data for QResizeEvent

The virtual table for the QResizeEvent class is described by Table 18-129

Table 18-129 Primary vtable for QResizeEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QResizeEvent
vfunc[0]:	QResizeEvent::~~QResizeEvent()
vfunc[1]:	QResizeEvent::~~QResizeEvent()

The Run Time Type Information for the QResizeEvent class is described by Table 18-130

Table 18-130 typeinfo for QResizeEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QResizeEvent
basetype:	typeinfo for QEvent

18.5.3.11 Class data for QCloseEvent

The virtual table for the QCloseEvent class is described by Table 18-131

Table 18-131 Primary vtable for QCloseEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCloseEvent
vfunc[0]:	QCloseEvent::~~QCloseEvent()
vfunc[1]:	QCloseEvent::~~QCloseEvent()

The Run Time Type Information for the QCloseEvent class is described by Table 18-132

Table 18-132 typeinfo for QCloseEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QCloseEvent
basetype:	typeinfo for QEvent

18.5.3.12 Class data for QIconDragEvent

The virtual table for the QIconDragEvent class is described by Table 18-133

Table 18-133 Primary vtable for QIconDragEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QIconDragEvent

vfunc[0]:	QIconDragEvent::~~QIconDragEvent()
vfunc[1]:	QIconDragEvent::~~QIconDragEvent()

The Run Time Type Information for the QIconDragEvent class is described by Table 18-134

Table 18-134 typeinfo for QIconDragEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QIconDragEvent
basetype:	typeinfo for QEvent

18.5.3.13 Class data for QShowEvent

The virtual table for the QShowEvent class is described by Table 18-135

Table 18-135 Primary vtable for QShowEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QShowEvent
vfunc[0]:	QShowEvent::~~QShowEvent()
vfunc[1]:	QShowEvent::~~QShowEvent()

The Run Time Type Information for the QShowEvent class is described by Table 18-136

Table 18-136 typeinfo for QShowEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QShowEvent
basetype:	typeinfo for QEvent

18.5.3.14 Class data for QHideEvent

The virtual table for the QHideEvent class is described by Table 18-137

Table 18-137 Primary vtable for QHideEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHideEvent
vfunc[0]:	QHideEvent::~~QHideEvent()
vfunc[1]:	QHideEvent::~~QHideEvent()

The Run Time Type Information for the QHideEvent class is described by Table 18-138

Table 18-138 typeinfo for QHideEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHideEvent
basetype:	typeinfo for QEvent

18.5.3.15 Class data for QContextMenuEvent

The virtual table for the QContextMenuEvent class is described by Table 18-139

Table 18-139 Primary vtable for QContextMenuEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QContextMenuEvent
vfunc[0]:	QContextMenuEvent::~~QContextMenuEvent()
vfunc[1]:	QContextMenuEvent::~~QContextMenuEvent()

The Run Time Type Information for the QContextMenuEvent class is described by Table 18-140

Table 18-140 typeinfo for QContextMenuEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QContextMenuEvent
basetype:	typeinfo for QInputEvent

18.5.3.16 Class data for QInputMethodEvent

The virtual table for the QInputMethodEvent class is described by Table 18-141

Table 18-141 Primary vtable for QInputMethodEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QInputMethodEvent
vfunc[0]:	NULL or QInputMethodEvent::~~QInputMethodEvent()

vfunc[1]:	NULL or QInputMethodEvent::~QInputMethodEvent()

The Run Time Type Information for the QInputMethodEvent class is described by Table 18-142

Table 18-142 typeinfo for QInputMethodEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QInputMethodEvent
basetype:	typeinfo for QEvent

18.5.3.17 Class data for QHelpEvent

The virtual table for the QHelpEvent class is described by Table 18-143

Table 18-143 Primary vtable for QHelpEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHelpEvent
vfunc[0]:	QHelpEvent::~QHelpEvent()
vfunc[1]:	QHelpEvent::~QHelpEvent()

The Run Time Type Information for the QHelpEvent class is described by Table 18-144

Table 18-144 typeinfo for QHelpEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHelpEvent
basetype:	typeinfo for QEvent

18.5.3.18 Class data for QStatusTipEvent

The virtual table for the QStatusTipEvent class is described by Table 18-145

Table 18-145 Primary vtable for QStatusTipEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStatusTipEvent
vfunc[0]:	QStatusTipEvent::~QStatusTipEvent()

vfunc[1]:	QStatusTipEvent::~~QStatusTipEvent()

The Run Time Type Information for the QStatusTipEvent class is described by Table 18-146

Table 18-146 typeinfo for QStatusTipEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStatusTipEvent
basetype:	typeinfo for QEvent

18.5.3.19 Class data for QWhatsThisClickedEvent

The virtual table for the QWhatsThisClickedEvent class is described by Table 18-147

Table 18-147 Primary vtable for QWhatsThisClickedEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWhatsThisClickedEvent
vfunc[0]:	QWhatsThisClickedEvent::~~QWhats ThisClickedEvent()
vfunc[1]:	QWhatsThisClickedEvent::~~QWhats ThisClickedEvent()

The Run Time Type Information for the QWhatsThisClickedEvent class is described by Table 18-148

Table 18-148 typeinfo for QWhatsThisClickedEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWhatsThisClickedEvent
basetype:	typeinfo for QEvent

18.5.3.20 Class data for QActionEvent

The virtual table for the QActionEvent class is described by Table 18-149

Table 18-149 Primary vtable for QActionEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QActionEvent
vfunc[0]:	QActionEvent::~~QActionEvent()

vfunc[1]:	QActionEvent::~~QActionEvent()

The Run Time Type Information for the QActionEvent class is described by Table 18-150

Table 18-150 typeinfo for QActionEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QActionEvent
basetype:	typeinfo for QEvent

18.5.3.21 Class data for QFileOpenEvent

The virtual table for the QFileOpenEvent class is described by Table 18-151

Table 18-151 Primary vtable for QFileOpenEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFileOpenEvent
vfunc[0]:	QFileOpenEvent::~~QFileOpenEvent()
vfunc[1]:	QFileOpenEvent::~~QFileOpenEvent()

The Run Time Type Information for the QFileOpenEvent class is described by Table 18-152

Table 18-152 typeinfo for QFileOpenEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFileOpenEvent
basetype:	typeinfo for QEvent

18.5.3.22 Class data for QToolBarChangeEvent

The virtual table for the QToolBarChangeEvent class is described by Table 18-153

Table 18-153 Primary vtable for QToolBarChangeEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QToolBarChangeEvent
vfunc[0]:	QToolBarChangeEvent::~~QToolBarChangeEvent()
vfunc[1]:	QToolBarChangeEvent::~~QToolBarChangeEvent()

The Run Time Type Information for the `QToolBarChangeEvent` class is described by Table 18-154

Table 18-154 typeinfo for `QToolBarChangeEvent`

Base Vtable	vtable for <code>__cxxabiv1::__si_class_type_info</code>
Name	typeinfo name for <code>QToolBarChangeEvent</code>
basetype:	typeinfo for <code>QEvent</code>

18.5.3.23 Class data for `QShortcutEvent`

The virtual table for the `QShortcutEvent` class is described by Table 18-155

Table 18-155 Primary vtable for `QShortcutEvent`

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for <code>QShortcutEvent</code>
<code>vfunc[0]:</code>	<code>QShortcutEvent::~~QShortcutEvent()</code>
<code>vfunc[1]:</code>	<code>QShortcutEvent::~~QShortcutEvent()</code>

The Run Time Type Information for the `QShortcutEvent` class is described by Table 18-156

Table 18-156 typeinfo for `QShortcutEvent`

Base Vtable	vtable for <code>__cxxabiv1::__si_class_type_info</code>
Name	typeinfo name for <code>QShortcutEvent</code>
basetype:	typeinfo for <code>QEvent</code>

18.5.3.24 Class data for `QClipboardEvent`

The virtual table for the `QClipboardEvent` class is described by Table 18-157

Table 18-157 Primary vtable for `QClipboardEvent`

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for <code>QClipboardEvent</code>
<code>vfunc[0]:</code>	<code>QClipboardEvent::~~QClipboardEvent()</code>
<code>vfunc[1]:</code>	<code>QClipboardEvent::~~QClipboardEvent()</code>

The Run Time Type Information for the `QClipboardEvent` class is described by Table 18-158

Table 18-158 typeinfo for QClipboardEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QClipboardEvent
basetype:	typeinfo for QEvent

18.5.3.25 Class data for QWindowStateChangeEvent

The virtual table for the QWindowStateChangeEvent class is described by Table 18-159

Table 18-159 Primary vtable for QWindowStateChangeEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWindowStateChangeEvent
vfunc[0]:	QWindowStateChangeEvent::~~QWin dowStateChangeEvent()
vfunc[1]:	QWindowStateChangeEvent::~~QWin dowStateChangeEvent()

The Run Time Type Information for the QWindowStateChangeEvent class is described by Table 18-160

Table 18-160 typeinfo for QWindowStateChangeEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWindowStateChangeEvent
basetype:	typeinfo for QEvent

18.5.3.26 Class data for QMenuBarUpdatedEvent

The virtual table for the QMenuBarUpdatedEvent class is described by Table 18-161

Table 18-161 Primary vtable for QMenuBarUpdatedEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMenuBarUpdatedEvent
vfunc[0]:	NULL or QMenuBarUpdatedEvent::~~QMenub arUpdatedEvent()

vfunc[1]:	NULL or QMenuBarUpdatedEvent::~QMenuBarUpdatedEvent()

The Run Time Type Information for the QMenuBarUpdatedEvent class is described by Table 18-162

Table 18-162 typeinfo for QMenuBarUpdatedEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMenuBarUpdatedEvent
basetype:	typeinfo for QEvent

18.5.3.27 Interfaces for Qt4 Events

An LSB conforming implementation shall provide the generic functions for Qt4 Events specified in Table 18-163, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-163 libQtGui - Qt4 Events Function Interfaces

ZN10QHelpEventC1EN6QEvent4TypeERK6QPointS4 [QtGui]	_ZN10QHelpEventC2EN6QEvent4TypeERK6QPointS4_ [QtGui]
_ZN10QHelpEventD0Ev [QtGui]	_ZN10QHelpEventD1Ev [QtGui]
_ZN10QHelpEventD2Ev [QtGui]	_ZN10QHideEventC1Ev [QtGui]
_ZN10QHideEventC2Ev [QtGui]	_ZN10QHideEventD0Ev [QtGui]
_ZN10QHideEventD1Ev [QtGui]	_ZN10QHideEventD2Ev [QtGui]
ZN10QMoveEventC1ERK6QPointS2 [QtGui]	_ZN10QMoveEventC2ERK6QPointS2_ [QtGui]
_ZN10QMoveEventD0Ev [QtGui]	_ZN10QMoveEventD1Ev [QtGui]
_ZN10QMoveEventD2Ev [QtGui]	_ZN10QShowEventC1Ev [QtGui]
_ZN10QShowEventC2Ev [QtGui]	_ZN10QShowEventD0Ev [QtGui]
_ZN10QShowEventD1Ev [QtGui]	_ZN10QShowEventD2Ev [QtGui]
_ZN11QCloseEventC1Ev [QtGui]	_ZN11QCloseEventC2Ev [QtGui]
_ZN11QCloseEventD0Ev [QtGui]	_ZN11QCloseEventD1Ev [QtGui]
_ZN11QCloseEventD2Ev [QtGui]	_ZN11QFocusEvent6reasonEv [QtGui]
_ZN11QFocusEventC1EN6QEvent4TypeEN2Qt11FocusReasonE [QtGui]	_ZN11QFocusEventC2EN6QEvent4TypeEN2Qt11FocusReasonE [QtGui]
_ZN11QFocusEventD0Ev [QtGui]	_ZN11QFocusEventD1Ev [QtGui]
_ZN11QFocusEventD2Ev [QtGui]	_ZN11QHoverEventC1EN6QEvent4TypeERK6QPointS4_ [QtGui]

ZN11QHoverEventC2EN6QEvent4TypeERK6QPointS4 [QtGui]	_ZN11QHoverEventD0Ev [QtGui]
_ZN11QHoverEventD1Ev [QtGui]	_ZN11QHoverEventD2Ev [QtGui]
_ZN11QInputEventC1EN6QEvent4TypeE6QFlagsIN2Qt16KeyboardModifierEE [QtGui]	_ZN11QInputEventC2EN6QEvent4TypeE6QFlagsIN2Qt16KeyboardModifierEE [QtGui]
_ZN11QInputEventD0Ev [QtGui]	_ZN11QInputEventD1Ev [QtGui]
_ZN11QInputEventD2Ev [QtGui]	_ZN11QMouseEventC1EN6QEvent4TypeERK6QPointN2Qt11MouseButtonE6QFlagsIS6_ES7_INS5_16KeyboardModifierEE [QtGui]
_ZN11QMouseEventC1EN6QEvent4TypeERK6QPointS4_N2Qt11MouseButtonE6QFlagsIS6_ES7_INS5_16KeyboardModifierEE [QtGui]	_ZN11QMouseEventC1EN6QEvent4TypeERK6QPointS4_ii [QtGui]
_ZN11QMouseEventC1EN6QEvent4TypeERK6QPointii [QtGui]	_ZN11QMouseEventC2EN6QEvent4TypeERK6QPointN2Qt11MouseButtonE6QFlagsIS6_ES7_INS5_16KeyboardModifierEE [QtGui]
_ZN11QMouseEventC2EN6QEvent4TypeERK6QPointS4_N2Qt11MouseButtonE6QFlagsIS6_ES7_INS5_16KeyboardModifierEE [QtGui]	_ZN11QMouseEventC2EN6QEvent4TypeERK6QPointS4_ii [QtGui]
_ZN11QMouseEventC2EN6QEvent4TypeERK6QPointii [QtGui]	_ZN11QMouseEventD0Ev [QtGui]
_ZN11QMouseEventD1Ev [QtGui]	_ZN11QMouseEventD2Ev [QtGui]
_ZN11QPaintEventC1ERK5QRect [QtGui]	_ZN11QPaintEventC1ERK7QRegion [QtGui]
_ZN11QPaintEventC1ERK7QRegionRK5QRect [QtGui]	_ZN11QPaintEventC2ERK5QRect [QtGui]
_ZN11QPaintEventC2ERK7QRegion [QtGui]	_ZN11QPaintEventC2ERK7QRegionRK5QRect [QtGui]
_ZN11QPaintEventD0Ev [QtGui]	_ZN11QPaintEventD1Ev [QtGui]
_ZN11QPaintEventD2Ev [QtGui]	_ZN11QWheelEventC1ERK6QPointS2_i6QFlagsIN2Qt11MouseButtonEES3_INS4_16KeyboardModifierEENS4_11OrientationE [QtGui]
_ZN11QWheelEventC1ERK6QPointS2_iiN2Qt11OrientationE [QtGui]	_ZN11QWheelEventC1ERK6QPointi6QFlagsIN2Qt11MouseButtonEES3_INS4_16KeyboardModifierEENS4_11OrientationE [QtGui]
_ZN11QWheelEventC1ERK6QPointiiN2Qt11OrientationE [QtGui]	_ZN11QWheelEventC2ERK6QPointS2_i6QFlagsIN2Qt11MouseButtonEES3_INS4_16KeyboardModifierEENS4_11OrientationE [QtGui]

_ZN11QWheelEventC2ERK6QPointS2_iiN2Qt11OrientationE [QtGui]	_ZN11QWheelEventC2ERK6QPointi6QFlagsIN2Qt11MouseButtonEES3_IN54_16KeyboardModifierEENS4_11OrientationE [QtGui]
_ZN11QWheelEventC2ERK6QPointiiN2Qt11OrientationE [QtGui]	_ZN11QWheelEventD0Ev [QtGui]
_ZN11QWheelEventD1Ev [QtGui]	_ZN11QWheelEventD2Ev [QtGui]
ZN12QActionEventC1EiP7QActionS1 [QtGui]	_ZN12QActionEventC2EiP7QActionS1_ [QtGui]
_ZN12QActionEventD0Ev [QtGui]	_ZN12QActionEventD1Ev [QtGui]
_ZN12QActionEventD2Ev [QtGui]	_ZN12QResizeEventC1ERK5QSizeS2_ [QtGui]
ZN12QResizeEventC2ERK5QSizeS2 [QtGui]	_ZN12QResizeEventD0Ev [QtGui]
_ZN12QResizeEventD1Ev [QtGui]	_ZN12QResizeEventD2Ev [QtGui]
_ZN12QTabletEventC1EN6QEvent4TypeERK6QPointS4_RK7QPointFiddi6QFlagsIN2Qt16KeyboardModifierEEx [QtGui]	_ZN12QTabletEventC2EN6QEvent4TypeERK6QPointS4_RK7QPointFiddi6QFlagsIN2Qt16KeyboardModifierEEx [QtGui]
_ZN12QTabletEventD0Ev [QtGui]	_ZN12QTabletEventD1Ev [QtGui]
_ZN12QTabletEventD2Ev [QtGui]	_ZN14QFileOpenEventC1ERK7QString [QtGui]
_ZN14QFileOpenEventC2ERK7QString [QtGui]	_ZN14QFileOpenEventD0Ev [QtGui]
_ZN14QFileOpenEventD1Ev [QtGui]	_ZN14QFileOpenEventD2Ev [QtGui]
_ZN14QIconDragEventC1Ev [QtGui]	_ZN14QIconDragEventC2Ev [QtGui]
_ZN14QIconDragEventD0Ev [QtGui]	_ZN14QIconDragEventD1Ev [QtGui]
_ZN14QIconDragEventD2Ev [QtGui]	_ZN14QShortcutEventC1ERK12QKeySequenceib [QtGui]
_ZN14QShortcutEventC2ERK12QKeySequenceib [QtGui]	_ZN14QShortcutEventD0Ev [QtGui]
_ZN14QShortcutEventD1Ev [QtGui]	_ZN14QShortcutEventD2Ev [QtGui]
_ZN15QStatusTipEventC1ERK7QString [QtGui]	_ZN15QStatusTipEventC2ERK7QString [QtGui]
_ZN15QStatusTipEventD0Ev [QtGui]	_ZN15QStatusTipEventD1Ev [QtGui]
_ZN15QStatusTipEventD2Ev [QtGui]	_ZN17QContextMenuEventC1ENS_6ReasonERK6QPoint [QtGui]
_ZN17QContextMenuEventC1ENS_6ReasonERK6QPointS3_ [QtGui]	_ZN17QContextMenuEventC1ENS_6ReasonERK6QPointS3_i [QtGui]

_ZN17QContextMenuEventC1ENS_6ReasonERK6QPointi [QtGui]	_ZN17QContextMenuEventC2ENS_6ReasonERK6QPoint [QtGui]
_ZN17QContextMenuEventC2ENS_6ReasonERK6QPointS3_ [QtGui]	_ZN17QContextMenuEventC2ENS_6ReasonERK6QPointS3_i [QtGui]
_ZN17QContextMenuEventC2ENS_6ReasonERK6QPointi [QtGui]	_ZN17QContextMenuEventD0Ev [QtGui]
_ZN17QContextMenuEventD1Ev [QtGui]	_ZN17QContextMenuEventD2Ev [QtGui]
_ZN17QInputMethodEvent15setCommitStringERK7QStringii [QtGui]	_ZN17QInputMethodEventC1ERK7QStringRK5QListINS_9AttributeEE [QtGui]
ZN17QInputMethodEventC1ERKS [QtGui]	_ZN17QInputMethodEventC1Ev [QtGui]
_ZN17QInputMethodEventC2ERK7QStringRK5QListINS_9AttributeEE [QtGui]	_ZN17QInputMethodEventC2ERKS_ [QtGui]
_ZN17QInputMethodEventC2Ev [QtGui]	_ZN19QToolBarChangeEventC1Eb [LSB]
_ZN19QToolBarChangeEventC2Eb [LSB]	_ZN19QToolBarChangeEventD0Ev [QtGui]
_ZN19QToolBarChangeEventD1Ev [QtGui]	_ZN19QToolBarChangeEventD2Ev [QtGui]
_ZN20QMenuBarUpdatedEventC1EP8QMenuBar [QtGui]	_ZN20QMenuBarUpdatedEventC2EP8QMenuBar [QtGui]
_ZN22QWhatsThisClickedEventC1ERK7QString [QtGui]	_ZN22QWhatsThisClickedEventC2ERK7QString [QtGui]
_ZN22QWhatsThisClickedEventD0Ev [QtGui]	_ZN22QWhatsThisClickedEventD1Ev [QtGui]
_ZN22QWhatsThisClickedEventD2Ev [QtGui]	_ZN23QWindowStateChangeEventC1E6QFlagsIN2Qt11WindowStateEE [QtGui]
_ZN23QWindowStateChangeEventC1E6QFlagsIN2Qt11WindowStateEEb [QtGui]	_ZN23QWindowStateChangeEventC2E6QFlagsIN2Qt11WindowStateEE [QtGui]
_ZN23QWindowStateChangeEventC2E6QFlagsIN2Qt11WindowStateEEb [QtGui]	_ZN23QWindowStateChangeEventD0Ev [QtGui]
_ZN23QWindowStateChangeEventD1Ev [QtGui]	_ZN23QWindowStateChangeEventD2Ev [QtGui]
_ZN9QKeyEvent22createExtendedKeyEventEN6QEvent4TypeEi6QFlagsIN2Qt16KeyboardModifierEEjjRK7QStringbt [QtXml]	_ZN9QKeyEventC1EN6QEvent4TypeEi6QFlagsIN2Qt16KeyboardModifierEERK7QStringbt [QtGui]

_ZN9QKeyEventC2EN6QEvent4TypeEi6QFlagsIN2Qt16KeyboardModifiersEERK7QStringbt [QtGui]	_ZN9QKeyEventD0Ev [QtGui]
_ZN9QKeyEventD1Ev [QtGui]	_ZN9QKeyEventD2Ev [QtGui]
_ZNK11QFocusEvent6reasonEv [QtXml]	_ZNK17QContextMenuEvent5stateEv [QtGui]
_ZNK23QWindowStateChangeEvent10isOverrideEv [QtGui]	_ZNK9QKeyEvent14nativeScanCodeEv [QtXml]
_ZNK9QKeyEvent15nativeModifiersEv [QtXml]	_ZNK9QKeyEvent16nativeVirtualKeyEv [QtXml]
_ZNK9QKeyEvent7matchesEN12QKeyEventSequence11StandardKeyE [QtXml]	_ZNK9QKeyEvent9modifiersEv [QtGui]
_Zls6QDebugPK6QEvent [QtGui]	

18.5.4 Qt4 Fonts

18.5.4.1 Interfaces for Qt4 Fonts

An LSB conforming implementation shall provide the generic functions for Qt4 Fonts specified in Table 18-164, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-164 libQtGui - Qt4 Fonts Function Interfaces

_ZN12QFontMetricsC1ERK5QFont [QtGui]	_ZN12QFontMetricsC1ERK5QFontP12QPaintDevice [QtGui]
ZN12QFontMetricsC1ERKS [QtGui]	_ZN12QFontMetricsC2ERK5QFont [QtGui]
_ZN12QFontMetricsC2ERK5QFontP12QPaintDevice [QtGui]	_ZN12QFontMetricsC2ERKS_ [QtGui]
_ZN12QFontMetricsD1Ev [QtGui]	_ZN12QFontMetricsD2Ev [QtGui]
ZN12QFontMetricsaSERKS [QtGui]	_ZN12QFontMetricseqERKS_ [QtGui]
_ZN13QFontComboBox11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN13QFontComboBox11qt_metacastEPKc [QtXml]
_ZN13QFontComboBox14setCurrentFontERK5QFont [QtXml]	_ZN13QFontComboBox14setFontFiltersE6QFlagsINS_10FontFilterEE [QtXml]
_ZN13QFontComboBox16setWritingSystemEN13QFontDatabase13WritingSystemE [QtXml]	_ZN13QFontComboBox18currentFontChangedERK5QFont [QtXml]
_ZN13QFontComboBox5eventEP6QEvent [QtXml]	_ZN13QFontComboBoxC1EP7QWid get [QtXml]
_ZN13QFontComboBoxC2EP7QWid get [QtXml]	_ZN13QFontComboBoxD0Ev [QtXml]

_ZN13QFontComboBoxD1Ev [QtXml]	_ZN13QFontComboBoxD2Ev [QtXml]
ZN13QFontDatabase10pointSizesERK7QStringS2 [QtGui]	_ZN13QFontDatabase11smoothSizesERK7QStringS2_ [QtGui]
_ZN13QFontDatabase11styleStringERK5QFont [QtGui]	_ZN13QFontDatabase11styleStringERK9QFontInfo [QtGui]
_ZN13QFontDatabase13standardSizeEv [QtGui]	_ZN13QFontDatabase17writingSystemNameENS_13WritingSystemE [QtGui]
_ZN13QFontDatabase18addApplicationFontERK7QString [QtXml]	_ZN13QFontDatabase19writingSystemSampleENS_13WritingSystemE [QtGui]
_ZN13QFontDatabase21removeApplicationFontEi [QtXml]	_ZN13QFontDatabase23applicationFontFamiliesEi [QtXml]
_ZN13QFontDatabase25removeAllApplicationFontsEv [QtXml]	_ZN13QFontDatabase26addApplicationFontFromDataERK10QByteArray [QtXml]
_ZN13QFontDatabaseC1Ev [QtGui]	_ZN13QFontDatabaseC2Ev [QtGui]
_ZN13QFontMetricsFC1ERK12QFontMetrics [QtXml]	_ZN13QFontMetricsFC1ERK5QFont [QtGui]
_ZN13QFontMetricsFC1ERK5QFontP12QPaintDevice [QtGui]	_ZN13QFontMetricsFC1ERKS_ [QtGui]
_ZN13QFontMetricsFC2ERK12QFontMetrics [QtXml]	_ZN13QFontMetricsFC2ERK5QFont [QtGui]
_ZN13QFontMetricsFC2ERK5QFontP12QPaintDevice [QtGui]	_ZN13QFontMetricsFC2ERKS_ [QtGui]
_ZN13QFontMetricsFD1Ev [QtGui]	_ZN13QFontMetricsFD2Ev [QtGui]
_ZN13QFontMetricsFaSERK12QFontMetrics [QtXml]	_ZN13QFontMetricsFaSERKS_ [QtGui]
ZN13QFontMetricsFeqERKS [QtGui]	_ZN5QFont10fromStringERK7QString [QtGui]
_ZN5QFont10initializeEv [QtGui]	_ZN5QFont10setKerningEb [QtGui]
_ZN5QFont10setRawModeEb [QtGui]	_ZN5QFont10setRawNameERK7QString [QtGui]
_ZN5QFont10setStretchEi [QtGui]	_ZN5QFont10substituteERK7QString [QtGui]
_ZN5QFont11defaultFontEv [QtGui]	_ZN5QFont11setOverlineEb [QtGui]
_ZN5QFont11substitutesERK7QString [QtGui]	_ZN5QFont12setPixelSizeEi [QtGui]
_ZN5QFont12setPointSizeEi [QtGui]	_ZN5QFont12setStrikeOutEb [QtGui]

_ZN5QFont12setStyleHintENS_9StyleHintENS_13StyleStrategyE [QtGui]	_ZN5QFont12setUnderlineEb [QtGui]
_ZN5QFont13setFixedPitchEb [QtGui]	_ZN5QFont13setPointSizeFed [QtGui]
_ZN5QFont13substitutionsEv [QtGui]	_ZN5QFont14setDefaultFontERKS_ [QtGui]
_ZN5QFont15cacheStatisticsEv [QtGui]	_ZN5QFont16setStyleStrategyENS_13StyleStrategyE [QtGui]
_ZN5QFont17setPixelSizeFloatEd [QtGui]	_ZN5QFont18insertSubstitutionERK7QStringS2_ [QtGui]
_ZN5QFont18removeSubstitutionERK7QString [QtGui]	_ZN5QFont19insertSubstitutionsERK7QStringRK11QStringList [QtGui]
_ZN5QFont7cleanupEv [LSB]	_ZN5QFont8setStyleENS_5StyleE [QtGui]
_ZN5QFont9setFamilyERK7QString [QtGui]	_ZN5QFont9setWeightEi [QtGui]
_ZN5QFontC1ERK7QStringiib [QtGui]	_ZN5QFontC1ERKS_ [QtGui]
_ZN5QFontC1ERKS_P12QPaintDevice [QtGui]	_ZN5QFontC1Ev [QtGui]
_ZN5QFontC2ERK7QStringiib [QtGui]	_ZN5QFontC2ERKS_ [QtGui]
_ZN5QFontC2ERKS_P12QPaintDevice [QtGui]	_ZN5QFontC2Ev [QtGui]
_ZN5QFontD1Ev [QtGui]	_ZN5QFontD2Ev [QtGui]
ZN5QFontaSERKS [QtGui]	_ZN9QFontInfoC1ERK5QFont [QtGui]
ZN9QFontInfoC1ERKS [QtGui]	_ZN9QFontInfoC2ERK5QFont [QtGui]
ZN9QFontInfoC2ERKS [QtGui]	_ZN9QFontInfoD1Ev [QtGui]
_ZN9QFontInfoD2Ev [QtGui]	_ZN9QFontInfoaSERKS_ [QtGui]
_ZNK12QFontMetrics10elidedTextERK7QStringN2Qt13TextElideModeEi [QtXml]	_ZNK12QFontMetrics11leftBearingE5QChar [QtGui]
_ZNK12QFontMetrics11lineSpacingEv [QtGui]	_ZNK12QFontMetrics11overlinePosEv [QtGui]
_ZNK12QFontMetrics12boundingRectE5QChar [QtGui]	_ZNK12QFontMetrics12boundingRectERK5QRectiRK7QStringiPi [QtGui]
_ZNK12QFontMetrics12boundingRectERK7QString [QtGui]	_ZNK12QFontMetrics12rightBearingE5QChar [QtGui]
_ZNK12QFontMetrics12strikeOutPosEv [QtGui]	_ZNK12QFontMetrics12underlinePosEv [QtGui]

_Znk12QFontMetrics14minLeftBearingEv [QtGui]	_Znk12QFontMetrics15minRightBearingEv [QtGui]
_Znk12QFontMetrics16averageCharWidthEv [QtXml]	_Znk12QFontMetrics4sizeEiRK7QStringiPi [QtGui]
_Znk12QFontMetrics5widthE5QChar [QtGui]	_Znk12QFontMetrics5widthERK7QStringi [QtGui]
_Znk12QFontMetrics6ascentEv [QtGui]	_Znk12QFontMetrics6heightEv [QtGui]
_Znk12QFontMetrics6inFontE5QChar [QtGui]	_Znk12QFontMetrics7descentEv [QtGui]
_Znk12QFontMetrics7leadingEv [QtGui]	_Znk12QFontMetrics7xHeightEv [QtGui]
_Znk12QFontMetrics8maxWidthEv [QtGui]	_Znk12QFontMetrics9charWidthERK7QStringi [QtGui]
_Znk12QFontMetrics9lineWidthEv [QtGui]	_Znk12QFontMetricseqERKS_ [QtGui]
_Znk13QFontComboBox10metaObjectEv [QtXml]	_Znk13QFontComboBox11currentFontEv [QtXml]
_Znk13QFontComboBox11fontFiltersEv [QtXml]	_Znk13QFontComboBox13writingSystemEv [QtXml]
_Znk13QFontComboBox8sizeHintEv [QtXml]	_Znk13QFontDatabase10isScalableERK7QStringS2_ [QtGui]
Znk13QFontDatabase12isFixedPitchERK7QStringS2 [QtGui]	_Znk13QFontDatabase14writingSystemsERK7QString [QtXml]
_Znk13QFontDatabase14writingSystemsEv [QtGui]	_Znk13QFontDatabase16isBitmapScalableERK7QStringS2_ [QtGui]
Znk13QFontDatabase18isSmoothlyScalableERK7QStringS2 [QtGui]	_Znk13QFontDatabase4boldERK7QStringS2_ [QtGui]
_Znk13QFontDatabase4fontERK7QStringS2_i [QtGui]	_Znk13QFontDatabase6italicERK7QStringS2_ [QtGui]
_Znk13QFontDatabase6stylesERK7QString [QtGui]	_Znk13QFontDatabase6weightERK7QStringS2_ [QtGui]
_Znk13QFontDatabase8familiesENS_13WritingSystemE [QtGui]	_Znk13QFontMetricsF10elidedTextERK7QStringN2Qt13TextElideModeEdi [QtXml]
_Znk13QFontMetricsF11leftBearingE5QChar [QtGui]	_Znk13QFontMetricsF11lineSpacingEv [QtGui]
_Znk13QFontMetricsF11overlinePosEv [QtGui]	_Znk13QFontMetricsF12boundingRectE5QChar [QtGui]
_Znk13QFontMetricsF12boundingRectERK6QRectFiRK7QStringiPi [QtGui]	_Znk13QFontMetricsF12boundingRectERK7QString [QtGui]

_ZNK13QFontMetricsF12rightBearin gE5QChar [QtGui]	_ZNK13QFontMetricsF12strikeOutP osEv [QtGui]
_ZNK13QFontMetricsF12underlineP osEv [QtGui]	_ZNK13QFontMetricsF14minLeftBea ringEv [QtGui]
_ZNK13QFontMetricsF15minRightBe aringEv [QtGui]	_ZNK13QFontMetricsF16averageCha rWidthEv [QtXml]
_ZNK13QFontMetricsF4sizeEiRK7Q StringiPi [QtGui]	_ZNK13QFontMetricsF5widthE5QC har [QtGui]
_ZNK13QFontMetricsF5widthERK7 QString [QtGui]	_ZNK13QFontMetricsF6ascentEv [QtGui]
_ZNK13QFontMetricsF6heightEv [QtGui]	_ZNK13QFontMetricsF6inFontE5QC har [QtGui]
_ZNK13QFontMetricsF7descentEv [QtGui]	_ZNK13QFontMetricsF7leadingEv [QtGui]
_ZNK13QFontMetricsF7xHeightEv [QtGui]	_ZNK13QFontMetricsF8maxWidthE v [QtGui]
_ZNK13QFontMetricsF9lineWidthEv [QtGui]	_ZNK13QFontMetricsFreqERKS_ [QtGui]
_ZNK5QFont10exactMatchEv [QtGui]	_ZNK5QFont10fixedPitchEv [QtGui]
_ZNK5QFont10pointSizeFEv [QtGui]	_ZNK5QFont13defaultFamilyEv [QtGui]
_ZNK5QFont13styleStrategyEv [QtGui]	_ZNK5QFont14lastResortFontEv [QtGui]
_ZNK5QFont16lastResortFamilyEv [QtGui]	_ZNK5QFont3keyEv [QtGui]
_ZNK5QFont5styleEv [QtGui]	_ZNK5QFont6familyEv [QtGui]
_ZNK5QFont6handleEv [QtGui]	_ZNK5QFont6weightEv [QtGui]
_ZNK5QFont7kerningEv [QtGui]	_ZNK5QFont7rawModeEv [QtGui]
_ZNK5QFont7rawNameEv [QtGui]	_ZNK5QFont7resolveERKS_ [QtGui]
_ZNK5QFont7stretchEv [QtGui]	_ZNK5QFont8isCopyOfERKS_ [QtGui]
_ZNK5QFont8overlineEv [QtGui]	_ZNK5QFont8toStringEv [QtGui]
_ZNK5QFont9pixelSizeEv [QtGui]	_ZNK5QFont9pointSizeEv [QtGui]
_ZNK5QFont9strikeOutEv [QtGui]	_ZNK5QFont9styleHintEv [QtGui]
_ZNK5QFont9underlineEv [QtGui]	_ZNK5QFontcv8QVariantEv [QtGui]
ZNK5QFonteqERKS [QtGui]	_ZNK5QFontltERKS_ [QtGui]
ZNK5QFontneERKS [QtGui]	_ZNK9QFontInfo10exactMatchEv [QtGui]

_ZNK9QFontInfo10fixedPitchEv [QtGui]	_ZNK9QFontInfo10pointSizeFEv [QtGui]
_ZNK9QFontInfo5styleEv [QtGui]	_ZNK9QFontInfo6familyEv [QtGui]
_ZNK9QFontInfo6italicEv [QtGui]	_ZNK9QFontInfo6weightEv [QtGui]
_ZNK9QFontInfo7rawModeEv [QtGui]	_ZNK9QFontInfo8overlineEv [LSB]
_ZNK9QFontInfo9pixelSizeEv [QtGui]	_ZNK9QFontInfo9pointSizeEv [QtGui]
_ZNK9QFontInfo9strikeOutEv [LSB]	_ZNK9QFontInfo9styleHintEv [QtGui]
_ZNK9QFontInfo9underlineEv [LSB]	_ZlsR11QDataStreamRK5QFont [QtGui]
_ZrsR11QDataStreamR5QFont [QtGui]	

18.5.5 Qt4 Colors

18.5.5.1 Interfaces for Qt4 Colors

An LSB conforming implementation shall provide the generic functions for Qt4 Colors specified in Table 18-165, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-165 libQtGui - Qt4 Colors Function Interfaces

_ZN6QColor10colorNamesEv [QtGui]	_ZN6QColor10invalidateEv [LSB]
_ZN6QColor13setNamedColorERK7QString [QtGui]	_ZN6QColor6setHsvEiiii [QtGui]
_ZN6QColor6setRedEi [QtGui]	_ZN6QColor6setRgbEiiii [QtGui]
_ZN6QColor6setRgbEj [QtGui]	_ZN6QColor7fromHsvEiiii [QtGui]
_ZN6QColor7fromRgbEiiii [QtGui]	_ZN6QColor7fromRgbEj [QtGui]
_ZN6QColor7getCmykEPiS0_S0_S0_S0_ [QtGui]	_ZN6QColor7setBlueEi [QtGui]
_ZN6QColor7setCmykEiiii [QtGui]	_ZN6QColor7setHsvFEddddd [QtGui]
_ZN6QColor7setRedFE [QtGui]	_ZN6QColor7setRgbFEddddd [QtGui]
_ZN6QColor7setRgbaEj [QtGui]	_ZN6QColor8fromCmykEiiii [QtGui]
_ZN6QColor8fromHsvFEddddd [QtGui]	_ZN6QColor8fromRgbFEddddd [QtGui]
_ZN6QColor8fromRgbaEj [QtGui]	_ZN6QColor8getCmykFEPdS0_S0_S0_S0_ [QtGui]
_ZN6QColor8setAlphaEi [QtGui]	_ZN6QColor8setBlueFE [QtGui]

_ZN6QColor8setCmykFEddddd [QtGui]	_ZN6QColor8setGreenEi [QtGui]
_ZN6QColor9fromCmykFEddddd [QtGui]	_ZN6QColor9setAlphaFE [QtGui]
_ZN6QColor9setGreenFE [QtGui]	_ZN6QColorC1EN2Qt11GlobalColorE [QtGui]
_ZN6QColorC1ENS_4SpecE [QtXml]	_ZN6QColorC1Ej [QtGui]
_ZN6QColorC2EN2Qt11GlobalColorE [QtGui]	_ZN6QColorC2ENS_4SpecE [QtXml]
_ZN6QColorC2Ej [QtGui]	_ZN6QColoraSEN2Qt11GlobalColorE [QtGui]
ZN6QColoraSERKS [QtGui]	_ZN8QPalette13setColorGroupENS_10ColorGroupERK6QBrushS3_S3_S3_S3_S3_S3_S3 [QtGui]
_ZN8QPalette8setBrushENS_10ColorGroupENS_9ColorRoleERK6QBrush [QtGui]	_ZN8QPaletteC1EN2Qt11GlobalColorE [QtGui]
_ZN8QPaletteC1ERK11QColorGroupS2_S2_ [QtGui]	_ZN8QPaletteC1ERK6QBrushS2_S2_S2_S2_S2_S2_S2 [QtGui]
_ZN8QPaletteC1ERK6QColor [QtGui]	_ZN8QPaletteC1ERK6QColorS2_ [QtGui]
_ZN8QPaletteC1ERK6QColorS2_S2_S2_S2_S2_S2_ [QtGui]	_ZN8QPaletteC1ERKS_ [QtGui]
_ZN8QPaletteC1Ev [QtGui]	_ZN8QPaletteC2EN2Qt11GlobalColorE [QtGui]
_ZN8QPaletteC2ERK11QColorGroupS2_S2_ [QtGui]	_ZN8QPaletteC2ERK6QBrushS2_S2_S2_S2_S2_S2_S2 [QtGui]
_ZN8QPaletteC2ERK6QColor [QtGui]	_ZN8QPaletteC2ERK6QColorS2_ [QtGui]
_ZN8QPaletteC2ERK6QColorS2_S2_S2_S2_S2_S2_ [QtGui]	_ZN8QPaletteC2ERKS_ [QtGui]
_ZN8QPaletteC2Ev [QtGui]	_ZN8QPaletteD1Ev [QtGui]
_ZN8QPaletteD2Ev [QtGui]	_ZN8QPaletteaSERKS_ [QtGui]
_ZN9QColormap10initializeEv [QtGui]	_ZN9QColormap7cleanupEv [QtGui]
_ZN9QColormap8instanceEi [QtGui]	_ZN9QColormapC1ERKS_ [QtGui]
ZN9QColormapC2ERKS [QtGui]	_ZN9QColormapD1Ev [QtGui]
_ZN9QColormapD2Ev [QtGui]	_ZN9QColomapaSERKS_ [QtXml]
_ZNK11QColorGroupcv8QVariantEv [QtGui]	_ZNK11QColorGroupcv8QVariantEv [QtGui]

_Znk6QColor10saturationEv [QtGui]	_Znk6QColor11saturationFEv [QtGui]
_Znk6QColor3hueEv [QtGui]	_Znk6QColor3redEv [QtGui]
_Znk6QColor3rgbEv [QtGui]	_Znk6QColor4blueEv [QtGui]
_Znk6QColor4cyanEv [QtGui]	_Znk6QColor4darkEi [QtGui]
_Znk6QColor4hueFEv [QtGui]	_Znk6QColor4nameEv [QtGui]
_Znk6QColor4redFEv [QtGui]	_Znk6QColor4rgbaEv [QtGui]
_Znk6QColor5alphaEv [QtGui]	_Znk6QColor5blackEv [QtGui]
_Znk6QColor5blueFEv [QtGui]	_Znk6QColor5cyanFEv [QtGui]
_Znk6QColor5greenEv [QtGui]	_Znk6QColor5lightEi [QtGui]
_Znk6QColor5pixelEi [QtGui]	_Znk6QColor5toHsvEv [QtGui]
_Znk6QColor5toRgbEv [QtGui]	_Znk6QColor5valueEv [QtGui]
_Znk6QColor6alphaFEv [QtGui]	_Znk6QColor6blackFEv [QtGui]
_Znk6QColor6getHsvEPiS0_S0_S0_ [QtGui]	_Znk6QColor6getRgbEPiS0_S0_S0_ [QtGui]
_Znk6QColor6greenFEv [QtGui]	_Znk6QColor6toCmykEv [QtGui]
_Znk6QColor6valueFEv [QtGui]	_Znk6QColor6yellowEv [QtGui]
_Znk6QColor7getHsvFEPdS0_S0_S0_ [QtGui]	_Znk6QColor7getRgbFEPdS0_S0_S0_ [QtGui]
_Znk6QColor7magentaEv [QtGui]	_Znk6QColor7yellowFEv [QtGui]
_Znk6QColor8magentaFEv [QtGui]	_Znk6QColor9convertToENS_4Spec E [QtGui]
_Znk6QColorcv8QVariantEv [QtGui]	_Znk6QColoreqERKS_ [QtGui]
Znk6QColorneERKS [QtGui]	_Znk8QPalette10isBrushSetENS_10 ColorGroupENS_9ColorRoleE [QtXml]
_Znk8QPalette12serialNumberEv [QtGui]	_Znk8QPalette5brushENS_10Color GroupENS_9ColorRoleE [QtGui]
_Znk8QPalette7isEqualENS_10ColorGroupES0_ [QtGui]	_Znk8QPalette7resolveERKS_ [QtGui]
Znk8QPalette8isCopyOfERKS [QtGui]	_Znk8QPalettectv8QVariantEv [QtGui]
Znk8QPaletteeqERKS [QtGui]	_Znk9QColormap4modeEv [QtGui]
_Znk9QColormap4sizeEv [QtGui]	_Znk9QColormap5depthEv [QtGui]
_Znk9QColormap5pixelERK6QColor [QtGui]	_Znk9QColormap7colorAtEj [QtGui]
_Znk9QColormap8colormapEv [QtGui]	_Zls6QDebugRK6QColor [QtGui]

_ZlsR11QDataStreamRK11QColorGroup [QtGui]	_ZlsR11QDataStreamRK6QColor [QtGui]
_ZlsR11QDataStreamRK8QPalette [QtGui]	_ZrsR11QDataStreamR11QColorGroup [QtGui]
_ZrsR11QDataStreamR6QColor [QtGui]	_ZrsR11QDataStreamR8QPalette [QtGui]

18.5.6 Qt4 Accessibility

18.5.6.1 Class data for QAccessibleBridgeFactoryInterface

The virtual table for the QAccessibleBridgeFactoryInterface class is described by Table 18-166

Table 18-166 Primary vtable for QAccessibleBridgeFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessibleBridgeFactoryInterface
vfunc[0]:	NULL or QAccessibleBridgeFactoryInterface::~QAccessibleBridgeFactoryInterface()
vfunc[1]:	NULL or QAccessibleBridgeFactoryInterface::~QAccessibleBridgeFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QAccessibleBridgeFactoryInterface class is described by Table 18-167

Table 18-167 typeinfo for QAccessibleBridgeFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAccessibleBridgeFactoryInterface
basetype:	typeinfo for QFactoryInterface

18.5.6.2 Class data for QAccessibleBridgePlugin

The virtual table for the QAccessibleBridgePlugin class is described by Table 18-168

Table 18-168 Primary vtable for QAccessibleBridgePlugin

Base Offset	0
Virtual Base Offset	0

RTTI	typeinfo for QAccessibleBridgePlugin
vfunc[0]:	QAccessibleBridgePlugin::metaObject() const
vfunc[1]:	QAccessibleBridgePlugin::qt_metacast(char const*)
vfunc[2]:	QAccessibleBridgePlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAccessibleBridgePlugin::~QAccessibleBridgePlugin()
vfunc[4]:	QAccessibleBridgePlugin::~QAccessibleBridgePlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual

The Run Time Type Information for the QAccessibleBridgePlugin class is described by Table 18-169

Table 18-169 typeinfo for QAccessibleBridgePlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QAccessibleBridgePlugin	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QAccessibleBridgeFactoryInterface	
		2050

18.5.6.3 Class data for QAccessibleInterface

The virtual table for the QAccessibleInterface class is described by Table 18-170

Table 18-170 Primary vtable for QAccessibleInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessibleInterface
vfunc[0]:	NULL or QAccessibleInterface::~~QAccessibleInterface()
vfunc[1]:	NULL or QAccessibleInterface::~~QAccessibleInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual
vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual
vfunc[8]:	__cxa_pure_virtual
vfunc[9]:	__cxa_pure_virtual
vfunc[10]:	__cxa_pure_virtual
vfunc[11]:	__cxa_pure_virtual
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual

The Run Time Type Information for the QAccessibleInterface class is described by Table 18-171

Table 18-171 typeinfo for QAccessibleInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAccessibleInterface
basetype:	typeinfo for QAccessible

18.5.6.4 Class data for QAccessibleObject

The virtual table for the QAccessibleObject class is described by Table 18-172

Table 18-172 Primary vtable for QAccessibleObject

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessibleObject
vfunc[0]:	QAccessibleObject::~~QAccessibleObject()
vfunc[1]:	QAccessibleObject::~~QAccessibleObject()
vfunc[2]:	QAccessibleObject::isValid() const
vfunc[3]:	QAccessibleObject::object() const
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual
vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual
vfunc[8]:	__cxa_pure_virtual
vfunc[9]:	__cxa_pure_virtual
vfunc[10]:	QAccessibleObject::setText(QAccessible::Text, int, QString const&)
vfunc[11]:	QAccessibleObject::rect(int) const
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	QAccessibleObject::userActionCount(int) const
vfunc[15]:	QAccessibleObject::actionText(int, QAccessible::Text, int) const
vfunc[16]:	QAccessibleObject::doAction(int, int, QList<QVariant> const&)

The Run Time Type Information for the QAccessibleObject class is described by Table 18-173

Table 18-173 typeinfo for QAccessibleObject

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAccessibleObject
basetype:	typeinfo for QAccessibleInterface

18.5.6.5 Class data for QAccessibleApplication

The virtual table for the QAccessibleApplication class is described by Table 18-174

Table 18-174 Primary vtable for QAccessibleApplication

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessibleApplication
vfunc[0]:	NULL or QAccessibleApplication::~~QAccessibleApplication()
vfunc[1]:	NULL or QAccessibleApplication::~~QAccessibleApplication()
vfunc[2]:	QAccessibleObject::isValid() const
vfunc[3]:	QAccessibleObject::object() const
vfunc[4]:	QAccessibleApplication::childCount()) const
vfunc[5]:	QAccessibleApplication::indexOfChild(QAccessibleInterface const*) const
vfunc[6]:	QAccessibleApplication::relationTo(int, QAccessibleInterface const*, int) const
vfunc[7]:	QAccessibleApplication::childAt(int, int) const
vfunc[8]:	QAccessibleApplication::navigate(QAccessible::RelationFlag, int, QAccessibleInterface**) const
vfunc[9]:	QAccessibleApplication::text(QAccessible::Text, int) const
vfunc[10]:	QAccessibleObject::setText(QAccessible::Text, int, QString const&)
vfunc[11]:	QAccessibleObject::rect(int) const
vfunc[12]:	QAccessibleApplication::role(int) const
vfunc[13]:	QAccessibleApplication::state(int) const
vfunc[14]:	QAccessibleApplication::userActionCount(int) const
vfunc[15]:	QAccessibleApplication::actionText(int, QAccessible::Text, int) const
vfunc[16]:	QAccessibleApplication::doAction(int, int, QList<QVariant> const&)

The Run Time Type Information for the QAccessibleApplication class is described by Table 18-175

Table 18-175 typeinfo for QAccessibleApplication

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAccessibleApplication
basetype:	typeinfo for QAccessibleObject

18.5.6.6 Class data for QAccessibleWidget

The virtual table for the QAccessibleWidget class is described by Table 18-176

Table 18-176 Primary vtable for QAccessibleWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessibleWidget
vfunc[0]:	QAccessibleWidget::~QAccessibleWidget()
vfunc[1]:	QAccessibleWidget::~~QAccessibleWidget()
vfunc[2]:	QAccessibleObject::isValid() const
vfunc[3]:	QAccessibleObject::object() const
vfunc[4]:	QAccessibleWidget::childCount() const
vfunc[5]:	QAccessibleWidget::indexOfChild(QAccessibleInterface const*) const
vfunc[6]:	QAccessibleWidget::relationTo(int, QAccessibleInterface const*, int) const
vfunc[7]:	QAccessibleWidget::childAt(int, int) const
vfunc[8]:	QAccessibleWidget::navigate(QAccessible::RelationFlag, int, QAccessibleInterface**) const
vfunc[9]:	QAccessibleWidget::text(QAccessible::Text, int) const
vfunc[10]:	QAccessibleObject::setText(QAccessible::Text, int, QString const&)
vfunc[11]:	QAccessibleWidget::rect(int) const
vfunc[12]:	QAccessibleWidget::role(int) const
vfunc[13]:	QAccessibleWidget::state(int) const
vfunc[14]:	QAccessibleObject::userActionCount(int) const

vfunc[15]:	QAccessibleWidget::actionText(int, QAccessible::Text, int) const
vfunc[16]:	QAccessibleWidget::doAction(int, QList<QVariant> const&)

The Run Time Type Information for the QAccessibleWidget class is described by Table 18-177

Table 18-177 typeinfo for QAccessibleWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAccessibleWidget
basetype:	typeinfo for QAccessibleObject

18.5.6.7 Class data for QAccessibleFactoryInterface

The virtual table for the QAccessibleFactoryInterface class is described by Table 18-178

Table 18-178 Primary vtable for QAccessibleFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessibleFactoryInterface
vfunc[0]:	NULL or QAccessibleFactoryInterface::~~QAccessibleFactoryInterface()
vfunc[1]:	NULL or QAccessibleFactoryInterface::~~QAccessibleFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QAccessibleFactoryInterface class is described by Table 18-179

Table 18-179 typeinfo for QAccessibleFactoryInterface

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	
Name	typeinfo name for QAccessibleFactoryInterface	
flags:	0	

basetype:	typeinfo for QAccessible	2
basetype:	typeinfo for QFactoryInterface	2

18.5.6.8 Class data for QAccessiblePlugin

The virtual table for the QAccessiblePlugin class is described by Table 18-180

Table 18-180 Primary vtable for QAccessiblePlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAccessiblePlugin
vfunc[0]:	QAccessiblePlugin::metaObject() const
vfunc[1]:	QAccessiblePlugin::qt_metacast(char const*)
vfunc[2]:	QAccessiblePlugin::qt_metacall(QMe taObject::Call, int, void**)
vfunc[3]:	QAccessiblePlugin::~~QAccessiblePlu gin()
vfunc[4]:	QAccessiblePlugin::~~QAccessiblePlu gin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual

The Run Time Type Information for the QAccessiblePlugin class is described by Table 18-181

Table 18-181 typeinfo for QAccessiblePlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	

Name	typeinfo name for QAccessiblePlugin	
flags:	0	
basetype:	typeinfo for QObject	2
basetype:	typeinfo for QAccessibleFactoryInterface	2050

18.5.6.9 Interfaces for Qt4 Accessibility

An LSB conforming implementation shall provide the generic functions for Qt4 Accessibility specified in Table 18-182, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-182 libQtGui - Qt4 Accessibility Function Interfaces

_ZN11QAccessible10initializeEv [QtGui]	_ZN11QAccessible13removeFactoryEPFP20QAccessibleInterfaceRK7QStringP7QObjectE [QtGui]
_ZN11QAccessible13setRootObjectEP7QObject [QtGui]	_ZN11QAccessible14installFactoryEPFP20QAccessibleInterfaceRK7QStringP7QObjectE [QtGui]
_ZN11QAccessible19updateAccessibilityEP7QObjectiNS_5EventE [QtGui]	_ZN11QAccessible20installUpdateHandlerEPFvP7QObjectiNS_5EventEE [QtGui]
_ZN11QAccessible24installRootObjectHandlerEPFvP7QObjectE [QtGui]	_ZN11QAccessible24queryAccessibleInterfaceEP7QObject [QtGui]
_ZN11QAccessible7cleanupEv [QtGui]	_ZN11QAccessible8isActiveEv [QtGui]
_ZN17QAccessibleObject7setTextEN11QAccessible4TextEiRK7QString [QtGui]	_ZN17QAccessibleObject8doActionEiiRK5QListI8QVariantE [QtGui]
_ZN17QAccessibleObjectC1EP7QObject [QtGui]	_ZN17QAccessibleObjectC2EP7QObject [QtGui]
_ZN17QAccessibleObjectD0Ev [QtGui]	_ZN17QAccessibleObjectD1Ev [QtGui]
_ZN17QAccessibleObjectD2Ev [QtGui]	_ZN17QAccessiblePlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN17QAccessiblePlugin11qt_metacastEPKc [QtGui]	_ZN17QAccessiblePluginC1EP7QObject [QtGui]
_ZN17QAccessiblePluginC2EP7QObject [QtGui]	_ZN17QAccessiblePluginD0Ev [QtGui]
_ZN17QAccessiblePluginD1Ev [QtGui]	_ZN17QAccessiblePluginD2Ev [QtGui]

_ZN17QAccessibleWidget14setAcceleratorERK7QString [QtGui]	_ZN17QAccessibleWidget14setDescriptionERK7QString [QtGui]
_ZN17QAccessibleWidget20addControllingSignalERK7QString [QtGui]	_ZN17QAccessibleWidget7setHelpERK7QString [QtGui]
_ZN17QAccessibleWidget8doActionEiiRK5QListI8QVariantE [QtGui]	_ZN17QAccessibleWidget8setValueERK7QString [QtGui]
_ZN17QAccessibleWidgetC1EP7QWidgetN11QAccessible4RoleERK7QString [QtGui]	_ZN17QAccessibleWidgetC2EP7QWidgetN11QAccessible4RoleERK7QString [QtGui]
_ZN17QAccessibleWidgetD0Ev [QtGui]	_ZN17QAccessibleWidgetD1Ev [QtGui]
_ZN17QAccessibleWidgetD2Ev [QtGui]	_ZN20QAccessibleInterface12invokeMethodEN11QAccessible6MethodEiiRK5QListI8QVariantE [QtGui]
_ZN22QAccessibleApplication8doActionEiiRK5QListI8QVariantE [QtGui]	_ZN22QAccessibleApplicationC1Ev [QtGui]
_ZN22QAccessibleApplicationC2Ev [QtGui]	_ZN23QAccessibleBridgePlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN23QAccessibleBridgePlugin11qt_metacastEPKc [QtGui]	_ZN23QAccessibleBridgePluginC1EP7QObject [QtGui]
_ZN23QAccessibleBridgePluginC2EP7QObject [QtGui]	_ZN23QAccessibleBridgePluginD0Ev [QtGui]
_ZN23QAccessibleBridgePluginD1Ev [QtGui]	_ZN23QAccessibleBridgePluginD2Ev [QtGui]
_ZNK17QAccessibleObject10actionTextEiN11QAccessible4TextEi [QtGui]	_ZNK17QAccessibleObject15userActionCountEi [QtGui]
_ZNK17QAccessibleObject4rectEi [QtGui]	_ZNK17QAccessibleObject6objectEv [QtGui]
_ZNK17QAccessibleObject7isValidEv [QtGui]	_ZNK17QAccessiblePlugin10metaObjectEv [QtGui]
_ZNK17QAccessibleWidget10actionTextEiN11QAccessible4TextEi [QtGui]	_ZNK17QAccessibleWidget10childCountEv [QtGui]
_ZNK17QAccessibleWidget10relationToEiPK20QAccessibleInterfacei [QtGui]	_ZNK17QAccessibleWidget12indexOfChildEPK20QAccessibleInterface [QtGui]
_ZNK17QAccessibleWidget12parentObjectEv [QtGui]	_ZNK17QAccessibleWidget4rectEi [QtGui]
_ZNK17QAccessibleWidget4roleEi [QtGui]	_ZNK17QAccessibleWidget4textEN11QAccessible4TextEi [QtGui]
_ZNK17QAccessibleWidget5stateEi [QtGui]	_ZNK17QAccessibleWidget6widgetEv [QtGui]

_Znk17QAccessibleWidget7childAtEii [QtGui]	_Znk17QAccessibleWidget8navigateEN11QAccessible12RelationFlagEiPP20QAccessibleInterface [QtGui]
_Znk22QAccessibleApplication10actionTextEiN11QAccessible4TextEi [QtGui]	_Znk22QAccessibleApplication10childCountEv [QtGui]
_Znk22QAccessibleApplication10relationToEiPK20QAccessibleInterfacei [QtGui]	_Znk22QAccessibleApplication12indexOfChildEPK20QAccessibleInterface [QtGui]
_Znk22QAccessibleApplication15userActionCountEi [QtGui]	_Znk22QAccessibleApplication4roleEi [QtGui]
_Znk22QAccessibleApplication4textEN11QAccessible4TextEi [QtGui]	_Znk22QAccessibleApplication5stateEi [QtGui]
_Znk22QAccessibleApplication7childAtEii [QtGui]	_Znk22QAccessibleApplication8navigateEN11QAccessible12RelationFlagEiPP20QAccessibleInterface [QtGui]
_Znk23QAccessibleBridgePlugin10metaObjectEv [QtGui]	

18.5.7 Qt4 Validators

18.5.7.1 Class data for QValidator

The virtual table for the QValidator class is described by Table 18-183

Table 18-183 Primary vtable for QValidator

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QValidator
vfunc[0]:	QValidator::metaObject() const
vfunc[1]:	QValidator::qt_metacast(char const*)
vfunc[2]:	QValidator::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QValidator::~~QValidator()
vfunc[4]:	QValidator::~~QValidator()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	QValidator::fixup(QString&) const

The Run Time Type Information for the QValidator class is described by Table 18-184

Table 18-184 typeinfo for QValidator

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QValidator
basetype:	typeinfo for QObject

18.5.7.2 Class data for QIntValidator

The virtual table for the QIntValidator class is described by Table 18-185

Table 18-185 Primary vtable for QIntValidator

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QIntValidator
vfunc[0]:	QIntValidator::metaObject() const
vfunc[1]:	QIntValidator::qt_metacast(char const*)
vfunc[2]:	QIntValidator::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QIntValidator::~~QIntValidator()
vfunc[4]:	QIntValidator::~~QIntValidator()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QIntValidator::validate(QString&, int&) const
vfunc[13]:	QValidator::fixup(QString&) const

vfunc[14]:	QIntValidator::setRange(int, int)

The Run Time Type Information for the QIntValidator class is described by Table 18-186

Table 18-186 typeinfo for QIntValidator

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QIntValidator
basetype:	typeinfo for QValidator

18.5.7.3 Class data for QDoubleValidator

The virtual table for the QDoubleValidator class is described by Table 18-187

Table 18-187 Primary vtable for QDoubleValidator

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDoubleValidator
vfunc[0]:	QDoubleValidator::metaObject() const
vfunc[1]:	QDoubleValidator::qt_metacast(char const*)
vfunc[2]:	QDoubleValidator::qt_metacall(QMe taObject::Call, int, void**)
vfunc[3]:	QDoubleValidator::~~QDoubleValidat or()
vfunc[4]:	QDoubleValidator::~~QDoubleValidat or()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QDoubleValidator::validate(QString &, int&) const
vfunc[13]:	QValidator::fixup(QString&) const

vfunc[14]:	QDoubleValidator::setRange(double, double, int)

The Run Time Type Information for the QDoubleValidator class is described by Table 18-188

Table 18-188 typeinfo for QDoubleValidator

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDoubleValidator
basetype:	typeinfo for QValidator

18.5.7.4 Class data for QRegExpValidator

The virtual table for the QRegExpValidator class is described by Table 18-189

Table 18-189 Primary vtable for QRegExpValidator

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QRegExpValidator
vfunc[0]:	QRegExpValidator::metaObject() const
vfunc[1]:	QRegExpValidator::qt_metacast(char const*)
vfunc[2]:	QRegExpValidator::qt_metacall(QMe taObject::Call, int, void**)
vfunc[3]:	QRegExpValidator::~~QRegExpValida tor()
vfunc[4]:	QRegExpValidator::~~QRegExpValida tor()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QRegExpValidator::validate(QString &, int&) const
vfunc[13]:	QValidator::fixup(QString&) const

The Run Time Type Information for the QRegExpValidator class is described by Table 18-190

Table 18-190 typeinfo for QRegExpValidator

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QRegExpValidator
basetype:	typeinfo for QValidator

18.5.7.5 Interfaces for Qt4 Validators

An LSB conforming implementation shall provide the generic functions for Qt4 Validators specified in Table 18-191, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-191 libQtGui - Qt4 Validators Function Interfaces

_ZN10QValidator11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN10QValidator11qt_metacastEPKc [QtGui]
_ZN10QValidatorC1EP7QObject [QtGui]	_ZN10QValidatorC1EP7QObjectPKc [QtGui]
_ZN10QValidatorC2EP7QObject [QtGui]	_ZN10QValidatorC2EP7QObjectPKc [QtGui]
_ZN10QValidatorD0Ev [QtGui]	_ZN10QValidatorD1Ev [QtGui]
_ZN10QValidatorD2Ev [QtGui]	_ZN13QIntValidator11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN13QIntValidator11qt_metacastEPKc [QtGui]	_ZN13QIntValidator6setTopEi [QtGui]
_ZN13QIntValidator8setRangeEii [QtGui]	_ZN13QIntValidator9setBottomEi [QtGui]
_ZN13QIntValidatorC1EP7QObject [QtGui]	_ZN13QIntValidatorC1EP7QObjectPKc [QtGui]
_ZN13QIntValidatorC1EiiP7QObject [QtGui]	_ZN13QIntValidatorC1EiiP7QObjectPKc [QtGui]
_ZN13QIntValidatorC2EP7QObject [QtGui]	_ZN13QIntValidatorC2EP7QObjectPKc [QtGui]
_ZN13QIntValidatorC2EiiP7QObject [QtGui]	_ZN13QIntValidatorC2EiiP7QObjectPKc [QtGui]
_ZN13QIntValidatorD0Ev [QtGui]	_ZN13QIntValidatorD1Ev [QtGui]
_ZN13QIntValidatorD2Ev [QtGui]	_ZN16QDoubleValidator11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN16QDoubleValidator11qt_metacastEPKc [QtGui]	_ZN16QDoubleValidator11setDecimalsEi [QtGui]

_ZN16QDoubleValidator6setTopEd [QtGui]	_ZN16QDoubleValidator8setRangeE ddi [QtGui]
_ZN16QDoubleValidator9setBottom Ed [QtGui]	_ZN16QDoubleValidatorC1EP7QObj ect [QtGui]
_ZN16QDoubleValidatorC1EP7QObj ectPKc [QtGui]	_ZN16QDoubleValidatorC1EddiP7Q Object [QtGui]
_ZN16QDoubleValidatorC1EddiP7Q ObjectPKc [QtGui]	_ZN16QDoubleValidatorC2EP7QObj ect [QtGui]
_ZN16QDoubleValidatorC2EP7QObj ectPKc [QtGui]	_ZN16QDoubleValidatorC2EddiP7Q Object [QtGui]
_ZN16QDoubleValidatorC2EddiP7Q ObjectPKc [QtGui]	_ZN16QDoubleValidatorD0Ev [QtGui]
_ZN16QDoubleValidatorD1Ev [QtGui]	_ZN16QDoubleValidatorD2Ev [QtGui]
_ZN16QRegExpValidator11qt_metac allEN11QMetaObject4CalleiPPv [QtGui]	_ZN16QRegExpValidator11qt_metac astEPKc [QtGui]
_ZN16QRegExpValidator9setRegExp ERK7QRegExp [QtGui]	_ZN16QRegExpValidatorC1EP7QOb ject [QtGui]
_ZN16QRegExpValidatorC1EP7QOb jectPKc [QtGui]	_ZN16QRegExpValidatorC1ERK7QR egExpP7QObject [QtGui]
_ZN16QRegExpValidatorC1ERK7QR egExpP7QObjectPKc [QtGui]	_ZN16QRegExpValidatorC2EP7QOb ject [QtGui]
_ZN16QRegExpValidatorC2EP7QOb jectPKc [QtGui]	_ZN16QRegExpValidatorC2ERK7QR egExpP7QObject [QtGui]
_ZN16QRegExpValidatorC2ERK7QR egExpP7QObjectPKc [QtGui]	_ZN16QRegExpValidatorD0Ev [QtGui]
_ZN16QRegExpValidatorD1Ev [QtGui]	_ZN16QRegExpValidatorD2Ev [QtGui]
_ZNK10QValidator10metaObjectEv [QtGui]	_ZNK10QValidator5fixupER7QStrin g [QtGui]
_ZNK13QIntValidator10metaObjectE v [QtGui]	_ZNK13QIntValidator8validateER7Q StringRi [QtGui]
_ZNK16QDoubleValidator10metaOb jectEv [QtGui]	_ZNK16QDoubleValidator8validateE R7QStringRi [QtGui]
_ZNK16QRegExpValidator10metaOb jectEv [QtGui]	_ZNK16QRegExpValidator8validate ER7QStringRi [QtGui]

18.5.8 Qt4 Plugins

18.5.8.1 Class data for QImageIOPlugin

The virtual table for the QImageIOPlugin class is described by Table 18-192

Table 18-192 Primary vtable for QImageIOPlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QImageIOPlugin
vfunc[0]:	QImageIOPlugin::metaObject() const
vfunc[1]:	QImageIOPlugin::qt_metacast(char const*)
vfunc[2]:	QImageIOPlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QImageIOPlugin::~~QImageIOPlugin()
vfunc[4]:	QImageIOPlugin::~~QImageIOPlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual

The Run Time Type Information for the QImageIOPlugin class is described by Table 18-193

Table 18-193 typeinfo for QImageIOPlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QImageIOPlugin	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QImageIOHandlerFactoryInterface	2050

18.5.8.2 Class data for QIconEnginePlugin

The virtual table for the QIconEnginePlugin class is described by Table 18-194

Table 18-194 Primary vtable for QIconEnginePlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QIconEnginePlugin
vfunc[0]:	QIconEnginePlugin::metaObject() const
vfunc[1]:	QIconEnginePlugin::qt_metacast(char const*)
vfunc[2]:	QIconEnginePlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QIconEnginePlugin::~QIconEnginePlugin()
vfunc[4]:	QIconEnginePlugin::~QIconEnginePlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual

The Run Time Type Information for the QIconEnginePlugin class is described by Table 18-195

Table 18-195 typeinfo for QIconEnginePlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QIconEnginePlugin	
flags:	0	
basetype:	typeinfo for QObject	

basetype:	typeinfo for QIconEngineFactoryInt erface	2050

18.5.8.3 Class data for QPictureFormatPlugin

The virtual table for the QPictureFormatPlugin class is described by Table 18-196

Table 18-196 Primary vtable for QPictureFormatPlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPictureFormatPlugin
vfunc[0]:	QPictureFormatPlugin::metaObject() const
vfunc[1]:	QPictureFormatPlugin::qt_metacast(c har const*)
vfunc[2]:	QPictureFormatPlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QPictureFormatPlugin::~QPictureFor matPlugin()
vfunc[4]:	QPictureFormatPlugin::~~QPictureFor matPlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	QPictureFormatPlugin::loadPicture(QString const&, QString const&, QPicture*)
vfunc[14]:	QPictureFormatPlugin::savePicture(QString const&, QString const&, QPicture const&)
vfunc[15]:	__cxa_pure_virtual

The Run Time Type Information for the QPictureFormatPlugin class is described by Table 18-197

Table 18-197 typeinfo for QPictureFormatPlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	
Name	typeinfo name for QPictureFormatPlugin	
flags:	0	
basetype:	typeinfo for QObject	2
basetype:	typeinfo for QPictureFormatInterfac e	2050

18.5.8.4 Class data for QStylePlugin

The virtual table for the QStylePlugin class is described by Table 18-198

Table 18-198 Primary vtable for QStylePlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStylePlugin
vfunc[0]:	QStylePlugin::metaObject() const
vfunc[1]:	QStylePlugin::qt_metacast(char const*)
vfunc[2]:	QStylePlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QStylePlugin::~~QStylePlugin()
vfunc[4]:	QStylePlugin::~~QStylePlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual

The Run Time Type Information for the QStylePlugin class is described by Table 18-199

Table 18-199 typeinfo for QStylePlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QStylePlugin	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QStyleFactoryInterface	2050

18.5.8.5 Class data for QInputContextPlugin

The virtual table for the QInputContextPlugin class is described by Table 18-200

Table 18-200 Primary vtable for QInputContextPlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QInputContextPlugin
vfunc[0]:	QInputContextPlugin::metaObject() const
vfunc[1]:	QInputContextPlugin::qt_metacast(c har const*)
vfunc[2]:	QInputContextPlugin::qt_metacall(Q MetaObject::Call, int, void**)
vfunc[3]:	QInputContextPlugin::~~QInputConte xtPlugin()
vfunc[4]:	QInputContextPlugin::~~QInputConte xtPlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual

vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual

The Run Time Type Information for the QInputContextPlugin class is described by Table 18-201

Table 18-201 typeinfo for QInputContextPlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QInputContextPlugin	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QInputContextFactoryI nterface	2050

18.5.8.6 Interfaces for Qt4 Plugins

An LSB conforming implementation shall provide the generic functions for Qt4 Plugins specified in Table 18-202, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-202 libQtGui - Qt4 Plugins Function Interfaces

_ZN12QStylePlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN12QStylePlugin11qt_metacastEPKc [QtGui]
_ZN12QStylePluginC1EP7QObject [QtGui]	_ZN12QStylePluginC2EP7QObject [QtGui]
_ZN12QStylePluginD0Ev [QtGui]	_ZN12QStylePluginD1Ev [QtGui]
_ZN12QStylePluginD2Ev [QtGui]	_ZN14QImageIOPlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN14QImageIOPlugin11qt_metacastEPKc [QtGui]	_ZN14QImageIOPluginC1EP7QObject [QtGui]
_ZN14QImageIOPluginC2EP7QObject [QtGui]	_ZN14QImageIOPluginD0Ev [QtGui]
_ZN14QImageIOPluginD1Ev [QtGui]	_ZN14QImageIOPluginD2Ev [QtGui]
_ZN17QIconEnginePlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN17QIconEnginePlugin11qt_metacastEPKc [QtGui]
_ZN17QIconEnginePluginC1EP7QObject [QtGui]	_ZN17QIconEnginePluginC2EP7QObject [QtGui]

_ZN17QIconEnginePluginD0Ev [QtGui]	_ZN17QIconEnginePluginD1Ev [QtGui]
_ZN17QIconEnginePluginD2Ev [QtGui]	_ZN19QInputContextPlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN19QInputContextPlugin11qt_metacastEPKc [QtGui]	_ZN19QInputContextPluginC1EP7QObject [QtGui]
_ZN19QInputContextPluginC2EP7QObject [QtGui]	_ZN19QInputContextPluginD0Ev [QtGui]
_ZN19QInputContextPluginD1Ev [QtGui]	_ZN19QInputContextPluginD2Ev [QtGui]
_ZN20QPictureFormatPlugin11loadPictureERK7QStringS2_P8QPicture [QtGui]	_ZN20QPictureFormatPlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN20QPictureFormatPlugin11qt_metacastEPKc [QtGui]	_ZN20QPictureFormatPlugin11savePictureERK7QStringS2_RK8QPicture [QtGui]
_ZN20QPictureFormatPluginC1EP7QObject [QtGui]	_ZN20QPictureFormatPluginC2EP7QObject [QtGui]
_ZN20QPictureFormatPluginD0Ev [QtGui]	_ZN20QPictureFormatPluginD1Ev [QtGui]
_ZN20QPictureFormatPluginD2Ev [QtGui]	_ZNK12QStylePlugin10metaObjectEv [QtGui]
_ZNK14QImageIOPlugin10metaObjectEv [QtGui]	_ZNK17QIconEnginePlugin10metaObjectEv [QtGui]
_ZNK19QInputContextPlugin10metaObjectEv [QtGui]	_ZNK20QPictureFormatPlugin10metaObjectEv [QtGui]

18.5.9 Qt4 Models and Views

18.5.9.1 Class data for QAbstractTextDocumentLayout

The virtual table for the QAbstractTextDocumentLayout class is described by Table 18-203

Table 18-203 Primary vtable for QAbstractTextDocumentLayout

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractTextDocumentLayout
vfunc[0]:	QAbstractTextDocumentLayout::metaObject() const
vfunc[1]:	QAbstractTextDocumentLayout::qt_metacast(char const*)

vfunc[2]:	QAbstractTextDocumentLayout::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractTextDocumentLayout::~QAbstractTextDocumentLayout()
vfunc[4]:	QAbstractTextDocumentLayout::~QAbstractTextDocumentLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	__cxa_pure_virtual
vfunc[19]:	QAbstractTextDocumentLayout::resizeInlineObject(QTextInlineObject, int, QTextFormat const&)
vfunc[20]:	QAbstractTextDocumentLayout::positionInlineObject(QTextInlineObject, int, QTextFormat const&)
vfunc[21]:	QAbstractTextDocumentLayout::drawInlineObject(QPainter*, QRectF const&, QTextInlineObject, int, QTextFormat const&)

The Run Time Type Information for the QAbstractTextDocumentLayout class is described by Table 18-204

Table 18-204 typeinfo for QAbstractTextDocumentLayout

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractTextDocumentLayout

basetype:	typeinfo for QObject

18.5.9.2 Class data for QAbstractPageSetupDialog

The virtual table for the QAbstractPageSetupDialog class is described by Table 18-205

Table 18-205 Primary vtable for QAbstractPageSetupDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractPageSetupDialog
vfunc[0]:	QDialog::metaObject() const
vfunc[1]:	QDialog::qt_metacast(char const*)
vfunc[2]:	QDialog::qt_metacall(QMetaObject:: Call, int, void**)
vfunc[3]:	NULL or QAbstractPageSetupDialog::~~QAbstr actPageSetupDialog()
vfunc[4]:	NULL or QAbstractPageSetupDialog::~~QAbstr actPageSetupDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouse Event*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMou seEvent*)

vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const

vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()
vfunc[57]:	__cxa_pure_virtual

The Run Time Type Information for the QAbstractPageSetupDialog class is described by Table 18-206

Table 18-206 typeinfo for QAbstractPageSetupDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractPageSetupDialog
basetype:	typeinfo for QDialog

18.5.9.3 Class data for QAbstractPrintDialog

The virtual table for the QAbstractPrintDialog class is described by Table 18-207

Table 18-207 Primary vtable for QAbstractPrintDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractPrintDialog
vfunc[0]:	QDialog::metaObject() const
vfunc[1]:	QDialog::qt_metacast(char const*)
vfunc[2]:	QDialog::qt_metacall(QMetaObject::Call, int, void**)

vfunc[3]:	NULL or QAbstractPrintDialog::~QAbstractPrintDialog()
vfunc[4]:	NULL or QAbstractPrintDialog::~QAbstractPrintDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouse Event*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMou seEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMous eEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent *)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEve nt*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent *)
vfunc[26]:	QWidget::focusOutEvent(QFocusEve nt*)
vfunc[27]:	QWidget::enterEvent(QEvent*)

vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)

vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()
vfunc[57]:	__cxa_pure_virtual

The Run Time Type Information for the QAbstractPrintDialog class is described by Table 18-208

Table 18-208 typeinfo for QAbstractPrintDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractPrintDialog
basetype:	typeinfo for QDialog

18.5.9.4 Class data for QItemSelectionModel

The virtual table for the QItemSelectionModel class is described by Table 18-209

Table 18-209 Primary vtable for QItemSelectionModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QItemSelectionModel
vfunc[0]:	QItemSelectionModel::metaObject() const
vfunc[1]:	QItemSelectionModel::qt_metacast(c har const*)
vfunc[2]:	QItemSelectionModel::qt_metacall(Q MetaObject::Call, int, void**)
vfunc[3]:	QItemSelectionModel::~~QItemSelecti onModel()
vfunc[4]:	QItemSelectionModel::~~QItemSelecti onModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QItemSelectionModel::select(QModelIndex const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[13]:	QItemSelectionModel::select(QItemSelection const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[14]:	QItemSelectionModel::clear()
vfunc[15]:	QItemSelectionModel::reset()

The Run Time Type Information for the QItemSelectionModel class is described by Table 18-210

Table 18-210 typeinfo for QItemSelectionModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QItemSelectionModel
basetype:	typeinfo for QObject

18.5.9.5 Class data for QAbstractScrollArea

The virtual table for the QAbstractScrollArea class is described by Table 18-211

Table 18-211 Primary vtable for QAbstractScrollArea

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractScrollArea
vfunc[0]:	QAbstractScrollArea::metaObject() const
vfunc[1]:	QAbstractScrollArea::qt_metacast(char const*)
vfunc[2]:	QAbstractScrollArea::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractScrollArea::~QAbstractScrollArea()
vfunc[4]:	QAbstractScrollArea::~QAbstractScrollArea()
vfunc[5]:	QAbstractScrollArea::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)

vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractScrollArea::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractScrollArea::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractScrollArea::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractScrollArea::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractScrollArea::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractScrollArea::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractScrollArea::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)

vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractScrollArea::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractScrollArea::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractScrollArea::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractScrollArea::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractScrollArea::viewportEvent(QEvent*)
vfunc[55]:	QAbstractScrollArea::scrollContentsBy(int, int)

The Run Time Type Information for the QAbstractScrollArea class is described by Table 18-212

Table 18-212 typeinfo for QAbstractScrollArea

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractScrollArea
basetype:	typeinfo for QFrame

18.5.9.6 Class data for QAbstractSpinBox

The virtual table for the QAbstractSpinBox class is described by Table 18-213

Table 18-213 Primary vtable for QAbstractSpinBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractSpinBox
vfunc[0]:	QAbstractSpinBox::metaObject() const
vfunc[1]:	QAbstractSpinBox::qt_metacast(char const*)
vfunc[2]:	QAbstractSpinBox::qt_metacall(QMe taObject::Call, int, void**)
vfunc[3]:	QAbstractSpinBox::~~QAbstractSpinB ox()
vfunc[4]:	QAbstractSpinBox::~~QAbstractSpinB ox()
vfunc[5]:	QAbstractSpinBox::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSpinBox::timerEvent(QTim erEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractSpinBox::sizeHint() const
vfunc[15]:	QAbstractSpinBox::minimumSizeHin t() const
vfunc[16]:	QWidget::heightForWidth(int) const

vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractSpinBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractSpinBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractSpinBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractSpinBox::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSpinBox::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractSpinBox::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractSpinBox::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractSpinBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractSpinBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractSpinBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QAbstractSpinBox::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractSpinBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)

vfunc[40]:	QAbstractSpinBox::showEvent(QShowEvent*)
vfunc[41]:	QAbstractSpinBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSpinBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractSpinBox::validate(QString&, int&) const
vfunc[55]:	QAbstractSpinBox::fixup(QString&) const
vfunc[56]:	QAbstractSpinBox::stepBy(int)
vfunc[57]:	QAbstractSpinBox::clear()
vfunc[58]:	QAbstractSpinBox::stepEnabled() const

The Run Time Type Information for the QAbstractSpinBox class is described by Table 18-214

Table 18-214 typeinfo for QAbstractSpinBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractSpinBox
basetype:	typeinfo for QWidget

18.5.9.7 Class data for QAbstractSlider

The virtual table for the QAbstractSlider class is described by Table 18-215

Table 18-215 Primary vtable for QAbstractSlider

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractSlider
vfunc[0]:	QAbstractSlider::metaObject() const
vfunc[1]:	QAbstractSlider::qt_metacast(char const*)
vfunc[2]:	QAbstractSlider::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractSlider::~~QAbstractSlider()
vfunc[4]:	QAbstractSlider::~~QAbstractSlider()
vfunc[5]:	QAbstractSlider::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSlider::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)

vfunc[22]:	QAbstractSlider::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSlider::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSlider::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)

vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractSlider::sliderChange(QAbstractSlider::SliderChange)

The Run Time Type Information for the QAbstractSlider class is described by Table 18-216

Table 18-216 typeinfo for QAbstractSlider

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractSlider
basetype:	typeinfo for QWidget

18.5.9.8 Class data for QAbstractItemDelegate

The virtual table for the QAbstractItemDelegate class is described by Table 18-217

Table 18-217 Primary vtable for QAbstractItemDelegate

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractItemDelegate
vfunc[0]:	QAbstractItemDelegate::metaObject() const
vfunc[1]:	QAbstractItemDelegate::qt_metacast(char const*)
vfunc[2]:	QAbstractItemDelegate::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractItemDelegate::~QAbstractItemDelegate()
vfunc[4]:	QAbstractItemDelegate::~QAbstractItemDelegate()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	QAbstractItemDelegate::createEditor(QWidget*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[15]:	QAbstractItemDelegate::setEditorData(QWidget*, QModelIndex const&) const
vfunc[16]:	QAbstractItemDelegate::setModelData(QWidget*, QAbstractItemModel*, QModelIndex const&) const
vfunc[17]:	QAbstractItemDelegate::updateEditorGeometry(QWidget*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[18]:	QAbstractItemDelegate::editorEvent(QEvent*, QAbstractItemModel*, QStyleOptionViewItem const&, QModelIndex const&)

The Run Time Type Information for the QAbstractItemDelegate class is described by Table 18-218

Table 18-218 typeinfo for QAbstractItemDelegate

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractItemDelegate
basetype:	typeinfo for QObject

18.5.9.9 Class data for QAbstractItemView

The virtual table for the QAbstractItemView class is described by Table 18-219

Table 18-219 Primary vtable for QAbstractItemView

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QAbstractItemView
vfunc[0]:	QAbstractItemView::metaObject() const
vfunc[1]:	QAbstractItemView::qt_metacast(char const*)
vfunc[2]:	QAbstractItemView::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractItemView::~~QAbstractItemView()
vfunc[4]:	QAbstractItemView::~~QAbstractItemView()
vfunc[5]:	QAbstractItemView::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractItemView::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractItemView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractItemView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractItemView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractItemView::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)

vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractScrollArea::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractItemView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractItemView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractItemView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractItemView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const

vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QAbstractScrollArea::scrollContentsBy(int, int)
vfunc[56]:	QAbstractItemView::setModel(QAbstractItemModel*)
vfunc[57]:	QAbstractItemView::setSelectionMode(QItemSelectionModel*)
vfunc[58]:	QAbstractItemView::keyboardSearch(QString const&)
vfunc[59]:	__cxa_pure_virtual
vfunc[60]:	__cxa_pure_virtual
vfunc[61]:	__cxa_pure_virtual
vfunc[62]:	QAbstractItemView::sizeHintForRow(int) const
vfunc[63]:	QAbstractItemView::sizeHintForColumn(int) const
vfunc[64]:	QAbstractItemView::reset()
vfunc[65]:	QAbstractItemView::setRootIndex(QModelIndex const&)
vfunc[66]:	QAbstractItemView::doItemsLayout()
vfunc[67]:	QAbstractItemView::selectAll()
vfunc[68]:	QAbstractItemView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QAbstractItemView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QAbstractItemView::rowsAboutToBeRemoved(QModelIndex const&, int, int)

vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)
vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QAbstractItemView::updateGeometries()
vfunc[76]:	QAbstractItemView::verticalScrollBarAction(int)
vfunc[77]:	QAbstractItemView::horizontalScrollBarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollBarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollBarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)
vfunc[83]:	__cxa_pure_virtual
vfunc[84]:	__cxa_pure_virtual
vfunc[85]:	__cxa_pure_virtual
vfunc[86]:	__cxa_pure_virtual
vfunc[87]:	__cxa_pure_virtual
vfunc[88]:	__cxa_pure_virtual
vfunc[89]:	QAbstractItemView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelIndex const&, QAbstractItemView::EditTrigger, QEvent*)

vfunc[91]:	QAbstractItemView::selectionComm and(QModelIndex const&, QEvent const*) const
vfunc[92]:	QAbstractItemView::startDrag(QFlag s<Qt::DropAction>)
vfunc[93]:	QAbstractItemView::viewOptions() const

The Run Time Type Information for the QAbstractItemView class is described by Table 18-220

Table 18-220 typeinfo for QAbstractItemView

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractItemView
basetype:	typeinfo for QAbstractScrollArea

18.5.9.10 Class data for QStringListModel

The virtual table for the QStringListModel class is described by Table 18-221

Table 18-221 Primary vtable for QStringListModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStringListModel
vfunc[0]:	QStringListModel::metaObject() const
vfunc[1]:	QStringListModel::qt_metacast(char const*)
vfunc[2]:	QStringListModel::qt_metacall(QMet aObject::Call, int, void**)
vfunc[3]:	NULL or QStringListModel::~~QStringListMod el()
vfunc[4]:	NULL or QStringListModel::~~QStringListMod el()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)

vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractListModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QAbstractListModel::parent(QModelIndex const&) const
vfunc[14]:	QStringListModel::rowCount(QModelIndex const&) const
vfunc[15]:	QAbstractListModel::columnCount(QModelIndex const&) const
vfunc[16]:	QAbstractListModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QStringListModel::data(QModelIndex const&, int) const
vfunc[18]:	QStringListModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QAbstractItemModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QAbstractItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractListModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QStringListModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QAbstractItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QStringListModel::removeRows(int, int, QModelIndex const&)

vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QStringListModel::flags(QModelIndex const&) const
vfunc[34]:	QStringListModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()

The Run Time Type Information for the QStringListModel class is described by Table 18-222

Table 18-222 typeinfo for QStringListModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStringListModel
basetype:	typeinfo for QAbstractListModel

18.5.9.11 Class data for QDirModel

The virtual table for the QDirModel class is described by Table 18-223

Table 18-223 Primary vtable for QDirModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDirModel
vfunc[0]:	QDirModel::metaObject() const
vfunc[1]:	QDirModel::qt_metacast(char const*)
vfunc[2]:	QDirModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDirModel::~~QDirModel()
vfunc[4]:	QDirModel::~~QDirModel()

vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QDirModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QDirModel::parent(QModelIndex const&) const
vfunc[14]:	QDirModel::rowCount(QModelIndex const&) const
vfunc[15]:	QDirModel::columnCount(QModelIndex const&) const
vfunc[16]:	QDirModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QDirModel::data(QModelIndex const&, int) const
vfunc[18]:	QDirModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QDirModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QAbstractItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QDirModel::mimeTypes() const
vfunc[24]:	QDirModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QDirModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QDirModel::supportedDropActions() const

vfunc[27]:	QAbstractItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QAbstractItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QDirModel::flags(QModelIndex const&) const
vfunc[34]:	QDirModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()

The Run Time Type Information for the QDirModel class is described by Table 18-224

Table 18-224 typeinfo for QDirModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDirModel
basetype:	typeinfo for QAbstractItemModel

18.5.9.12 Class data for QStandardItemModel

The virtual table for the QStandardItemModel class is described by Table 18-225

Table 18-225 Primary vtable for QStandardItemModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStandardItemModel

vfunc[0]:	QStandardItemModel::metaObject() const
vfunc[1]:	QStandardItemModel::qt_metacast(char const*)
vfunc[2]:	QStandardItemModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QStandardItemModel::~~QStandardItemModel()
vfunc[4]:	QStandardItemModel::~~QStandardItemModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QStandardItemModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QStandardItemModel::parent(QModelIndex const&) const
vfunc[14]:	QStandardItemModel::rowCount(QModelIndex const&) const
vfunc[15]:	QStandardItemModel::columnCount(QModelIndex const&) const
vfunc[16]:	QStandardItemModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QStandardItemModel::data(QModelIndex const&, int) const
vfunc[18]:	QStandardItemModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QStandardItemModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QStandardItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const

vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractItemModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QStandardItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QStandardItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QStandardItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QStandardItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QStandardItemModel::flags(QModelIndex const&) const
vfunc[34]:	QAbstractItemModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()

The Run Time Type Information for the QStandardItemModel class is described by Table 18-226

Table 18-226 typeinfo for QStandardItemModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStandardItemModel
basetype:	typeinfo for QAbstractItemModel

18.5.9.13 Class data for QAbstractProxyModel

The virtual table for the QAbstractProxyModel class is described by Table 18-227

Table 18-227 Primary vtable for QAbstractProxyModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractProxyModel
vfunc[0]:	QAbstractProxyModel::metaObject() const
vfunc[1]:	QAbstractProxyModel::qt_metacast(c har const*)
vfunc[2]:	QAbstractProxyModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractProxyModel::~~QAbstractPr oxyModel()
vfunc[4]:	QAbstractProxyModel::~~QAbstractPr oxyModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	QAbstractItemModel::hasChildren(Q ModelIndex const&) const
vfunc[17]:	__cxa_pure_virtual

vfunc[18]:	QAbstractItemModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QAbstractItemModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QAbstractItemModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractItemModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QAbstractItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QAbstractItemModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QAbstractItemModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QAbstractItemModel::flags(QModelIndex const&) const
vfunc[34]:	QAbstractItemModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const

vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractProxyModel::submit()
vfunc[39]:	QAbstractProxyModel::revert()
vfunc[40]:	QAbstractProxyModel::setSourceModel(QAbstractItemModel*)
vfunc[41]:	__cxa_pure_virtual
vfunc[42]:	__cxa_pure_virtual
vfunc[43]:	QAbstractProxyModel::mapSelectionToSource(QItemSelection const&) const
vfunc[44]:	QAbstractProxyModel::mapSelectionFromSource(QItemSelection const&) const

The Run Time Type Information for the QAbstractProxyModel class is described by Table 18-228

Table 18-228 typeinfo for QAbstractProxyModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractProxyModel
basetype:	typeinfo for QAbstractItemModel

18.5.9.14 Class data for QSortFilterProxyModel

The virtual table for the QSortFilterProxyModel class is described by Table 18-229

Table 18-229 Primary vtable for QSortFilterProxyModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSortFilterProxyModel
vfunc[0]:	QSortFilterProxyModel::metaObject() const
vfunc[1]:	QSortFilterProxyModel::qt_metacast(char const*)
vfunc[2]:	QSortFilterProxyModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSortFilterProxyModel::~~QSortFilterProxyModel()
vfunc[4]:	QSortFilterProxyModel::~~QSortFilterProxyModel()

vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QSortFilterProxyModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QSortFilterProxyModel::parent(QModelIndex const&) const
vfunc[14]:	QSortFilterProxyModel::rowCount(QModelIndex const&) const
vfunc[15]:	QSortFilterProxyModel::columnCount(QModelIndex const&) const
vfunc[16]:	QSortFilterProxyModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QSortFilterProxyModel::data(QModelIndex const&, int) const
vfunc[18]:	QSortFilterProxyModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QSortFilterProxyModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QSortFilterProxyModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QSortFilterProxyModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QSortFilterProxyModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)

vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QSortFilterProxyModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QSortFilterProxyModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QSortFilterProxyModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QSortFilterProxyModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QSortFilterProxyModel::fetchMore(QModelIndex const&)
vfunc[32]:	QSortFilterProxyModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QSortFilterProxyModel::flags(QModelIndex const&) const
vfunc[34]:	QSortFilterProxyModel::sort(int, Qt::SortOrder)
vfunc[35]:	QSortFilterProxyModel::buddy(QModelIndex const&) const
vfunc[36]:	QSortFilterProxyModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QSortFilterProxyModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractProxyModel::submit()
vfunc[39]:	QAbstractProxyModel::revert()
vfunc[40]:	QSortFilterProxyModel::setSourceModel(QAbstractItemModel*)
vfunc[41]:	QSortFilterProxyModel::mapToSource(QModelIndex const&) const
vfunc[42]:	QSortFilterProxyModel::mapFromSource(QModelIndex const&) const
vfunc[43]:	QSortFilterProxyModel::mapSelectionToSource(QItemSelection const&) const
vfunc[44]:	QSortFilterProxyModel::mapSelectionFromSource(QItemSelection const&) const
vfunc[45]:	QSortFilterProxyModel::filterAcceptsRow(int, QModelIndex const&) const

vfunc[46]:	QSortFilterProxyModel::filterAcceptsColumn(int, QModelIndex const&) const
vfunc[47]:	QSortFilterProxyModel::lessThan(QModelIndex const&, QModelIndex const&) const

The Run Time Type Information for the QSortFilterProxyModel class is described by Table 18-230

Table 18-230 typeinfo for QSortFilterProxyModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSortFilterProxyModel
basetype:	typeinfo for QAbstractProxyModel

18.5.9.15 Class data for QItemDelegate

The virtual table for the QItemDelegate class is described by Table 18-231

Table 18-231 Primary vtable for QItemDelegate

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QItemDelegate
vfunc[0]:	QItemDelegate::metaObject() const
vfunc[1]:	QItemDelegate::qt_metacast(char const*)
vfunc[2]:	QItemDelegate::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QItemDelegate::~QItemDelegate()
vfunc[4]:	QItemDelegate::~QItemDelegate()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QItemDelegate::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QItemDelegate::paint(QPainter*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[13]:	QItemDelegate::sizeHint(QStyleOpti onViewItem const&, QModelIndex const&) const
vfunc[14]:	QItemDelegate::createEditor(QWidge t*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[15]:	QItemDelegate::setEditorData(QWid get*, QModelIndex const&) const
vfunc[16]:	QItemDelegate::setModelData(QWid get*, QAbstractItemModel*, QModelIndex const&) const
vfunc[17]:	QItemDelegate::updateEditorGeomet ry(QWidget*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[18]:	QItemDelegate::editorEvent(QEvent* , QAbstractItemModel*, QStyleOptionViewItem const&, QModelIndex const&)
vfunc[19]:	QItemDelegate::drawDisplay(QPaint er*, QStyleOptionViewItem const&, QRect const&, QString const&) const
vfunc[20]:	QItemDelegate::drawDecoration(QPa inter*, QStyleOptionViewItem const&, QRect const&, QPixmap const&) const
vfunc[21]:	QItemDelegate::drawFocus(QPainter *, QStyleOptionViewItem const&, QRect const&) const
vfunc[22]:	QItemDelegate::drawCheck(QPainter *, QStyleOptionViewItem const&, QRect const&, Qt::CheckState) const

The Run Time Type Information for the QItemDelegate class is described by Table 18-232

Table 18-232 typeinfo for QItemDelegate

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QItemDelegate
basetype:	typeinfo for QAbstractItemDelegate

18.5.9.16 Class data for QTableView

The virtual table for the QTableView class is described by Table 18-233

Table 18-233 Primary vtable for QTableView

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTableView
vfunc[0]:	QTableView::metaObject() const
vfunc[1]:	QTableView::qt_metacast(char const*)
vfunc[2]:	QTableView::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTableView::~QTableView()
vfunc[4]:	QTableView::~QTableView()
vfunc[5]:	QAbstractItemView::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTableView::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractItemView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractItemView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractItemView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractItemView::mouseMoveEvent(QMouseEvent*)

vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTableView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractItemView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractItemView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractItemView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractItemView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)

vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QTableView::scrollContentsBy(int, int)
vfunc[56]:	QTableView::setModel(QAbstractItemModel*)
vfunc[57]:	QTableView::setSelectionModel(QItemSelectionModel*)
vfunc[58]:	QAbstractItemView::keyboardSearch(QString const&)
vfunc[59]:	QTableView::visualRect(QModelIndex const&) const
vfunc[60]:	QTableView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QTableView::indexAt(QPoint const&) const
vfunc[62]:	QTableView::sizeHintForRow(int) const
vfunc[63]:	QTableView::sizeHintForColumn(int) const
vfunc[64]:	QAbstractItemView::reset()
vfunc[65]:	QTableView::setRootIndex(QModelIndex const&)
vfunc[66]:	QAbstractItemView::doItemsLayout()
vfunc[67]:	QAbstractItemView::selectAll()
vfunc[68]:	QAbstractItemView::dataChanged(QModelIndex const&, QModelIndex const&)

vfunc[69]:	QAbstractItemView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QAbstractItemView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)
vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QTableView::updateGeometries()
vfunc[76]:	QTableView::verticalScrollbarAction(int)
vfunc[77]:	QTableView::horizontalScrollbarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollbarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollbarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)
vfunc[83]:	QTableView::moveCursor(QAbstractItemView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QTableView::horizontalOffset() const
vfunc[85]:	QTableView::verticalOffset() const
vfunc[86]:	QTableView::isIndexHidden(QModelIndex const&) const
vfunc[87]:	QTableView::setSelection(QRect const&, QFlags<QItemSelectionModel::SelectionFlag>)

vfunc[88]:	QTableView::visualRegionForSelection(QItemSelection const&) const
vfunc[89]:	QTableView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelIndex const&, QAbstractItemView::EditTrigger, QEvent*)
vfunc[91]:	QAbstractItemView::selectionCommand(QModelIndex const&, QEvent const*) const
vfunc[92]:	QAbstractItemView::startDrag(QFlags<Qt::DropAction>)
vfunc[93]:	QTableView::viewOptions() const

The Run Time Type Information for the QTableView class is described by Table 18-234

Table 18-234 typeinfo for QTableView

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTableView
basetype:	typeinfo for QAbstractItemView

18.5.9.17 Class data for QTableWidgetItem

The virtual table for the QTableWidgetItem class is described by Table 18-235

Table 18-235 Primary vtable for QTableWidgetItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTableWidgetItem
vfunc[0]:	QTableWidgetItem::~~QTableWidgetItem()
vfunc[1]:	QTableWidgetItem::~~QTableWidgetItem()
vfunc[2]:	QTableWidgetItem::clone() const
vfunc[3]:	QTableWidgetItem::data(int) const
vfunc[4]:	QTableWidgetItem::setData(int, QVariant const&)
vfunc[5]:	QTableWidgetItem::operator<(QTableWidgetItem const&) const
vfunc[6]:	QTableWidgetItem::read(QDataStream&)

vfunc[7]:	QTableWidgetItem::write(QDataStream&) const

The Run Time Type Information for the QTableWidgetItem class is described by Table 18-236

Table 18-236 typeinfo for QTableWidgetItem

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTableWidgetItem

18.5.9.18 Class data for QTableWidget

The virtual table for the QTableWidget class is described by Table 18-237

Table 18-237 Primary vtable for QTableWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTableWidget
vfunc[0]:	QTableWidget::metaObject() const
vfunc[1]:	QTableWidget::qt_metacast(char const*)
vfunc[2]:	QTableWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTableWidget::~QTableWidget()
vfunc[4]:	QTableWidget::~QTableWidget()
vfunc[5]:	QTableWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTableView::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const

vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractItemView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractItemView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractItemView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractItemView::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTableView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractItemView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractItemView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractItemView::dragLeaveEvent(QDragLeaveEvent*)

vfunc[39]:	QAbstractItemView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QTableView::scrollContentsBy(int, int)
vfunc[56]:	QTableWidget::setModel(QAbstractItemModel*)
vfunc[57]:	QTableView::setSelectionModel(QItemSelectionModel*)
vfunc[58]:	QAbstractItemView::keyboardSearch(QString const&)
vfunc[59]:	QTableView::visualRect(QModelIndex const&) const
vfunc[60]:	QTableView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QTableView::indexAt(QPoint const&) const
vfunc[62]:	QTableView::sizeHintForRow(int) const

vfunc[63]:	QTableView::sizeHintForColumn(int)) const
vfunc[64]:	QAbstractItemView::reset()
vfunc[65]:	QTableView::setRootIndex(QModelIndex const&)
vfunc[66]:	QAbstractItemView::doItemsLayout())
vfunc[67]:	QAbstractItemView::selectAll()
vfunc[68]:	QAbstractItemView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QAbstractItemView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QAbstractItemView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)
vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QTableView::updateGeometries()
vfunc[76]:	QTableView::verticalScrollbarAction(int)
vfunc[77]:	QTableView::horizontalScrollbarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollbarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollbarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)

vfunc[83]:	QTableView::moveCursor(QAbstractItemView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QTableView::horizontalOffset() const
vfunc[85]:	QTableView::verticalOffset() const
vfunc[86]:	QTableView::isIndexHidden(QModelIndex const&) const
vfunc[87]:	QTableView::setSelection(QRect const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[88]:	QTableView::visualRegionForSelection(QItemSelection const&) const
vfunc[89]:	QTableView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelIndex const&, QAbstractItemView::EditTrigger, QEvent*)
vfunc[91]:	QAbstractItemView::selectionCommand(QModelIndex const&, QEvent const*) const
vfunc[92]:	QAbstractItemView::startDrag(QFlags<Qt::DropAction>)
vfunc[93]:	QTableView::viewOptions() const
vfunc[94]:	QTableWidget::mimeTypes() const
vfunc[95]:	QTableWidget::mimeData(QList<QTableWidgetItem*>) const
vfunc[96]:	QTableWidget::dropMimeData(int, int, QMimeData const*, Qt::DropAction)
vfunc[97]:	QTableWidget::supportedDropActions() const

The Run Time Type Information for the QTableWidget class is described by Table 18-238

Table 18-238 typeinfo for QTableWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTableWidget
basetype:	typeinfo for QTableView

18.5.9.19 Class data for QTreeView

The virtual table for the QTreeView class is described by Table 18-239

Table 18-239 Primary vtable for QTreeView

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTreeView
vfunc[0]:	QTreeView::metaObject() const
vfunc[1]:	QTreeView::qt_metacast(char const*)
vfunc[2]:	QTreeView::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTreeView::~~QTreeView()
vfunc[4]:	QTreeView::~~QTreeView()
vfunc[5]:	QAbstractItemView::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTreeView::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QTreeView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QTreeView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QTreeView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractItemView::mouseMoveEvent(QMouseEvent*)

vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTreeView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractItemView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractItemView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractItemView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractItemView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)

vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QTreeView::scrollContentsBy(int, int)
vfunc[56]:	QTreeView::setModel(QAbstractItemModel*)
vfunc[57]:	QTreeView::setSelectionModel(QItemSelectionModel*)
vfunc[58]:	QTreeView::keyboardSearch(QString const&)
vfunc[59]:	QTreeView::visualRect(QModelIndex const&) const
vfunc[60]:	QTreeView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QTreeView::indexAt(QPoint const&) const
vfunc[62]:	QAbstractItemView::sizeHintForRow(int) const
vfunc[63]:	QTreeView::sizeHintForColumn(int) const
vfunc[64]:	QTreeView::reset()
vfunc[65]:	QTreeView::setRootIndex(QModelIndex const&)
vfunc[66]:	QTreeView::doItemsLayout()
vfunc[67]:	QTreeView::selectAll()
vfunc[68]:	QTreeView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QTreeView::rowsInserted(QModelIndex const&, int, int)

vfunc[70]:	QTreeView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)
vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QTreeView::updateGeometries()
vfunc[76]:	QAbstractItemView::verticalScrollbarAction(int)
vfunc[77]:	QTreeView::horizontalScrollbarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollbarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollbarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)
vfunc[83]:	QTreeView::moveCursor(QAbstractItemView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QTreeView::horizontalOffset() const
vfunc[85]:	QTreeView::verticalOffset() const
vfunc[86]:	QTreeView::isIndexHidden(QModelIndex const&) const
vfunc[87]:	QTreeView::setSelection(QRect const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[88]:	QTreeView::visualRegionForSelection(QItemSelection const&) const
vfunc[89]:	QTreeView::selectedIndexes() const

vfunc[90]:	QAbstractItemView::edit(QModelIndex const&, QAbstractItemView::EditTrigger, QEvent*)
vfunc[91]:	QAbstractItemView::selectionCommand(QModelIndex const&, QEvent const*) const
vfunc[92]:	QAbstractItemView::startDrag(QFlags<Qt::DropAction>)
vfunc[93]:	QAbstractItemView::viewOptions() const
vfunc[94]:	QTreeView::drawRow(QPainter*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[95]:	QTreeView::drawBranches(QPainter*, QRect const&, QModelIndex const&) const

The Run Time Type Information for the QTreeView class is described by Table 18-240

Table 18-240 typeinfo for QTreeView

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTreeView
basetype:	typeinfo for QAbstractItemView

18.5.9.20 Class data for QProxyModel

The virtual table for the QProxyModel class is described by Table 18-241

Table 18-241 Primary vtable for QProxyModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QProxyModel
vfunc[0]:	QProxyModel::metaObject() const
vfunc[1]:	QProxyModel::qt_metacast(char const*)
vfunc[2]:	QProxyModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QProxyModel::~~QProxyModel()
vfunc[4]:	QProxyModel::~~QProxyModel()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QProxyModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QProxyModel::parent(QModelIndex const&) const
vfunc[14]:	QProxyModel::rowCount(QModelIndex const&) const
vfunc[15]:	QProxyModel::columnCount(QModelIndex const&) const
vfunc[16]:	QProxyModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QProxyModel::data(QModelIndex const&, int) const
vfunc[18]:	QProxyModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QProxyModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QProxyModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QProxyModel::mimeTypes() const
vfunc[24]:	QProxyModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QProxyModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QProxyModel::supportedDropActions() const
vfunc[27]:	QProxyModel::insertRows(int, int, QModelIndex const&)

vfunc[28]:	QProxyModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QAbstractItemModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QProxyModel::fetchMore(QModelIndex const&)
vfunc[32]:	QAbstractItemModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QProxyModel::flags(QModelIndex const&) const
vfunc[34]:	QProxyModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QProxyModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QProxyModel::span(QModelIndex const&) const
vfunc[38]:	QProxyModel::submit()
vfunc[39]:	QProxyModel::revert()
vfunc[40]:	QProxyModel::setModel(QAbstractItemModel*)

The Run Time Type Information for the QProxyModel class is described by Table 18-242

Table 18-242 typeinfo for QProxyModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QProxyModel
basetype:	typeinfo for QAbstractItemModel

18.5.9.21 Class data for QHeaderView

The virtual table for the QHeaderView class is described by Table 18-243

Table 18-243 Primary vtable for QHeaderView

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHeaderView

vfunc[0]:	QHeaderView::metaObject() const
vfunc[1]:	QHeaderView::qt_metacast(char const*)
vfunc[2]:	QHeaderView::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QHeaderView::~QHeaderView()
vfunc[4]:	QHeaderView::~QHeaderView()
vfunc[5]:	QHeaderView::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractItemView::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QHeaderView::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QHeaderView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QHeaderView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QHeaderView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QHeaderView::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)

vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QHeaderView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractItemView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractItemView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractItemView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractItemView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)

vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QHeaderView::viewportEvent(QEvent*)
vfunc[55]:	QHeaderView::scrollContentsBy(int, int)
vfunc[56]:	QHeaderView::setModel(QAbstractItemModel*)
vfunc[57]:	QAbstractItemView::setSelectionMode(QItemSelectionModel*)
vfunc[58]:	QAbstractItemView::keyboardSearch(QString const&)
vfunc[59]:	QHeaderView::visualRect(QModelIndex const&) const
vfunc[60]:	QHeaderView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QHeaderView::indexAt(QPoint const&) const
vfunc[62]:	QAbstractItemView::sizeHintForRow(int) const
vfunc[63]:	QAbstractItemView::sizeHintForColumn(int) const
vfunc[64]:	QAbstractItemView::reset()
vfunc[65]:	QAbstractItemView::setRootIndex(QModelIndex const&)
vfunc[66]:	QHeaderView::doItemsLayout()
vfunc[67]:	QAbstractItemView::selectAll()
vfunc[68]:	QHeaderView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QHeaderView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QAbstractItemView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)

vfunc[72]:	QHeaderView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QHeaderView::updateGeometries()
vfunc[76]:	QAbstractItemView::verticalScrollBarAction(int)
vfunc[77]:	QAbstractItemView::horizontalScrollBarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollBarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollBarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)
vfunc[83]:	QHeaderView::moveCursor(QAbstractItemView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QHeaderView::horizontalOffset() const
vfunc[85]:	QHeaderView::verticalOffset() const
vfunc[86]:	QHeaderView::isIndexHidden(QModelIndex const&) const
vfunc[87]:	QHeaderView::setSelection(QRect const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[88]:	QHeaderView::visualRegionForSelection(QItemSelection const&) const
vfunc[89]:	QAbstractItemView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelIndex const&,

	QAbstractItemView::EditTrigger, QEvent*)
vfunc[91]:	QAbstractItemView::selectionComm and(QModelIndex const&, QEvent const*) const
vfunc[92]:	QAbstractItemView::startDrag(QFlags<Qt::DropAction>)
vfunc[93]:	QAbstractItemView::viewOptions() const
vfunc[94]:	QHeaderView::paintSection(QPainter*, QRect const&, int) const
vfunc[95]:	QHeaderView::sectionSizeFromContents(int) const

The Run Time Type Information for the QHeaderView class is described by Table 18-244

Table 18-244 typeinfo for QHeaderView

Base Vtable	vtable for __cxxabi1::__si_class_type_info
Name	typeinfo name for QHeaderView
basetype:	typeinfo for QAbstractItemView

18.5.9.22 Class data for QItemEditorFactory

The virtual table for the QItemEditorFactory class is described by Table 18-245

Table 18-245 Primary vtable for QItemEditorFactory

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QItemEditorFactory
vfunc[0]:	QItemEditorFactory::~~QItemEditorFactory()
vfunc[1]:	QItemEditorFactory::~~QItemEditorFactory()
vfunc[2]:	QItemEditorFactory::createEditor(QVariant::Type, QWidget*) const
vfunc[3]:	QItemEditorFactory::valuePropertyName(QVariant::Type) const

The Run Time Type Information for the QItemEditorFactory class is described by Table 18-246

Table 18-246 typeinfo for QItemEditorFactory

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QItemEditorFactory

18.5.9.23 Class data for QTreeWidgetItem

The virtual table for the QTreeWidgetItem class is described by Table 18-247

Table 18-247 Primary vtable for QTreeWidgetItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTreeWidgetItem
vfunc[0]:	QTreeWidgetItem::~~QTreeWidgetItem()
vfunc[1]:	QTreeWidgetItem::~~QTreeWidgetItem()
vfunc[2]:	QTreeWidgetItem::clone() const
vfunc[3]:	QTreeWidgetItem::data(int, int) const
vfunc[4]:	QTreeWidgetItem::setData(int, int, QVariant const&)
vfunc[5]:	QTreeWidgetItem::operator<(QTreeWidgetItem const&) const
vfunc[6]:	QTreeWidgetItem::read(QDataStream&)
vfunc[7]:	QTreeWidgetItem::write(QDataStream&) const

The Run Time Type Information for the QTreeWidgetItem class is described by Table 18-248

Table 18-248 typeinfo for QTreeWidgetItem

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTreeWidgetItem

18.5.9.24 Class data for QTreeWidget

The virtual table for the QTreeWidget class is described by Table 18-249

Table 18-249 Primary vtable for QTreeWidget

Base Offset	0
Virtual Base Offset	0

RTTI	typeinfo for QWidget
vfunc[0]:	QWidget::metaObject() const
vfunc[1]:	QWidget::qt_metacast(char const*)
vfunc[2]:	QWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QWidget::~~QWidget()
vfunc[4]:	QWidget::~~QWidget()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTreeView::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QTreeView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QTreeView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QTreeView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractItemView::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)

vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTreeView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractItemView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractItemView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractItemView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractItemView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)

vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QTreeView::scrollContentsBy(int, int)
vfunc[56]:	QTreeView::setModel(QAbstractItemModel*)
vfunc[57]:	QTreeView::setSelectionModel(QItemSelectionModel*)
vfunc[58]:	QTreeView::keyboardSearch(QString const&)
vfunc[59]:	QTreeView::visualRect(QModelIndex const&) const
vfunc[60]:	QTreeView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QTreeView::indexAt(QPoint const&) const
vfunc[62]:	QAbstractItemView::sizeHintForRow(int) const
vfunc[63]:	QTreeView::sizeHintForColumn(int) const
vfunc[64]:	QTreeView::reset()
vfunc[65]:	QTreeView::setRootIndex(QModelIndex const&)
vfunc[66]:	QTreeView::doItemsLayout()
vfunc[67]:	QTreeView::selectAll()
vfunc[68]:	QTreeView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QTreeView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QTreeView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)

vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QTreeView::updateGeometries()
vfunc[76]:	QAbstractItemView::verticalScrollbarAction(int)
vfunc[77]:	QTreeView::horizontalScrollbarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollbarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollbarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)
vfunc[83]:	QTreeView::moveCursor(QAbstractItemView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QTreeView::horizontalOffset() const
vfunc[85]:	QTreeView::verticalOffset() const
vfunc[86]:	QTreeView::isIndexHidden(QModelIndex const&) const
vfunc[87]:	QTreeView::setSelection(QRect const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[88]:	QTreeView::visualRegionForSelection(QItemSelection const&) const
vfunc[89]:	QTreeView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelIndex const&, QAbstractItemView::EditTrigger, QEvent*)

vfunc[91]:	QAbstractItemView::selectionComm and(QModelIndex const&, QEvent const*) const
vfunc[92]:	QAbstractItemView::startDrag(QFlag s<Qt::DropAction>)
vfunc[93]:	QAbstractItemView::viewOptions() const
vfunc[94]:	QTreeView::drawRow(QPainter*, QStyleOptionViewItem const&, QModelIndex const&) const
vfunc[95]:	QTreeView::drawBranches(QPainter*, QRect const&, QModelIndex const&) const
vfunc[96]:	QTreeWidget::mimeTypes() const
vfunc[97]:	QTreeWidget::mimeData(QList<QTr eeWidgetItem*>) const
vfunc[98]:	QTreeWidget::dropMimeData(QTree WidgetItem*, int, QMimeData const*, Qt::DropAction)
vfunc[99]:	QTreeWidget::supportedDropAction s() const

The Run Time Type Information for the QTreeWidget class is described by Table 18-250

Table 18-250 typeinfo for QTreeWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTreeWidget
basetype:	typeinfo for QTreeView

18.5.9.25 Class data for QAbstractButton

The virtual table for the QAbstractButton class is described by Table 18-251

Table 18-251 Primary vtable for QAbstractButton

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractButton
vfunc[0]:	QAbstractButton::metaObject() const
vfunc[1]:	QAbstractButton::qt_metacast(char const*)
vfunc[2]:	QAbstractButton::qt_metacall(QMeta Object::Call, int, void**)

vfunc[3]:	QAbstractButton::~~QAbstractButton()
vfunc[4]:	QAbstractButton::~~QAbstractButton()
vfunc[5]:	QAbstractButton::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractButton::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractButton::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractButton::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractButton::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractButton::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractButton::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractButton::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractButton::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)

vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	__cxa_pure_virtual
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractButton::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

vfunc[54]:	QAbstractButton::hitButton(QPoint const&) const
vfunc[55]:	QAbstractButton::checkStateSet()
vfunc[56]:	QAbstractButton::nextCheckState()

The Run Time Type Information for the QAbstractButton class is described by Table 18-252

Table 18-252 typeinfo for QAbstractButton

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractButton
basetype:	typeinfo for QWidget

18.5.9.26 Class data for QAbstractGraphicsShapeItem

The virtual table for the QAbstractGraphicsShapeItem class is described by Table 18-253

Table 18-253 Primary vtable for QAbstractGraphicsShapeItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractGraphicsShapeItem
vfunc[0]:	QAbstractGraphicsShapeItem::~~QAbstractGraphicsShapeItem()
vfunc[1]:	QAbstractGraphicsShapeItem::~~QAbstractGraphicsShapeItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	QGraphicsItem::shape() const
vfunc[5]:	QGraphicsItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QAbstractGraphicsShapeItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QAbstractGraphicsShapeItem::opaqueArea() const

vfunc[10]:	__cxa_pure_virtual
vfunc[11]:	QGraphicsItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)

vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)
vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsItem::extension(QVariant const&) const

18.5.9.27 Class data for QCompleter

The virtual table for the QCompleter class is described by Table 18-254

Table 18-254 Primary vtable for QCompleter

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCompleter
vfunc[0]:	QCompleter::metaObject() const
vfunc[1]:	QCompleter::qt_metacast(char const*)
vfunc[2]:	QCompleter::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QCompleter::~QCompleter()
vfunc[4]:	QCompleter::~~QCompleter()
vfunc[5]:	QCompleter::event(QEvent*)
vfunc[6]:	QCompleter::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QCompleter::pathFromIndex(QModelIndex const&) const

vfunc[13]:	QCompleter::splitPath(QString const&) const

18.5.9.28 Class data for QDataWidgetMapper

The virtual table for the QDataWidgetMapper class is described by Table 18-255

Table 18-255 Primary vtable for QDataWidgetMapper

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDataWidgetMapper
vfunc[0]:	QDataWidgetMapper::metaObject() const
vfunc[1]:	QDataWidgetMapper::qt_metacast(char const*)
vfunc[2]:	QDataWidgetMapper::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDataWidgetMapper::~~QDataWidgetMapper()
vfunc[4]:	QDataWidgetMapper::~~QDataWidgetMapper()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QDataWidgetMapper::setCurrentIndex(int)

18.5.9.29 Class data for QStandardItem

The virtual table for the QStandardItem class is described by Table 18-256

Table 18-256 Primary vtable for QStandardItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStandardItem
vfunc[0]:	QStandardItem::~~QStandardItem()

vfunc[1]:	QStandardItem::~~QStandardItem()
vfunc[2]:	QStandardItem::data(int) const
vfunc[3]:	QStandardItem::setData(QVariant const&, int)
vfunc[4]:	QStandardItem::clone() const
vfunc[5]:	QStandardItem::type() const
vfunc[6]:	QStandardItem::read(QDataStream&)
vfunc[7]:	QStandardItem::write(QDataStream &) const
vfunc[8]:	QStandardItem::operator<(QStandardItem const&) const

18.5.9.30 Interfaces for Qt4 Models and Views

An LSB conforming implementation shall provide the generic functions for Qt4 Models and Views specified in Table 18-257, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-257 libQtGui - Qt4 Models and Views Function Interfaces

_ZN10QCompleter11eventFilterEP7QObjectP6QEvent [QtXml]	_ZN10QCompleter11highlightedERK11QModelIndex [QtXml]
_ZN10QCompleter11highlightedERK7QString [QtXml]	_ZN10QCompleter11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN10QCompleter11qt_metacastEPKc [QtXml]	_ZN10QCompleter13setCurrentRowEi [QtXml]
_ZN10QCompleter15setModelSortingENS_12ModelSortingE [QtXml]	_ZN10QCompleter17setCompletionModeENS_14CompletionModeE [QtXml]
_ZN10QCompleter17setCompletionRoleEi [QtXml]	_ZN10QCompleter18setCaseSensitivityEN2Qt15CaseSensitivityE [QtXml]
_ZN10QCompleter19setCompletionColumnEi [QtXml]	_ZN10QCompleter19setCompletionPrefixERK7QString [QtXml]
_ZN10QCompleter5eventEP6QEvent [QtXml]	_ZN10QCompleter8completeERK5QRect [QtXml]
_ZN10QCompleter8setModelEP18QAbstractItemModel [QtXml]	_ZN10QCompleter8setPopupEP17QAbstractItemView [QtXml]
_ZN10QCompleter9activatedERK11QModelIndex [QtXml]	_ZN10QCompleter9activatedERK7QString [QtXml]
_ZN10QCompleter9setWidgetEP7QWidget [QtXml]	_ZN10QCompleterC1EP18QAbstractItemModelP7QObject [QtXml]
_ZN10QCompleterC1EP7QObject [QtXml]	_ZN10QCompleterC1ERK11QStringListP7QObject [QtXml]

_ZN10QCompleterC2EP18QAbstractItemModelP7QObject [QtXml]	_ZN10QCompleterC2EP7QObject [QtXml]
_ZN10QCompleterC2ERK11QStringListP7QObject [QtXml]	_ZN10QCompleterD0Ev [QtXml]
_ZN10QCompleterD1Ev [QtXml]	_ZN10QCompleterD2Ev [QtXml]
_ZN10QTableView10hideColumnEi [QtGui]	_ZN10QTableView10moveCursorEN17QAbstractItemView12CursorActionE6QFlagsIN2Qt16KeyboardModifiersEE [QtGui]
_ZN10QTableView10paintEventEP11QPaintEvent [QtGui]	_ZN10QTableView10rowResizedEiii [QtGui]
_ZN10QTableView10showColumnEi [QtGui]	_ZN10QTableView10timerEventEP11QTimerEvent [QtGui]
_ZN10QTableView11columnMovedEiii [QtGui]	_ZN10QTableView11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN10QTableView11qt_metacastEPKc [QtGui]	_ZN10QTableView11setShowGridEb [QtGui]
_ZN10QTableView12selectColumnEi [QtGui]	_ZN10QTableView12setGridStyleEN2Qt8PenStyleE [QtGui]
_ZN10QTableView12setRootIndexERK11QModelIndex [QtGui]	_ZN10QTableView12setRowHeightEii [QtGui]
_ZN10QTableView12setRowHiddenEib [QtGui]	_ZN10QTableView12setSelectionERK5QRect6QFlagsIN19QItemSelectionModel13SelectionFlagEE [QtGui]
_ZN10QTableView12sortByColumnEi [QtGui]	_ZN10QTableView12sortByColumnEiN2Qt9SortOrderE [QtXml]
_ZN10QTableView13columnResizedEiii [QtGui]	_ZN10QTableView14setColumnWidthEii [QtGui]
_ZN10QTableView15rowCountChangedEii [QtGui]	_ZN10QTableView15setColumnHiddenEib [QtGui]
_ZN10QTableView16scrollContentsByEii [QtGui]	_ZN10QTableView16updateGeometriesEv [QtGui]
_ZN10QTableView17setSelectionModeEP19QItemSelectionModel [QtGui]	_ZN10QTableView17setSortingEnabledEb [QtXml]
_ZN10QTableView17setVerticalHeaderEP11QHeaderView [QtGui]	_ZN10QTableView18columnCountChangedEii [QtGui]
_ZN10QTableView19resizeRowToContentsEi [QtGui]	_ZN10QTableView19setHorizontalHeaderEP11QHeaderView [QtGui]
_ZN10QTableView20resizeRowsToContentsEv [QtGui]	_ZN10QTableView22resizeColumnToContentsEi [QtGui]
_ZN10QTableView23resizeColumnsToContentsEv [QtGui]	_ZN10QTableView23verticalScrollBarActionEi [QtGui]

_ZN10QTableView25horizontalScrollBarActionEi [QtGui]	_ZN10QTableView7hideRowEi [QtGui]
_ZN10QTableView7setSpanEiiii [QtXml]	_ZN10QTableView7showRowEi [QtGui]
_ZN10QTableView8rowMovedEiii [QtGui]	_ZN10QTableView8scrollToERK11QModelIndexN17QAbstractItemView10ScrollHintE [QtGui]
_ZN10QTableView8setModelEP18QAbstractItemModel [QtGui]	_ZN10QTableView9selectRowEi [QtGui]
_ZN10QTableViewC1EP7QWidget [QtGui]	_ZN10QTableViewC2EP7QWidget [QtGui]
_ZN10QTableViewD0Ev [QtGui]	_ZN10QTableViewD1Ev [QtGui]
_ZN10QTableViewD2Ev [QtGui]	_ZN11QHeaderView10initializeEv [QtGui]
_ZN11QHeaderView10moveCursorEN17QAbstractItemView12CursorActionE6QFlagsIN2Qt16KeyboardModifierEE [QtGui]	_ZN11QHeaderView10paintEventEP11QPaintEvent [QtGui]
_ZN11QHeaderView10setMovableEb [QtGui]	_ZN11QHeaderView11dataChangedERK11QModelIndexS2_ [QtGui]
_ZN11QHeaderView11moveSectionEii [QtGui]	_ZN11QHeaderView11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN11QHeaderView11qt_metacastEPKc [QtGui]	_ZN11QHeaderView12rowsInsertedERK11QModelIndexii [QtGui]
_ZN11QHeaderView12sectionMovedEiii [QtGui]	_ZN11QHeaderView12setClickableEb [QtGui]
_ZN11QHeaderView12setSelectionERK5QRect6QFlagsIN19QItemSelectionModel13SelectionFlagEE [QtGui]	_ZN11QHeaderView12swapSectionsEii [QtXml]
_ZN11QHeaderView13doItemsLayoutEv [QtGui]	_ZN11QHeaderView13resizeSectionEii [QtGui]
_ZN11QHeaderView13setResizeModeENS_10ResizeModeE [QtGui]	_ZN11QHeaderView13setResizeModeEiNS_10ResizeModeE [QtGui]
_ZN11QHeaderView13updateSectionEi [LSB]	_ZN11QHeaderView13viewportEventEP6QEvent [QtGui]
ZN11QHeaderView14currentChangedERK11QModelIndexS2 [QtGui]	_ZN11QHeaderView14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN11QHeaderView14resizeSectionsENS_10ResizeModeE [QtGui]	_ZN11QHeaderView14resizeSectionsEv [QtGui]
_ZN11QHeaderView14sectionClickedEi [QtGui]	_ZN11QHeaderView14sectionPressedEi [QtGui]
_ZN11QHeaderView14sectionResizedEiii [QtGui]	_ZN11QHeaderView15mousePressEventEP11QMouseEvent [QtGui]

_ZN11QHeaderView16scrollContentsByEii [QtGui]	_ZN11QHeaderView16sectionsInsertedERK11QModelIndexii [QtGui]
_ZN11QHeaderView16setSectionHiddenEib [QtGui]	_ZN11QHeaderView16setSortIndicatorEiN2Qt9SortOrderE [QtGui]
_ZN11QHeaderView16updateGeometriesEv [QtGui]	_ZN11QHeaderView17geometriesChangedEv [QtXml]
_ZN11QHeaderView17headerDataChangedEN2Qt11OrientationEii [QtGui]	_ZN11QHeaderView17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN11QHeaderView17sectionAutoResizeEiNS_10ResizeModeE [QtGui]	_ZN11QHeaderView18initializeSectionsEii [QtGui]
_ZN11QHeaderView18initializeSectionsEv [QtGui]	_ZN11QHeaderView19sectionCountChangedEii [QtGui]
_ZN11QHeaderView19setDefaultAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN11QHeaderView20sectionDoubleClickedEi [QtGui]
_ZN11QHeaderView20setHighlightSectionsEb [QtGui]	_ZN11QHeaderView21mouseDoubleClickEventEP11QMouseEvent [QtGui]
_ZN11QHeaderView21setDefaultSectionSizeEi [QtGui]	_ZN11QHeaderView21setMinimumSectionSizeEi [QtXml]
_ZN11QHeaderView21setSortIndicatorShownEb [QtGui]	_ZN11QHeaderView21setStretchLastSectionEb [QtGui]
_ZN11QHeaderView24sectionsAboutToBeRemovedERK11QModelIndexii [QtGui]	_ZN11QHeaderView26sectionHandleDoubleClickedEi [QtGui]
_ZN11QHeaderView26setCascadingSectionResizesEb [QtXml]	_ZN11QHeaderView26setOffsetToSectionPositionEi [QtXml]
_ZN11QHeaderView5eventEP6QEvent [QtGui]	_ZN11QHeaderView8scrollToERK11QModelIndexN17QAbstractItemView10ScrollHintE [QtGui]
_ZN11QHeaderView8setModelEP18QAbstractItemModel [QtGui]	_ZN11QHeaderView9setOffsetEi [QtGui]
_ZN11QHeaderViewC1EN2Qt11OrientationEP7QWidget [QtGui]	_ZN11QHeaderViewC2EN2Qt11OrientationEP7QWidget [QtGui]
_ZN11QHeaderViewD0Ev [QtGui]	_ZN11QHeaderViewD1Ev [QtGui]
_ZN11QHeaderViewD2Ev [QtGui]	_ZN11QProxyModel10insertRowsEiiRK11QModelIndex [QtGui]
_ZN11QProxyModel11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QProxyModel11qt_metacastEPKc [QtGui]
_ZN11QProxyModel12dropMimeTypeDataEPK9QMimeTypeDataN2Qt10DropActionEiiRK11QModelIndex [QtGui]	_ZN11QProxyModel13insertColumnsEiiRK11QModelIndex [QtGui]

_ZN11QProxyModel13setHeaderDataEiN2Qt11OrientationERK8QVarianti [QtGui]	_ZN11QProxyModel4sortEiN2Qt9SortOrderE [QtGui]
_ZN11QProxyModel6revertEv [QtGui]	_ZN11QProxyModel6submitEv [QtGui]
_ZN11QProxyModel7setDataERK11QModelIndexRK8QVarianti [QtGui]	_ZN11QProxyModel8setModelEP18QAbstractItemModel [QtGui]
_ZN11QProxyModel9fetchMoreERK11QModelIndex [QtGui]	_ZN11QProxyModelC1EP7QObject [QtGui]
_ZN11QProxyModelC2EP7QObject [QtGui]	_ZN11QProxyModelD0Ev [QtGui]
_ZN11QProxyModelD1Ev [QtGui]	_ZN11QProxyModelD2Ev [QtGui]
_ZN11QTreeWidget10expandItemEPK15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget11itemChangedEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget11itemClickedEP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget11itemEnteredEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget11itemPressedEP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN11QTreeWidget11qt_metacastEPKc [QtGui]	_ZN11QTreeWidget12collapseItemEPK15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget12dropMimeTypeEP15QTreeWidgetItemPK9QMimeTypeDataN2Qt10DropActionE [QtGui]	_ZN11QTreeWidget12itemExpandedEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget12scrollToItemEPK15QTreeWidgetItemN17QAbstractItemView10ScrollHintE [QtGui]	_ZN11QTreeWidget13itemActivatedEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget13itemCollapsedEP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget13setHeaderItemEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget13setItemHiddenEPK15QTreeWidgetItemb [QtGui]	_ZN11QTreeWidget13setItemWidgetEP15QTreeWidgetItemP7QWidget [QtGui]
_ZN11QTreeWidget14setColumnCountEi [QtGui]	_ZN11QTreeWidget14setCurrentItemEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget14setCurrentItemEP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget15addTopLevelItemEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget15setHeaderLabelsERK11QStringList [QtGui]	_ZN11QTreeWidget15setItemExpandedEPK15QTreeWidgetItemb [QtGui]
_ZN11QTreeWidget15setItemSelectedEPK15QTreeWidgetItemb [QtGui]	_ZN11QTreeWidget16addTopLevelItemsERK5QListIP15QTreeWidgetItemE [QtGui]
_ZN11QTreeWidget16takeTopLevelItemEi [QtGui]	_ZN11QTreeWidget17itemDoubleClickedEP15QTreeWidgetItem [QtGui]

_ZN11QTreeWidget17setSortingEnabledEb [QtGui]	_ZN11QTreeWidget18currentItemChangedEP15QTreeWidgetItemS1_ [QtGui]
_ZN11QTreeWidget18insertTopLevelItemEiP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget19indexOfTopLevelItemEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget19insertTopLevelItemsEiRK5QListIP15QTreeWidgetItemE [QtGui]	_ZN11QTreeWidget20itemSelectionChangedEv [QtGui]
_ZN11QTreeWidget20openPersistentEditorEP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget21closePersistentEditorEP15QTreeWidgetItem [QtGui]
_ZN11QTreeWidget5clearEv [QtGui]	_ZN11QTreeWidget5eventEP6QEvent [QtGui]
_ZN11QTreeWidget8editItemEP15QTreeWidgetItem [QtGui]	_ZN11QTreeWidget8setModelEP18QAbstractItemModel [QtGui]
_ZN11QTreeWidget9dropEventEP10QDropEvent [QtXml]	_ZN11QTreeWidget9sortItemsEiN2Qt9SortOrderE [QtGui]
_ZN11QTreeWidgetC1EP7QWidget [QtGui]	_ZN11QTreeWidgetC2EP7QWidget [QtGui]
_ZN11QTreeWidgetD0Ev [QtGui]	_ZN11QTreeWidgetD1Ev [QtGui]
_ZN11QTreeWidgetD2Ev [QtGui]	_ZN12QTableWidget11cellChangedEii [QtGui]
_ZN12QTableWidget11cellClickedEii [QtGui]	_ZN12QTableWidget11cellEnteredEii [QtGui]
_ZN12QTableWidget11cellPressedEii [QtGui]	_ZN12QTableWidget11itemChangedEP16QTableWidgetItem [QtGui]
_ZN12QTableWidget11itemClickedEP16QTableWidgetItem [QtGui]	_ZN12QTableWidget11itemEnteredEP16QTableWidgetItem [QtGui]
_ZN12QTableWidget11itemPressedEP16QTableWidgetItem [QtGui]	_ZN12QTableWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN12QTableWidget11qt_metacastEPKc [QtGui]	_ZN12QTableWidget11setRowCountEi [QtGui]
_ZN12QTableWidget12dropMimeTypeEiiPK9QMimeTypeN2Qt10DropActionE [QtGui]	_ZN12QTableWidget12insertColumnEi [QtGui]
_ZN12QTableWidget12removeColumnEi [QtGui]	_ZN12QTableWidget12scrollToItemEPK16QTableWidgetItemN17QAbstractItemView10ScrollHintE [QtGui]
_ZN12QTableWidget13cellActivatedEii [QtGui]	_ZN12QTableWidget13clearContentsEv [QtXml]
_ZN12QTableWidget13itemActivatedEP16QTableWidgetItem [QtGui]	_ZN12QTableWidget13selectedItemsEv [QtGui]

_ZN12QTableWidget13setCellWidthEiiP7QWidget [QtGui]	_ZN12QTableWidget14setColumnCountEi [QtGui]
_ZN12QTableWidget14setCurrentCellEii [QtGui]	_ZN12QTableWidget14setCurrentItemEP16QTableWidgetItem [QtGui]
_ZN12QTableWidget15setItemSelectedEPK16QTableWidgetItemb [QtGui]	_ZN12QTableWidget16setItemPrototypeEPK16QTableWidgetItem [QtGui]
_ZN12QTableWidget16setRangeSelectedERK26QTableWidgetSelectionRangeb [QtGui]	_ZN12QTableWidget17cellDoubleClickedEii [QtGui]
_ZN12QTableWidget17itemDoubleClickedEP16QTableWidgetItem [QtGui]	_ZN12QTableWidget17setSortingEnabledEb [QtGui]
_ZN12QTableWidget18currentCellChangedEiiii [QtGui]	_ZN12QTableWidget18currentItemChangedEP16QTableWidgetItemS1_ [QtGui]
_ZN12QTableWidget20itemSelectionChangedEv [QtGui]	_ZN12QTableWidget20openPersistentEditorEP16QTableWidgetItem [QtGui]
_ZN12QTableWidget21closePersistentEditorEP16QTableWidgetItem [QtGui]	_ZN12QTableWidget21setVerticalHeaderItemEiP16QTableWidgetItem [QtGui]
_ZN12QTableWidget22takeVerticalHeaderItemEi [QtGui]	_ZN12QTableWidget23setHorizontalHeaderItemEiP16QTableWidgetItem [QtGui]
_ZN12QTableWidget23setVerticalHeaderLabelsERK11QStringList [QtGui]	_ZN12QTableWidget24takeHorizontalHeaderItemEi [QtGui]
_ZN12QTableWidget25setHorizontalHeaderLabelsERK11QStringList [QtGui]	_ZN12QTableWidget5clearEv [QtGui]
_ZN12QTableWidget5eventEP6QEvent [QtGui]	_ZN12QTableWidget7setItemEiiP16QTableWidgetItem [QtGui]
_ZN12QTableWidget8editItemEP16QTableWidgetItem [QtGui]	_ZN12QTableWidget8setModelEP18QAbstractItemModel [QtGui]
_ZN12QTableWidget8takeItemEii [QtGui]	_ZN12QTableWidget9dropEventEP10QDropEvent [QtGui]
_ZN12QTableWidget9insertRowEi [QtGui]	_ZN12QTableWidget9removeRowEi [QtGui]
_ZN12QTableWidget9sortItemsEiN2Qt9SortOrderE [QtGui]	_ZN12QTableWidgetC1EP7QWidget [QtGui]
_ZN12QTableWidgetC1EiiP7QWidget [QtGui]	_ZN12QTableWidgetC2EP7QWidget [QtGui]
_ZN12QTableWidgetC2EiiP7QWidget [QtGui]	_ZN12QTableWidgetD0Ev [QtGui]

_ZN12QWidgetD1Ev [QtGui]	_ZN12QWidgetD2Ev [QtGui]
_ZN13QItemDelegate11editorEventEP6QEventP18QAbstractItemModelRK20QStyleOptionViewItemRK11QModelIndex [QtGui]	_ZN13QItemDelegate11eventFilterEP7QObjectP6QEvent [QtGui]
_ZN13QItemDelegate11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN13QItemDelegate11qt_metacastEPKc [QtGui]
_ZN13QItemDelegate11setClippingEb [QtXml]	_ZN13QItemDelegate20setItemEditorFactoryEP18QItemEditorFactory [QtGui]
_ZN13QItemDelegateC1EP7QObject [QtGui]	_ZN13QItemDelegateC2EP7QObject [QtGui]
_ZN13QItemDelegateD0Ev [QtGui]	_ZN13QItemDelegateD1Ev [QtGui]
_ZN13QItemDelegateD2Ev [QtGui]	_ZN13QStandardItem10insertRowsEii [QtXml]
_ZN13QStandardItem10removeRowsEii [QtXml]	_ZN13QStandardItem10setEnabledEb [QtXml]
_ZN13QStandardItem10takeColumnEi [QtXml]	_ZN13QStandardItem11setEditableEb [QtXml]
_ZN13QStandardItem11setRowCountEi [QtXml]	_ZN13QStandardItem11setTristateEb [QtXml]
_ZN13QStandardItem12insertColumnEiRK5QListIPS_E [QtXml]	_ZN13QStandardItem12removeColumnEi [QtXml]
_ZN13QStandardItem12setCheckableEb [QtXml]	_ZN13QStandardItem12sortChildrenEiN2Qt9SortOrderE [QtXml]
_ZN13QStandardItem13insertColumnsEii [QtXml]	_ZN13QStandardItem13removeColumnsEii [QtXml]
_ZN13QStandardItem13setSelectableEb [QtXml]	_ZN13QStandardItem14setColumnCountEi [QtXml]
_ZN13QStandardItem14setDragEnabledEb [QtXml]	_ZN13QStandardItem14setDropEnabledEb [QtXml]
_ZN13QStandardItem4readER11QDataStream [QtXml]	_ZN13QStandardItem7setDataERK8QVarianti [QtXml]
_ZN13QStandardItem7takeRowEi [QtXml]	_ZN13QStandardItem8setChildEiiPS_ [QtXml]
_ZN13QStandardItem8setFlagsE6QFlagsIN2Qt8ItemFlagEE [QtXml]	_ZN13QStandardItem9insertRowEiRK5QListIPS_E [QtXml]
_ZN13QStandardItem9removeRowEi [QtXml]	_ZN13QStandardItem9takeChildEii [QtXml]
_ZN13QStandardItemC1ERK5QIconRK7QString [QtXml]	_ZN13QStandardItemC1ERK7QString [QtXml]

ZN13QStandardItemC1ERKS [QtXml]	_ZN13QStandardItemC1Eii [QtXml]
_ZN13QStandardItemC1Ev [QtXml]	_ZN13QStandardItemC2ERK5QIcon RK7QString [QtXml]
_ZN13QStandardItemC2ERK7QStrin g [QtXml]	_ZN13QStandardItemC2ERKS_ [QtXml]
_ZN13QStandardItemC2Eii [QtXml]	_ZN13QStandardItemC2Ev [QtXml]
_ZN13QStandardItemD0Ev [QtXml]	_ZN13QStandardItemD1Ev [QtXml]
_ZN13QStandardItemD2Ev [QtXml]	_ZN13QStandardItemaSERKS_ [QtXml]
_ZN14QItemSelection5mergeERKS_6 QFlagsIN19QItemSelectionModel13S electionFlagEE [QtGui]	_ZN14QItemSelection5splitERK19QIt emSelectionRangeS2_PS_ [QtGui]
ZN14QItemSelection6selectERK11Q ModelIndexS2 [QtGui]	_ZN14QItemSelectionC1ERK11QMo delIndexS2_ [QtGui]
ZN14QItemSelectionC2ERK11QMo delIndexS2 [QtGui]	_ZN15QAbstractButton10setChecked Eb [QtGui]
_ZN15QAbstractButton10timerEvent EP11QTimerEvent [QtGui]	_ZN15QAbstractButton11changeEve ntEP6QEvent [QtGui]
_ZN15QAbstractButton11qt_metacall EN11QMetaObject4CallEiPPv [QtGui]	_ZN15QAbstractButton11qt_metacas tEPKc [QtGui]
_ZN15QAbstractButton11setIconSize ERK5QSize [QtGui]	_ZN15QAbstractButton11setShortcut ERK12QKeySequence [QtGui]
_ZN15QAbstractButton12animateCli ckEi [QtGui]	_ZN15QAbstractButton12focusInEve ntEP11QFocusEvent [QtGui]
_ZN15QAbstractButton12setCheckab leEb [QtGui]	_ZN15QAbstractButton13checkState SetEv [QtGui]
_ZN15QAbstractButton13focusOutE ventEP11QFocusEvent [QtGui]	_ZN15QAbstractButton13keyPressEv entEP9QKeyEvent [QtGui]
_ZN15QAbstractButton13setAutoRe peatEb [QtGui]	_ZN15QAbstractButton14mouseMov eEventEP11QMouseEvent [QtGui]
_ZN15QAbstractButton14nextCheck StateEv [QtGui]	_ZN15QAbstractButton15keyRelease EventEP9QKeyEvent [QtGui]
_ZN15QAbstractButton15mousePres sEventEP11QMouseEvent [QtGui]	_ZN15QAbstractButton16setAutoExc lusiveEb [QtGui]
_ZN15QAbstractButton17mouseRele aseEventEP11QMouseEvent [QtGui]	_ZN15QAbstractButton18setAutoRe peatDelayEi [QtXml]
_ZN15QAbstractButton21setAutoRe peatIntervalEi [QtXml]	_ZN15QAbstractButton5clickEv [QtGui]
_ZN15QAbstractButton5eventEP6QE vent [QtGui]	_ZN15QAbstractButton6toggleEv [QtGui]

_ZN15QAbstractButton7clickedEb [QtGui]	_ZN15QAbstractButton7pressedEv [QtGui]
_ZN15QAbstractButton7setDownEb [QtGui]	_ZN15QAbstractButton7setIconERK5QIcon [QtGui]
_ZN15QAbstractButton7setTextERK7QString [QtGui]	_ZN15QAbstractButton7toggledEb [QtGui]
_ZN15QAbstractButton8releasedEv [QtGui]	_ZN15QAbstractButtonC1EP7QWid get [QtGui]
_ZN15QAbstractButtonC1EP7QWid getPKc6QFlagsIN2Qt10WindowType EE [QtGui]	_ZN15QAbstractButtonC2EP7QWid get [QtGui]
_ZN15QAbstractButtonC2EP7QWid getPKc6QFlagsIN2Qt10WindowType EE [QtGui]	_ZN15QAbstractButtonD0Ev [QtGui]
_ZN15QAbstractButtonD1Ev [QtGui]	_ZN15QAbstractButtonD2Ev [QtGui]
_ZN15QAbstractSlider10setMaximumEi [QtGui]	_ZN15QAbstractSlider10setMinimumEi [QtGui]
_ZN15QAbstractSlider10timerEventEP11QTimerEvent [QtGui]	_ZN15QAbstractSlider10wheelEventEP11QWheelEvent [QtGui]
_ZN15QAbstractSlider11changeEventEP6QEvent [QtGui]	_ZN15QAbstractSlider11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN15QAbstractSlider11qt_metacastEPKc [QtGui]	_ZN15QAbstractSlider11setPageStepEi [QtGui]
_ZN15QAbstractSlider11setTrackingEb [QtGui]	_ZN15QAbstractSlider11sliderMovedEi [QtGui]
_ZN15QAbstractSlider12rangeChangedEii [QtGui]	_ZN15QAbstractSlider12sliderChangeENS_12SliderChangeE [QtGui]
_ZN15QAbstractSlider12valueChangedEi [QtGui]	_ZN15QAbstractSlider13keyPressEventEP9QKeyEvent [QtGui]
_ZN15QAbstractSlider13setSingleStepEi [QtGui]	_ZN15QAbstractSlider13setSliderDownEb [QtGui]
_ZN15QAbstractSlider13sliderPressedEv [QtGui]	_ZN15QAbstractSlider13triggerActionENS_12SliderActionE [QtGui]
_ZN15QAbstractSlider14setOrientationEN2Qt11OrientationE [QtGui]	_ZN15QAbstractSlider14sliderReleasedEv [QtGui]
_ZN15QAbstractSlider15actionTriggeredEi [QtGui]	_ZN15QAbstractSlider15setRepeatActionENS_12SliderActionEii [QtGui]
_ZN15QAbstractSlider17setSliderPositionEi [QtGui]	_ZN15QAbstractSlider19setInvertedControlsEb [QtGui]
_ZN15QAbstractSlider21setInvertedAppearanceEb [QtGui]	_ZN15QAbstractSlider5eventEP6QEvent [QtGui]

_ZN15QAbstractSlider8setRangeEii [QtGui]	_ZN15QAbstractSlider8setValueEi [QtGui]
_ZN15QAbstractSliderC1EP7QWidg et [QtGui]	_ZN15QAbstractSliderC2EP7QWidg et [QtGui]
_ZN15QAbstractSliderD0Ev [QtGui]	_ZN15QAbstractSliderD1Ev [QtGui]
_ZN15QAbstractSliderD2Ev [QtGui]	_ZN15QTreeWidgetItem11addChildr enERK5QListIPS_E [QtGui]
ZN15QTreeWidgetItem11insertChil dEiPS [QtGui]	_ZN15QTreeWidgetItem11itemChan gedEv [QtXml]
_ZN15QTreeWidgetItem12takeChild renEv [QtGui]	_ZN15QTreeWidgetItem14insertChil drenEiRK5QListIPS_E [QtGui]
_ZN15QTreeWidgetItem4readER11Q DataStream [QtGui]	_ZN15QTreeWidgetItem7setDataEii RK8QVariant [QtGui]
ZN15QTreeWidgetItem8addChildE PS [QtGui]	_ZN15QTreeWidgetItem9takeChildE i [QtGui]
_ZN15QTreeWidgetItemC1EP11QTr eeWidgetPS_i [QtGui]	_ZN15QTreeWidgetItemC1EP11QTr eeWidgetRK11QStringListi [QtGui]
_ZN15QTreeWidgetItemC1EP11QTr eeWidgeti [QtGui]	_ZN15QTreeWidgetItemC1EPS_RK1 1QStringListi [QtGui]
_ZN15QTreeWidgetItemC1EPS_S0_i [QtGui]	_ZN15QTreeWidgetItemC1EPS_i [QtGui]
_ZN15QTreeWidgetItemC1ERK11QS tringListi [QtGui]	_ZN15QTreeWidgetItemC1ERKS_ [QtGui]
_ZN15QTreeWidgetItemC1Ei [QtGui]	_ZN15QTreeWidgetItemC2EP11QTr eeWidgetPS_i [QtGui]
_ZN15QTreeWidgetItemC2EP11QTr eeWidgetRK11QStringListi [QtGui]	_ZN15QTreeWidgetItemC2EP11QTr eeWidgeti [QtGui]
_ZN15QTreeWidgetItemC2EPS_RK1 1QStringListi [QtGui]	_ZN15QTreeWidgetItemC2EPS_S0_i [QtGui]
_ZN15QTreeWidgetItemC2EPS_i [QtGui]	_ZN15QTreeWidgetItemC2ERK11QS tringListi [QtGui]
ZN15QTreeWidgetItemC2ERKS [QtGui]	_ZN15QTreeWidgetItemC2Ei [QtGui]
_ZN15QTreeWidgetItemD0Ev [QtGui]	_ZN15QTreeWidgetItemD1Ev [QtGui]
_ZN15QTreeWidgetItemD2Ev [QtGui]	_ZN15QTreeWidgetItemD3Ev [QtGui]
_ZN16QAbstractSpinBox10closeEven tEP11QCloseEvent [QtGui]	_ZN16QAbstractSpinBox10paintEve ntEP11QPaintEvent [QtGui]
_ZN16QAbstractSpinBox10timerEve ntEP11QTimerEvent [QtGui]	_ZN16QAbstractSpinBox10wheelEve ntEP11QWheelEvent [QtGui]

_ZN16QAbstractSpinBox11changeEventEP6QEvent [QtGui]	_ZN16QAbstractSpinBox11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN16QAbstractSpinBox11qt_metacastEPKc [QtGui]	_ZN16QAbstractSpinBox11resizeEventEP12QResizeEvent [QtGui]
_ZN16QAbstractSpinBox11setLineEditEP9QLineEdit [QtGui]	_ZN16QAbstractSpinBox11setReadOnlyEb [QtGui]
_ZN16QAbstractSpinBox11setWrappingEb [QtGui]	_ZN16QAbstractSpinBox12focusInEventEP11QFocusEvent [QtGui]
_ZN16QAbstractSpinBox12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN16QAbstractSpinBox13focusOutEventEP11QFocusEvent [QtGui]
_ZN16QAbstractSpinBox13interpretTextEv [QtGui]	_ZN16QAbstractSpinBox13keyPressEventEP9QKeyEvent [QtGui]
_ZN16QAbstractSpinBox14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN16QAbstractSpinBox14setAcceleratedEb [QtGui]
_ZN16QAbstractSpinBox15editingFinishedEv [QtGui]	_ZN16QAbstractSpinBox15keyReleaseEventEP9QKeyEvent [QtGui]
_ZN16QAbstractSpinBox15mousePressEventEP11QMouseEvent [QtGui]	_ZN16QAbstractSpinBox16contextMenuEventEP17QContextMenuEvent [QtGui]
_ZN16QAbstractSpinBox16setButtonSymbolsENS_13ButtonSymbolsE [QtGui]	_ZN16QAbstractSpinBox17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN16QAbstractSpinBox17setCorrectionModeENS_14CorrectionModeE [QtGui]	_ZN16QAbstractSpinBox19setSpecialValueTextERK7QString [QtGui]
_ZN16QAbstractSpinBox5clearEv [QtGui]	_ZN16QAbstractSpinBox5eventEP6QEvent [QtGui]
_ZN16QAbstractSpinBox6stepByEi [QtGui]	_ZN16QAbstractSpinBox6stepUpEv [QtGui]
_ZN16QAbstractSpinBox8setFrameEb [QtGui]	_ZN16QAbstractSpinBox8stepDownEv [QtGui]
_ZN16QAbstractSpinBox9hideEventEP10QHideEvent [QtGui]	_ZN16QAbstractSpinBox9selectAllEv [QtGui]
_ZN16QAbstractSpinBox9showEventEP10QShowEvent [QtGui]	_ZN16QAbstractSpinBoxC1EP7QWidget [QtGui]
_ZN16QAbstractSpinBoxC2EP7QWidget [QtGui]	_ZN16QAbstractSpinBoxD0Ev [QtGui]
_ZN16QAbstractSpinBoxD1Ev [QtGui]	_ZN16QAbstractSpinBoxD2Ev [QtGui]
_ZN16QStringListModel10insertRowEiiRK11QModelIndex [QtGui]	_ZN16QStringListModel10removeRowEiiRK11QModelIndex [QtGui]

_ZN16QStringListModel11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN16QStringListModel11qt_metacastEPKc [QtGui]
_ZN16QStringListModel13setStringListERK11QStringList [QtGui]	_ZN16QStringListModel4sortEiN2Qt9SortOrderE [QtGui]
_ZN16QStringListModel7setDataERK11QModelIndexRK8QVarianti [QtGui]	_ZN16QStringListModelC1EP7QObject [QtGui]
_ZN16QStringListModelC1ERK11QStringListP7QObject [QtGui]	_ZN16QStringListModelC2EP7QObject [QtGui]
_ZN16QStringListModelC2ERK11QStringListP7QObject [QtGui]	_ZN16QTableWidgetItem4readER11QDataStream [QtGui]
_ZN16QTableWidgetItem7setDataEiRK8QVariant [QtGui]	_ZN16QTableWidgetItem8setFlagsE6QFlagsIN2Qt8ItemFlagEE [QtGui]
_ZN16QTableWidgetItemC1ERK5QIconRK7QStringi [QtGui]	_ZN16QTableWidgetItemC1ERK7QStringi [QtGui]
ZN16QTableWidgetItemC1ERKS [QtGui]	_ZN16QTableWidgetItemC1Ei [QtGui]
_ZN16QTableWidgetItemC2ERK5QIconRK7QStringi [QtGui]	_ZN16QTableWidgetItemC2ERK7QStringi [QtGui]
ZN16QTableWidgetItemC2ERKS [QtGui]	_ZN16QTableWidgetItemC2Ei [QtGui]
_ZN16QTableWidgetItemD0Ev [QtGui]	_ZN16QTableWidgetItemD1Ev [QtGui]
_ZN16QTableWidgetItemD2Ev [QtGui]	_ZN16QTableWidgetItemD3Ev [QtGui]
_ZN17QAbstractItemView10commitDataEP7QWidget [QtGui]	_ZN17QAbstractItemView10timerEventEP11QTimerEvent [QtGui]
_ZN17QAbstractItemView11closeEditorEP7QWidgetN21QAbstractItemDelegate11EndEditHintE [QtGui]	_ZN17QAbstractItemView11dataChangedERK11QModelIndexS2_ [QtGui]
_ZN17QAbstractItemView11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN17QAbstractItemView11qt_metacastEPKc [QtGui]
_ZN17QAbstractItemView11resizeEventEP12QResizeEvent [QtGui]	_ZN17QAbstractItemView11scrollToTopEv [QtGui]
_ZN17QAbstractItemView11setIconSizeERK5QSize [QtGui]	_ZN17QAbstractItemView12doAutoScrollEv [QtGui]
_ZN17QAbstractItemView12focusInEventEP11QFocusEvent [QtGui]	_ZN17QAbstractItemView12rowsInsertedERK11QModelIndexi [QtGui]
_ZN17QAbstractItemView12setRootIndexERK11QModelIndex [QtGui]	_ZN17QAbstractItemView13doItemsLayoutEv [LSB]

_ZN17QAbstractItemView13doubleClickedERK11QModelIndex [QtGui]	_ZN17QAbstractItemView13dragMoveEventEP14QDragMoveEvent [QtGui]
_ZN17QAbstractItemView13focusOutEventEP11QFocusEvent [QtGui]	_ZN17QAbstractItemView13keyPressEventEP9QKeyEvent [QtGui]
_ZN17QAbstractItemView13setAutoScrollEb [QtGui]	_ZN17QAbstractItemView13viewportEventEP6QEvent [QtGui]
_ZN17QAbstractItemView14clearSelectionEv [QtGui]	_ZN17QAbstractItemView14currentChangedERK11QModelIndexS2_ [QtGui]
_ZN17QAbstractItemView14dragEnterEventEP15QDragEnterEvent [QtGui]	_ZN17QAbstractItemView14dragLeaveEventEP15QDragLeaveEvent [QtGui]
_ZN17QAbstractItemView14keyboardSearchERK7QString [QtGui]	_ZN17QAbstractItemView14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN17QAbstractItemView14scrollToBottomEv [QtGui]	_ZN17QAbstractItemView14setDirtyRegionERK7QRegion [QtGui]
_ZN17QAbstractItemView14setDragEnabledEb [QtGui]	_ZN17QAbstractItemView14setIndexWidgetERK11QModelIndexP7QWidget [QtGui]
_ZN17QAbstractItemView14stopAutoScrollEv [QtGui]	_ZN17QAbstractItemView15editorDestroyedEP7QObject [QtGui]
_ZN17QAbstractItemView15mousePressEventEP11QMouseEvent [QtGui]	_ZN17QAbstractItemView15setCurrentIndexERK11QModelIndex [QtGui]
_ZN17QAbstractItemView15setDragDropModeENS_12DragDropModeE [QtXml]	_ZN17QAbstractItemView15setEditTriggersE6QFlagsINS_11EditTriggerEE [QtGui]
_ZN17QAbstractItemView15setItemDelegateEP21QAbstractItemDelegate [QtGui]	_ZN17QAbstractItemView15startAutoScrollEv [QtGui]
_ZN17QAbstractItemView15viewportEnteredEv [QtGui]	_ZN17QAbstractItemView16inputMethodEventEP17QInputMethodEvent [QtXml]
ZN17QAbstractItemView16selectionChangedERK14QItemSelectionS2 [QtGui]	_ZN17QAbstractItemView16setSelectionModeENS_13SelectionModeE [QtGui]
_ZN17QAbstractItemView16setTextElideModeEN2Qt13TextElideModeE [QtGui]	_ZN17QAbstractItemView16updateEditorDataEv [LSB]
_ZN17QAbstractItemView16updateGeometriesEv [QtGui]	_ZN17QAbstractItemView17mouseReleaseEventEP11QMouseEvent [QtGui]

_ZN17QAbstractItemView17scrollDirtyRegionEii [QtGui]	_ZN17QAbstractItemView17setSelectionModeEP19QItemSelectionModel [QtGui]
_ZN17QAbstractItemView18focusNextPrevChildEb [QtXml]	_ZN17QAbstractItemView19setTabKeyNavigationEb [QtGui]
_ZN17QAbstractItemView20openPersistentEditorERK11QModelIndex [QtGui]	_ZN17QAbstractItemView20rowsAboutToBeRemovedERK11QModelIndex [QtGui]
_ZN17QAbstractItemView20setSelectionBehaviorENS_17SelectionBehaviorE [QtGui]	_ZN17QAbstractItemView21closePersistentEditorERK11QModelIndex [QtGui]
_ZN17QAbstractItemView21mouseDoubleClickEventEP11QMouseEvent [QtGui]	_ZN17QAbstractItemView21setDropIndicatorShownEb [QtGui]
_ZN17QAbstractItemView21setItemDelegateForRowEiP21QAbstractItemDelegate [QtXml]	_ZN17QAbstractItemView21setVerticalScrollModeENS_10ScrollModeE [QtXml]
_ZN17QAbstractItemView22updateEditorGeometriesEv [LSB]	_ZN17QAbstractItemView23setAlternatingRowColorsEb [QtGui]
_ZN17QAbstractItemView23setHorizontalScrollModeENS_10ScrollModeE [QtXml]	_ZN17QAbstractItemView23setVerticalStepsPerItemEi [QtGui]
_ZN17QAbstractItemView23verticalScrollbarActionEi [LSB]	_ZN17QAbstractItemView24setDragDropOverwriteModeEb [QtXml]
_ZN17QAbstractItemView24setItemDelegateForColumnEiP21QAbstractItemDelegate [QtXml]	_ZN17QAbstractItemView25executeDelayedItemsLayoutEv [QtGui]
_ZN17QAbstractItemView25horizontalScrollbarActionEi [LSB]	_ZN17QAbstractItemView25setHorizontalStepsPerItemEi [QtGui]
_ZN17QAbstractItemView26scheduleDelayedItemsLayoutEv [QtGui]	_ZN17QAbstractItemView29verticalScrollbarValueChangedEi [QtGui]
_ZN17QAbstractItemView31horizontalScrollbarValueChangedEi [QtGui]	_ZN17QAbstractItemView4editERK11QModelIndex [QtGui]
_ZN17QAbstractItemView4editERK11QModelIndexNS_11EditTriggerEP6QEvent [QtGui]	_ZN17QAbstractItemView5eventEP6QEvent [QtGui]
_ZN17QAbstractItemView5resetEv [QtGui]	_ZN17QAbstractItemView7clickedERK11QModelIndex [QtGui]
_ZN17QAbstractItemView7enteredERK11QModelIndex [QtGui]	_ZN17QAbstractItemView7pressedERK11QModelIndex [QtGui]
_ZN17QAbstractItemView8setModelEP18QAbstractItemModel [QtGui]	_ZN17QAbstractItemView8setStateENS_5StateE [QtGui]
_ZN17QAbstractItemView9activatedERK11QModelIndex [QtGui]	_ZN17QAbstractItemView9dropEventEP10QDropEvent [QtGui]

_ZN17QAbstractItemView9selectAllEv [QtGui]	_ZN17QAbstractItemView9startDragE6QFlagsIN2Qt10DropActionEE [QtGui]
_ZN17QAbstractItemViewC1EP7QWidget [QtGui]	_ZN17QAbstractItemViewC2EP7QWidget [QtGui]
_ZN17QAbstractItemViewD0Ev [QtGui]	_ZN17QAbstractItemViewD1Ev [QtGui]
_ZN17QAbstractItemViewD2Ev [QtGui]	_ZN17QDataWidgetMapper10addMappingEP7QWidgeti [QtXml]
_ZN17QDataWidgetMapper10toPreviousEv [QtXml]	_ZN17QDataWidgetMapper11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN17QDataWidgetMapper11qt_metacastEPKc [QtXml]	_ZN17QDataWidgetMapper12clearMappingEv [QtXml]
_ZN17QDataWidgetMapper12setRootIndexERK11QModelIndex [QtXml]	_ZN17QDataWidgetMapper13removeMappingEP7QWidget [QtXml]
_ZN17QDataWidgetMapper14setOrientationEN2Qt11OrientationE [QtXml]	_ZN17QDataWidgetMapper15setCurrentIndexEi [QtXml]
_ZN17QDataWidgetMapper15setItemDelegateEP21QAbstractItemDelegate [QtXml]	_ZN17QDataWidgetMapper15setSubmitPolicyENS_12SubmitPolicyE [QtXml]
_ZN17QDataWidgetMapper19currentIndexChangedEi [QtXml]	_ZN17QDataWidgetMapper20setCurrentModelIndexERK11QModelIndex [QtXml]
_ZN17QDataWidgetMapper6revertEv [QtXml]	_ZN17QDataWidgetMapper6submitEv [QtXml]
_ZN17QDataWidgetMapper6toLastEv [QtXml]	_ZN17QDataWidgetMapper6toNextEv [QtXml]
_ZN17QDataWidgetMapper7toFirstEv [QtXml]	_ZN17QDataWidgetMapper8setModelEP18QAbstractItemModel [QtXml]
_ZN17QDataWidgetMapperC1EP7QObject [QtXml]	_ZN17QDataWidgetMapperC2EP7QObject [QtXml]
_ZN17QDataWidgetMapperD0Ev [QtXml]	_ZN17QDataWidgetMapperD1Ev [QtXml]
_ZN17QDataWidgetMapperD2Ev [QtXml]	_ZN18QItemEditorFactory14defaultFactoryEv [QtGui]
_ZN18QItemEditorFactory14registerEditorEN8QVariant4TypeEP22QItemEditorCreatorBase [QtGui]	_ZN18QItemEditorFactory17setDefaultFactoryEPS_ [QtGui]
_ZN18QItemEditorFactoryD0Ev [QtGui]	_ZN18QItemEditorFactoryD1Ev [QtGui]

_ZN18QItemEditorFactoryD2Ev [QtGui]	_ZN18QStandardItemModel10insertRowsEiiRK11QModelIndex [QtGui]
_ZN18QStandardItemModel10removeRowsEiiRK11QModelIndex [QtGui]	_ZN18QStandardItemModel10takeColumnEi [QtXml]
_ZN18QStandardItemModel11itemChangedEP13QStandardItem [QtXml]	_ZN18QStandardItemModel11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN18QStandardItemModel11qt_metacastEPKc [QtGui]	_ZN18QStandardItemModel11setItemDataERK11QModelIndexRK4QMapli8QVariantE [QtXml]
_ZN18QStandardItemModel11setRowCountEi [QtXml]	_ZN18QStandardItemModel11setSortRoleEi [QtXml]
_ZN18QStandardItemModel12appendColumnERK5QListIP13QStandardItemE [QtXml]	_ZN18QStandardItemModel12insertColumnEiRK5QListIP13QStandardItemE [QtXml]
_ZN18QStandardItemModel13insertColumnsEiiRK11QModelIndex [QtGui]	_ZN18QStandardItemModel13removeColumnsEiiRK11QModelIndex [QtGui]
_ZN18QStandardItemModel13setHeaderDataEiN2Qt11OrientationERK8QVarianti [QtGui]	_ZN18QStandardItemModel14setColumnCountEi [QtXml]
_ZN18QStandardItemModel16setItemPrototypeEPK13QStandardItem [QtXml]	_ZN18QStandardItemModel21setVerticalHeaderItemEiP13QStandardItem [QtXml]
_ZN18QStandardItemModel22takeVerticalHeaderItemEi [QtXml]	_ZN18QStandardItemModel23setHorizontalHeaderItemEiP13QStandardItem [QtXml]
_ZN18QStandardItemModel23setVerticalHeaderLabelsERK11QStringList [QtXml]	_ZN18QStandardItemModel24takeHorizontalHeaderItemEi [QtXml]
_ZN18QStandardItemModel25setHorizontalHeaderLabelsERK11QStringList [QtXml]	_ZN18QStandardItemModel4sortEiN2Qt9SortOrderE [QtXml]
_ZN18QStandardItemModel5clearEv [QtGui]	_ZN18QStandardItemModel7setDataERK11QModelIndexRK8QVarianti [QtGui]
_ZN18QStandardItemModel7setItemEiiP13QStandardItem [QtXml]	_ZN18QStandardItemModel7takeRowEi [QtXml]
_ZN18QStandardItemModel8takeItemEii [QtXml]	_ZN18QStandardItemModel9appendRowERK5QListIP13QStandardItemE [QtXml]
_ZN18QStandardItemModel9insertRowEiRK5QListIP13QStandardItemE [QtXml]	_ZN18QStandardItemModelC1EP7QObject [QtGui]

_ZN18QStandardItemModelC1EiiP7 QObject [QtGui]	_ZN18QStandardItemModelC2EP7Q Object [QtGui]
_ZN18QStandardItemModelC2EiiP7 QObject [QtGui]	_ZN18QStandardItemModelD0Ev [QtGui]
_ZN18QStandardItemModelD1Ev [QtGui]	_ZN18QStandardItemModelD2Ev [QtGui]
_ZN19QAbstractProxyModel11qt_m etacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN19QAbstractProxyModel11qt_m etacastEPKc [QtGui]
_ZN19QAbstractProxyModel14setSo urceModelEP18QAbstractItemModel [QtGui]	_ZN19QAbstractProxyModel6revert Ev [QtGui]
_ZN19QAbstractProxyModel6submit Ev [QtGui]	_ZN19QAbstractProxyModelC1EP7 QObject [QtGui]
_ZN19QAbstractProxyModelC2EP7 QObject [QtGui]	_ZN19QAbstractProxyModelD0Ev [QtGui]
_ZN19QAbstractProxyModelD1Ev [QtGui]	_ZN19QAbstractProxyModelD2Ev [QtGui]
_ZN19QAbstractScrollArea10paintEv entEP11QPaintEvent [QtGui]	_ZN19QAbstractScrollArea10wheelE ventEP11QWheelEvent [QtGui]
_ZN19QAbstractScrollArea11qt_met acallEN11QMetaObject4CallEiPPv [QtGui]	_ZN19QAbstractScrollArea11qt_met acastEPKc [QtGui]
_ZN19QAbstractScrollArea11resizeE ventEP12QResizeEvent [QtGui]	_ZN19QAbstractScrollArea11setVie wportEP7QWidget [QtGui]
_ZN19QAbstractScrollArea13dragM oveEventEP14QDragMoveEvent [QtGui]	_ZN19QAbstractScrollArea13keyPre ssEventEP9QKeyEvent [QtGui]
_ZN19QAbstractScrollArea13setupVi ewportEP7QWidget [QtGui]	_ZN19QAbstractScrollArea13viewpo rtEventEP6QEvent [QtGui]
_ZN19QAbstractScrollArea14dragEn terEventEP15QDragEnterEvent [QtGui]	_ZN19QAbstractScrollArea14dragLe aveEventEP15QDragLeaveEvent [QtGui]
_ZN19QAbstractScrollArea14mouse MoveEventEP11QMouseEvent [QtGui]	_ZN19QAbstractScrollArea15mouse PressEventEP11QMouseEvent [QtGui]
_ZN19QAbstractScrollArea15setCorn erWidgetEP7QWidget [QtGui]	_ZN19QAbstractScrollArea16context MenuEventEP17QContextMenuEven t [QtGui]
_ZN19QAbstractScrollArea16scrollB arWidgetsE6QFlagsIN2Qt13Alignme ntFlagEE [QtGui]	_ZN19QAbstractScrollArea16scrollC ontentsByEii [QtGui]

_ZN19QAbstractScrollArea17mouseReleaseEventEP11QMouseEvent [QtGui]	_ZN19QAbstractScrollArea18addScrollBarWidgetEP7QWidget6QFlagsIN2Qt13AlignmentFlagEE [QtXml]
_ZN19QAbstractScrollArea18setViewportMarginsEiiii [QtGui]	_ZN19QAbstractScrollArea20setVerticalScrollBarEP10QScrollBar [QtXml]
_ZN19QAbstractScrollArea21mouseDoubleClickEventEP11QMouseEvent [QtGui]	_ZN19QAbstractScrollArea22setHorizontalScrollBarEP10QScrollBar [QtXml]
_ZN19QAbstractScrollArea26setVerticalScrollBarPolicyEN2Qt15ScrollBarPolicyE [QtGui]	_ZN19QAbstractScrollArea28setHorizontalScrollBarPolicyEN2Qt15ScrollBarPolicyE [QtGui]
_ZN19QAbstractScrollArea5eventEP6QEvent [QtGui]	_ZN19QAbstractScrollArea9dropEventEP10QDropEvent [QtGui]
_ZN19QAbstractScrollAreaC1EP7QWidget [QtGui]	_ZN19QAbstractScrollAreaC2EP7QWidget [QtGui]
_ZN19QAbstractScrollAreaD0Ev [QtGui]	_ZN19QAbstractScrollAreaD1Ev [QtGui]
_ZN19QAbstractScrollAreaD2Ev [QtGui]	_ZN19QItemSelectionModel11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN19QItemSelectionModel11qt_metacastEPKc [QtGui]	_ZN19QItemSelectionModel14clearSelectionEv [QtXml]
ZN19QItemSelectionModel14currentChangedERK11QModelIndexS2 [QtGui]	_ZN19QItemSelectionModel15setCurrentIndexERK11QModelIndex6QFlagsINS_13SelectionFlagEE [QtGui]
ZN19QItemSelectionModel16selectionChangedERK14QItemSelectionS2 [QtGui]	_ZN19QItemSelectionModel17currentRowChangedERK11QModelIndexS2_ [QtGui]
ZN19QItemSelectionModel20currentColumnChangedERK11QModelIndexS2 [QtGui]	_ZN19QItemSelectionModel20emitSelectionChangedERK14QItemSelectionS2_ [QtGui]
_ZN19QItemSelectionModel5clearEv [QtGui]	_ZN19QItemSelectionModel5resetEv [QtGui]
_ZN19QItemSelectionModel6selectERK11QModelIndex6QFlagsINS_13SelectionFlagEE [QtGui]	_ZN19QItemSelectionModel6selectERK14QItemSelection6QFlagsINS_13SelectionFlagEE [QtGui]
_ZN19QItemSelectionModelC1EP18QAbstractItemModel [QtGui]	_ZN19QItemSelectionModelC1EP18QAbstractItemModelP7QObject [QtGui]
_ZN19QItemSelectionModelC2EP18QAbstractItemModel [QtGui]	_ZN19QItemSelectionModelC2EP18QAbstractItemModelP7QObject [QtGui]
_ZN19QItemSelectionModelD0Ev [QtGui]	_ZN19QItemSelectionModelD1Ev [QtGui]

_ZN19QItemSelectionModelD2Ev [QtGui]	_ZN20QAbstractPrintDialog11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN20QAbstractPrintDialog11qt_metacastEPKc [QtXml]	_ZN20QAbstractPrintDialog13setPrintRangeENS_10PrintRangeE [QtGui]
_ZN20QAbstractPrintDialog16addEnabledOptionENS_17PrintDialogOptionE [QtGui]	_ZN20QAbstractPrintDialog17setEnabledOptionsE6QFlagsINS_17PrintDialogOptionEE [QtGui]
_ZN20QAbstractPrintDialog9setFromToEii [QtGui]	_ZN20QAbstractPrintDialog9setMinMaxEii [QtGui]
_ZN20QAbstractPrintDialogC1EP8QPrinterP7QWidget [QtGui]	_ZN20QAbstractPrintDialogC2EP8QPrinterP7QWidget [QtGui]
_ZN21QAbstractItemDelegate10commitDataEP7QWidget [QtGui]	_ZN21QAbstractItemDelegate10elidedTextERK12QFontMetricsi12Qt13TextElideModeERK7QString [QtGui]
_ZN21QAbstractItemDelegate11closeEditorEP7QWidgetNS_11EndEditHintE [QtGui]	_ZN21QAbstractItemDelegate11editorEventEP6QEventP18QAbstractItemModelRK20QStyleOptionViewItemRK11QModelIndex [QtGui]
_ZN21QAbstractItemDelegate11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN21QAbstractItemDelegate11qt_metacastEPKc [QtGui]
_ZN21QAbstractItemDelegateC1EP7QObject [QtGui]	_ZN21QAbstractItemDelegateC2EP7QObject [QtGui]
_ZN21QAbstractItemDelegateD0Ev [QtGui]	_ZN21QAbstractItemDelegateD1Ev [QtGui]
_ZN21QAbstractItemDelegateD2Ev [QtGui]	_ZN21QSortFilterProxyModel10insertRowsEiiRK11QModelIndex [QtGui]
_ZN21QSortFilterProxyModel10removeRowsEiiRK11QModelIndex [QtGui]	_ZN21QSortFilterProxyModel11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN21QSortFilterProxyModel11qt_metacastEPKc [QtGui]	_ZN21QSortFilterProxyModel11setSortRoleEi [QtXml]
_ZN21QSortFilterProxyModel12dropMimeTypeDataEPK9QMimeTypeDataN2Qt10DropActionEiiRK11QModelIndex [QtGui]	_ZN21QSortFilterProxyModel13filterChangedEv [QtXml]
_ZN21QSortFilterProxyModel13insertColumnsEiiRK11QModelIndex [QtGui]	_ZN21QSortFilterProxyModel13removeColumnsEiiRK11QModelIndex [QtGui]
_ZN21QSortFilterProxyModel13setFilterRoleEi [QtXml]	_ZN21QSortFilterProxyModel13setHeaderDataEiN2Qt11OrientationERK8QVarianti [QtGui]

_ZN21QSortFilterProxyModel14setSourceModelEP18QAbstractItemModel [QtGui]	_ZN21QSortFilterProxyModel15setFilterRegExpERK7QRegExp [QtGui]
_ZN21QSortFilterProxyModel15setFilterRegExpERK7QString [QtGui]	_ZN21QSortFilterProxyModel17setFilterWildcardERK7QString [QtGui]
_ZN21QSortFilterProxyModel18setFilterKeyColumnEi [QtGui]	_ZN21QSortFilterProxyModel20setDynamicSortFilterEb [QtXml]
_ZN21QSortFilterProxyModel20setFilterFixedStringERK7QString [QtGui]	_ZN21QSortFilterProxyModel22setSortCaseSensitivityEN2Qt15CaseSensitivityE [QtXml]
_ZN21QSortFilterProxyModel24setFilterCaseSensitivityEN2Qt15CaseSensitivityE [QtGui]	_ZN21QSortFilterProxyModel4sortEiN2Qt9SortOrderE [QtGui]
_ZN21QSortFilterProxyModel5clearEv [QtGui]	_ZN21QSortFilterProxyModel7setDataERK11QModelIndexRK8QVarianti [QtGui]
_ZN21QSortFilterProxyModel9fetchMoreERK11QModelIndex [QtGui]	_ZN21QSortFilterProxyModelC1EP7QObject [QtGui]
_ZN21QSortFilterProxyModelC2EP7QObject [QtGui]	_ZN21QSortFilterProxyModelD0Ev [QtGui]
_ZN21QSortFilterProxyModelD1Ev [QtGui]	_ZN21QSortFilterProxyModelD2Ev [QtGui]
_ZN23QTreeWidgetItemIteratorC1EP11QTreeWidget6QFlagsINS_12IteratorFlagEE [QtGui]	_ZN23QTreeWidgetItemIteratorC1EP15QTreeWidgetItem6QFlagsINS_12IteratorFlagEE [QtGui]
ZN23QTreeWidgetItemIteratorC1ERKS [QtGui]	_ZN23QTreeWidgetItemIteratorC2EP11QTreeWidget6QFlagsINS_12IteratorFlagEE [QtGui]
_ZN23QTreeWidgetItemIteratorC2EP15QTreeWidgetItem6QFlagsINS_12IteratorFlagEE [QtGui]	_ZN23QTreeWidgetItemIteratorC2ERKS_ [QtGui]
_ZN23QTreeWidgetItemIteratorD1Ev [QtGui]	_ZN23QTreeWidgetItemIteratorD2Ev [QtGui]
ZN23QTreeWidgetItemIteratoraSERKS [QtGui]	_ZN23QTreeWidgetItemIteratormmEv [QtGui]
_ZN23QTreeWidgetItemIteratorppEv [QtGui]	_ZN24QAbstractPageSetupDialog11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN24QAbstractPageSetupDialog11qt_metacastEPKc [QtXml]	_ZN24QAbstractPageSetupDialog7printerEv [LSB]
_ZN24QAbstractPageSetupDialogC1EP8QPrinterP7QWidget [LSB]	_ZN24QAbstractPageSetupDialogC2EP8QPrinterP7QWidget [LSB]
_ZN26QAbstractGraphicsShapeItem6setPenERK4QPen [QtXml]	_ZN26QAbstractGraphicsShapeItem8setBrushERK6QBrush [QtXml]

_ZN26QAbstractGraphicsShapeItem C1EP13QGraphicsItemP14QGraphics Scene [QtXml]	_ZN26QAbstractGraphicsShapeItem C2EP13QGraphicsItemP14QGraphics Scene [QtXml]
_ZN26QAbstractGraphicsShapeItem D0Ev [QtXml]	_ZN26QAbstractGraphicsShapeItem D1Ev [QtXml]
_ZN26QAbstractGraphicsShapeItem D2Ev [QtXml]	_ZN26QTableWidgetSelectionRange C1ERKS_ [QtGui]
_ZN26QTableWidgetSelectionRange C1Eiiii [QtGui]	_ZN26QTableWidgetSelectionRange C1Ev [QtGui]
ZN26QTableWidgetSelectionRange C2ERKS [QtGui]	_ZN26QTableWidgetSelectionRange C2Eiiii [QtGui]
_ZN26QTableWidgetSelectionRange C2Ev [QtGui]	_ZN26QTableWidgetSelectionRange D1Ev [QtGui]
_ZN26QTableWidgetSelectionRange D2Ev [QtGui]	_ZN27QAbstractTextDocumentLayo ut11formatIndexEi [QtGui]
_ZN27QAbstractTextDocumentLayo ut11qt_metacallEN11QMetaObject4C alleiPPv [QtGui]	_ZN27QAbstractTextDocumentLayo ut11qt_metacastEPKc [QtGui]
_ZN27QAbstractTextDocumentLayo ut14setPaintDeviceEP12QPaintDevic e [QtGui]	_ZN27QAbstractTextDocumentLayo ut15registerHandlerEiP7QObject [QtGui]
_ZN27QAbstractTextDocumentLayo ut16drawInlineObjectEP8QPainterR K6QRectF17QTextInlineObjectiRK11 QTextFormat [QtGui]	_ZN27QAbstractTextDocumentLayo ut16pageCountChangedEi [QtGui]
_ZN27QAbstractTextDocumentLayo ut18resizeInlineObjectE17QTextInlin eObjectiRK11QTextFormat [QtGui]	_ZN27QAbstractTextDocumentLayo ut19documentSizeChangedERK6QSi zeF [QtGui]
_ZN27QAbstractTextDocumentLayo ut20positionInlineObjectE17QTextInl ineObjectiRK11QTextFormat [QtGui]	_ZN27QAbstractTextDocumentLayo ut6formatEi [QtGui]
_ZN27QAbstractTextDocumentLayo ut6updateERK6QRectF [QtGui]	_ZN27QAbstractTextDocumentLayo utC1EP13QTextDocument [QtGui]
_ZN27QAbstractTextDocumentLayo utC2EP13QTextDocument [QtGui]	_ZN27QAbstractTextDocumentLayo utD0Ev [QtGui]
_ZN27QAbstractTextDocumentLayo utD1Ev [QtGui]	_ZN27QAbstractTextDocumentLayo utD2Ev [QtGui]
_ZN9QDirModel10setSortingE6QFla gsIN4QDir8SortFlagEE [QtGui]	_ZN9QDirModel11qt_metacallEN11 QMetaObject4CallEiPPv [QtGui]
_ZN9QDirModel11qt_metacastEPKc [QtGui]	_ZN9QDirModel11setReadOnlyEb [QtGui]
_ZN9QDirModel12dropMimeDataE PK9QMimeDataN2Qt10DropActionE iiRK11QModelIndex [QtGui]	_ZN9QDirModel14setNameFiltersER K11QStringList [QtGui]

_ZN9QDirModel15setIconProviderEP17QFileIconProvider [QtGui]	_ZN9QDirModel17setLazyChildCountEb [QtGui]
_ZN9QDirModel18setResolveSymlinksEb [QtGui]	_ZN9QDirModel4sortEiN2Qt9SortOrderE [QtGui]
_ZN9QDirModel5mkdirERK11QModelIndexRK7QString [QtGui]	_ZN9QDirModel5rmdirERK11QModelIndex [QtGui]
_ZN9QDirModel6removeERK11QModelIndex [QtGui]	_ZN9QDirModel7refreshERK11QModelIndex [QtGui]
_ZN9QDirModel7setDataERK11QModelIndexRK8QVarianti [QtGui]	_ZN9QDirModel9setFilterE6QFlagsIN4QDir6FilterEE [QtGui]
_ZN9QDirModelC1EP7QObject [QtGui]	_ZN9QDirModelC1ERK11QStringList6QFlagsIN4QDir6FilterEES3_INS4_8SortFlagEEP7QObject [QtGui]
_ZN9QDirModelC2EP7QObject [QtGui]	_ZN9QDirModelC2ERK11QStringList6QFlagsIN4QDir6FilterEES3_INS4_8SortFlagEEP7QObject [QtGui]
_ZN9QDirModelD0Ev [QtGui]	_ZN9QDirModelD1Ev [QtGui]
_ZN9QDirModelD2Ev [QtGui]	_ZN9QTreeView10hideColumnEi [QtGui]
_ZN9QTreeView10moveCursorEN17QAbstractItemView12CursorActionE6QFlagsIN2Qt16KeyboardModifierEE [QtGui]	_ZN9QTreeView10paintEventEP11QPaintEvent [QtGui]
_ZN9QTreeView10showColumnEi [QtGui]	_ZN9QTreeView10timerEventEP11QTimerEvent [QtGui]
_ZN9QTreeView11collapseAllEv [QtGui]	_ZN9QTreeView11columnMovedEv [QtGui]
ZN9QTreeView11dataChangedERK11QModelIndexS2 [QtGui]	_ZN9QTreeView11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QTreeView11qt_metacastEPKc [QtGui]	_ZN9QTreeView11rowsRemovedERK11QModelIndexi [QtGui]
_ZN9QTreeView11setAnimatedEb [QtGui]	_ZN9QTreeView11setExpandedERK11QModelIndexb [QtGui]
_ZN9QTreeView12rowsInsertedERK11QModelIndexi [QtGui]	_ZN9QTreeView12setRootIndexERK11QModelIndex [QtGui]
_ZN9QTreeView12setRowHiddenEiRK11QModelIndexb [QtGui]	_ZN9QTreeView12setSelectionERK5QRect6QFlagsIN19QItemSelectionMode13SelectionFlagEE [QtGui]
_ZN9QTreeView12sortByColumnEi [QtGui]	_ZN9QTreeView12sortByColumnEiN2Qt9SortOrderE [QtGui]
_ZN9QTreeView13columnResizedEiii [QtGui]	_ZN9QTreeView13doItemsLayoutEv [QtGui]

_ZN9QTreeView13keyPressEventEP9QKeyEvent [QtXml]	_ZN9QTreeView14keyboardSearchERK7QString [QtGui]
_ZN9QTreeView14mouseMoveEventEP11QMouseEvent [QtXml]	_ZN9QTreeView14setColumnWidthEii [QtXml]
_ZN9QTreeView14setIndentationEi [QtGui]	_ZN9QTreeView15mousePressEventEP11QMouseEvent [QtGui]
_ZN9QTreeView15setColumnHiddenEib [QtGui]	_ZN9QTreeView16scrollContentsByEii [QtGui]
_ZN9QTreeView16updateGeometriesEv [QtGui]	_ZN9QTreeView17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN9QTreeView17setSelectionModelEP19QItemSelectionModel [QtGui]	_ZN9QTreeView17setSortingEnabledEb [QtXml]
_ZN9QTreeView18columnCountChangedEii [QtGui]	_ZN9QTreeView18setItemsExpandableEb [QtGui]
_ZN9QTreeView18setRootIsDecoratedEb [QtGui]	_ZN9QTreeView20rowsAboutToBeRemovedERK11QModelIndexii [QtGui]
_ZN9QTreeView20setUniformRowHeightsEb [QtGui]	_ZN9QTreeView21mouseDoubleClickEventEP11QMouseEvent [QtGui]
_ZN9QTreeView22resizeColumnToContentsEi [QtGui]	_ZN9QTreeView22setAllColumnsShowFocusEb [QtXml]
_ZN9QTreeView25horizontalScrollBarActionEi [QtGui]	_ZN9QTreeView5resetEv [QtGui]
_ZN9QTreeView6expandERK11QModelIndex [QtGui]	_ZN9QTreeView8collapseERK11QModelIndex [QtGui]
_ZN9QTreeView8expandedERK11QModelIndex [QtGui]	_ZN9QTreeView8reexpandEv [QtGui]
_ZN9QTreeView8scrollToERK11QModelIndexN17QAbstractItemView10ScrollHintE [QtGui]	_ZN9QTreeView8setModelEP18QAbstractItemModel [QtGui]
_ZN9QTreeView9collapsedERK11QModelIndex [QtGui]	_ZN9QTreeView9expandAllEv [QtXml]
_ZN9QTreeView9selectAllEv [QtGui]	_ZN9QTreeView9setHeaderEP11QHeaderView [QtGui]
_ZN9QTreeViewC1EP7QWidget [QtGui]	_ZN9QTreeViewC2EP7QWidget [QtGui]
_ZN9QTreeViewD0Ev [QtGui]	_ZN9QTreeViewD1Ev [QtGui]
_ZN9QTreeViewD2Ev [QtGui]	_ZNK10QCompleter10currentRowEv [QtXml]
_ZNK10QCompleter10metaObjectEv [QtXml]	_ZNK10QCompleter12currentIndexEv [QtXml]

_Znk10QCompleter12modelSortingEv [QtXml]	_Znk10QCompleter13pathFromIndexERK11QModelIndex [QtXml]
_Znk10QCompleter14completionModeEv [QtXml]	_Znk10QCompleter14completionRoleEv [QtXml]
_Znk10QCompleter15caseSensitivityEv [QtXml]	_Znk10QCompleter15completionCountEv [QtXml]
_Znk10QCompleter15completionModelEv [QtXml]	_Znk10QCompleter16completionColumnEv [QtXml]
_Znk10QCompleter16completionPrefixEv [QtXml]	_Znk10QCompleter17currentCompletionEv [QtXml]
_Znk10QCompleter5modelEv [QtXml]	_Znk10QCompleter5popupEv [QtXml]
_Znk10QCompleter6widgetEv [QtXml]	_Znk10QCompleter9splitPathERK7QString [QtXml]
_Znk10QTableView10columnSpanEii [QtGui]	_Znk10QTableView10metaObjectEv [QtGui]
_Znk10QTableView10visualRectERK11QModelIndex [QtGui]	_Znk10QTableView11columnWidthEi [QtGui]
_Znk10QTableView11isRowHiddenEi [QtGui]	_Znk10QTableView11viewOptionsEv [QtGui]
_Znk10QTableView13isIndexHiddenERK11QModelIndex [QtGui]	_Znk10QTableView14isColumnHiddenEi [QtGui]
_Znk10QTableView14sizeHintForRowEi [QtGui]	_Znk10QTableView14verticalHeaderEv [QtGui]
_Znk10QTableView14verticalOffsetEv [QtGui]	_Znk10QTableView15selectedIndexEv [QtGui]
_Znk10QTableView16horizontalHeaderEv [QtGui]	_Znk10QTableView16horizontalOffsetEv [QtGui]
_Znk10QTableView16isSortingEnabledEv [QtGui]	_Znk10QTableView17sizeHintForColumnEi [QtGui]
_Znk10QTableView19rowViewportPositionEi [QtGui]	_Znk10QTableView22columnViewportPositionEi [QtGui]
_Znk10QTableView24visualRegionForSelectionERK14QItemSelection [QtGui]	_Znk10QTableView5rowAtEi [QtGui]
_Znk10QTableView7indexAtERK6QPoint [QtGui]	_Znk10QTableView7rowSpanEii [QtGui]
_Znk10QTableView8columnAtEi [QtGui]	_Znk10QTableView8showGridEv [QtGui]
_Znk10QTableView9gridStyleEv [QtGui]	_Znk10QTableView9rowHeightEi [QtGui]

_ZNK11QHeaderView10metaObjectEv [QtGui]	_ZNK11QHeaderView10resizeModeEi [QtGui]
_ZNK11QHeaderView10visualRectERK11QModelIndex [QtGui]	_ZNK11QHeaderView11isClickableEv [QtGui]
_ZNK11QHeaderView11orientationEv [QtGui]	_ZNK11QHeaderView11sectionSizeEi [QtGui]
_ZNK11QHeaderView11visualIndexEi [QtGui]	_ZNK11QHeaderView12logicalIndexEi [QtGui]
_ZNK11QHeaderView12paintSectionEP8QPainterRK5QRecti [QtGui]	_ZNK11QHeaderView13isIndexHiddenERK11QModelIndex [QtGui]
_ZNK11QHeaderView13sectionsMovedEv [QtGui]	_ZNK11QHeaderView13visualIndexAtEi [QtGui]
_ZNK11QHeaderView14logicalIndexAtEi [QtGui]	_ZNK11QHeaderView14sectionsHiddenEv [QtGui]
_ZNK11QHeaderView14verticalOffsetEv [QtGui]	_ZNK11QHeaderView15isSectionHiddenEi [QtGui]
_ZNK11QHeaderView15sectionPositionEi [QtGui]	_ZNK11QHeaderView15sectionSizeHintEi [QtGui]
_ZNK11QHeaderView16defaultAlignmentEv [QtGui]	_ZNK11QHeaderView16horizontalOffsetEv [QtGui]
_ZNK11QHeaderView17highlightSectionsEv [QtGui]	_ZNK11QHeaderView18defaultSectionSizeEv [QtGui]
_ZNK11QHeaderView18hiddenSectionCountEv [QtGui]	_ZNK11QHeaderView18minimumSectionSizeEv [QtXml]
_ZNK11QHeaderView18sortIndicatorOrderEv [QtGui]	_ZNK11QHeaderView18stretchLastSectionEv [QtGui]
_ZNK11QHeaderView19stretchSectionCountEv [QtGui]	_ZNK11QHeaderView20isSortIndicatorShownEv [QtGui]
_ZNK11QHeaderView20sortIndicatorSectionEv [QtGui]	_ZNK11QHeaderView23cascadingSectionResizesEv [QtXml]
_ZNK11QHeaderView23sectionSizeFromContentsEi [QtGui]	_ZNK11QHeaderView23sectionViewportPositionEi [QtGui]
_ZNK11QHeaderView24visualRegionForSelectionERK14QItemSelection [QtGui]	_ZNK11QHeaderView5countEv [QtGui]
_ZNK11QHeaderView6lengthEv [QtGui]	_ZNK11QHeaderView6offsetEv [QtGui]
_ZNK11QHeaderView7indexAtERK6QPoint [QtGui]	_ZNK11QHeaderView8sizeHintEv [QtGui]
_ZNK11QHeaderView9isMovableEv [QtGui]	_ZNK11QProxyModel10headerDataEiN2Qt11OrientationEi [QtGui]

_Znk11QProxyModel10metaObjectEv [QtGui]	_Znk11QProxyModel11columnCountERK11QModelIndex [QtGui]
_Znk11QProxyModel11hasChildrenERK11QModelIndex [QtGui]	_Znk11QProxyModel13setProxyModelERK11QModelIndex [LSB]
_Znk11QProxyModel14connectToModelEPK18QAbstractItemModel [LSB]	_Znk11QProxyModel14setSourceModelERK11QModelIndex [LSB]
_Znk11QProxyModel19disconnectFromModelEPK18QAbstractItemModel [LSB]	_Znk11QProxyModel20supportedDropActionsEv [QtGui]
_Znk11QProxyModel4dataERK11QModelIndex [QtGui]	_Znk11QProxyModel4spanERK11QModelIndex [QtGui]
_Znk11QProxyModel5flagsERK11QModelIndex [QtGui]	_Znk11QProxyModel5indexErrK11QModelIndex [QtGui]
_Znk11QProxyModel5matchERK11QModelIndexiRK8QVarianti6QFlagsIN2Qt9MatchFlagEE [QtGui]	_Znk11QProxyModel5modelEv [QtGui]
_Znk11QProxyModel6parentERK11QModelIndex [QtGui]	_Znk11QProxyModel8mimeDataERK5QList11QModelIndexE [QtGui]
_Znk11QProxyModel8rowCountERK11QModelIndex [QtGui]	_Znk11QProxyModel9mimeTypesEv [QtGui]
_Znk11QTreeWidget10headerItemEv [QtGui]	_Znk11QTreeWidget10itemWidgetEP15QTreeWidgetItem [QtGui]
_Znk11QTreeWidget10metaObjectEv [QtGui]	_Znk11QTreeWidget10sortColumnEv [QtGui]
_Znk11QTreeWidget11columnCountEv [QtGui]	_Znk11QTreeWidget11currentItemEv [QtGui]
_Znk11QTreeWidget12isItemHiddenEPK15QTreeWidgetItem [QtGui]	_Znk11QTreeWidget12topLevelItemEi [QtGui]
_Znk11QTreeWidget13currentColumnEv [QtGui]	_Znk11QTreeWidget13indexFromItemEP15QTreeWidgetItem [QtGui]
_Znk11QTreeWidget13itemFromIndexERK11QModelIndex [QtGui]	_Znk11QTreeWidget13selectedItemsEv [QtGui]
_Znk11QTreeWidget14isItemExpandedEPK15QTreeWidgetItem [QtGui]	_Znk11QTreeWidget14isItemSelectedEPK15QTreeWidgetItem [QtGui]
_Znk11QTreeWidget14visualItemRectEPK15QTreeWidgetItem [QtGui]	_Znk11QTreeWidget16isSortingEnabledEv [QtGui]
_Znk11QTreeWidget17invisibleRootItemEv [QtXml]	_Znk11QTreeWidget17topLevelItemCountEv [QtGui]
_Znk11QTreeWidget19indexOfTopLevelItemEP15QTreeWidgetItem [QtXml]	_Znk11QTreeWidget20supportedDropActionsEv [QtGui]

_Znk11QTreeWidget5itemsEPK9Q MimeData [QtGui]	_Znk11QTreeWidget6itemAtERK6Q Point [QtGui]
_Znk11QTreeWidget8mimeDataE5 QListIP15QTreeWidgetItemE [QtGui]	_Znk11QTreeWidget9findItemsERK 7QString6QFlagsIN2Qt9MatchFlagE Ei [QtGui]
_Znk11QTreeWidget9mimeTypesEv [QtGui]	_Znk12QTableWidget10cellWidgetE ii [QtGui]
_Znk12QTableWidget10currentRow Ev [QtGui]	_Znk12QTableWidget10metaObject Ev [QtGui]
_Znk12QTableWidget11columnCou ntEv [QtGui]	_Znk12QTableWidget11currentItem Ev [QtGui]
_Znk12QTableWidget12visualColu mnEi [QtGui]	_Znk12QTableWidget13currentColu mnEv [QtGui]
_Znk12QTableWidget13indexFromI temEPK16QTableWidgetItem [QtGui]	_Znk12QTableWidget13itemFromIn dexERK11QModelIndex [QtGui]
_Znk12QTableWidget13itemPrototy peEv [QtGui]	_Znk12QTableWidget14isItemSelectedEPK16QTableWidgetItem [QtGui]
_Znk12QTableWidget14selectedRan gesEv [QtGui]	_Znk12QTableWidget14visualItemR ectEPK16QTableWidgetItem [QtGui]
_Znk12QTableWidget16isSortingEn abledEv [QtGui]	_Znk12QTableWidget18verticalHea derItemEi [QtGui]
_Znk12QTableWidget20horizontalH eaderItemEi [QtGui]	_Znk12QTableWidget20supportedD ropActionsEv [QtGui]
_Znk12QTableWidget3rowEPK16Q TableWidgetItem [QtGui]	_Znk12QTableWidget4itemEii [QtGui]
_Znk12QTableWidget5itemsEPK9Q MimeData [QtGui]	_Znk12QTableWidget6columnEPK1 6QTableWidgetItem [QtGui]
_Znk12QTableWidget6itemAtERK6 QPoint [QtGui]	_Znk12QTableWidget8mimeDataE5 QListIP16QTableWidgetItemE [QtGui]
_Znk12QTableWidget8rowCountEv [QtGui]	_Znk12QTableWidget9findItemsER K7QString6QFlagsIN2Qt9MatchFlag EE [QtGui]
_Znk12QTableWidget9mimeTypesE v [QtGui]	_Znk12QTableWidget9visualRowEi [QtGui]
_Znk13QItemDelegate10decoration ERK20QStyleOptionViewItemRK8Q Variant [LSB]	_Znk13QItemDelegate10metaObject Ev [QtGui]
_Znk13QItemDelegate10setOptions ERK11QModelIndexRK20QStyleOpti onViewItem [QtXml]	_Znk13QItemDelegate11drawDispla yEP8QPainterRK20QStyleOptionVie wItemRK5QRectRK7QString [QtGui]

_ZNK13QItemDelegate11hasClippingEv [QtXml]	_ZNK13QItemDelegate12createEditorEP7QWidgetRK20QStyleOptionViewItemRK11QModelIndex [QtGui]
_ZNK13QItemDelegate12setModelDataEP7QWidgetP18QAbstractItemModelIndexRK11QModelIndex [QtGui]	_ZNK13QItemDelegate13setEditorDataEP7QWidgetRK11QModelIndex [QtGui]
_ZNK13QItemDelegate13textRectangleEP8QPainterRK5QRectRK5QFontRK7QString [QtXml]	_ZNK13QItemDelegate14drawBackgroundEP8QPainterRK20QStyleOptionViewItemRK11QModelIndex [QtXml]
_ZNK13QItemDelegate14drawDecorationEP8QPainterRK20QStyleOptionViewItemRK5QRectRK7QPixmap [QtGui]	_ZNK13QItemDelegate17itemEditorFactoryEv [QtGui]
_ZNK13QItemDelegate20updateEditorGeometryEP7QWidgetRK20QStyleOptionViewItemRK11QModelIndex [QtGui]	_ZNK13QItemDelegate4rectERK20QStyleOptionViewItemRK11QModelIndex [QtXml]
_ZNK13QItemDelegate5checkERK20QStyleOptionViewItemRK5QRectRK8QVariant [QtGui]	_ZNK13QItemDelegate5paintEP8QPainterRK20QStyleOptionViewItemRK11QModelIndex [QtGui]
_ZNK13QItemDelegate8doLayoutERK20QStyleOptionViewItemP5QRectS4_S4_b [QtGui]	_ZNK13QItemDelegate8selectedERK7QPixmapRK8QPaletteb [LSB]
_ZNK13QItemDelegate8sizeHintERK20QStyleOptionViewItemRK11QModelIndex [QtGui]	_ZNK13QItemDelegate9drawCheckEP8QPainterRK20QStyleOptionViewItemRK5QRectN2Qt10CheckStateE [QtGui]
_ZNK13QItemDelegate9drawFocusEP8QPainterRK20QStyleOptionViewItemRK5QRect [QtGui]	_ZNK13QStandardItem11columnCountEv [QtXml]
_ZNK13QStandardItem11hasChildrenEv [QtXml]	_ZNK13QStandardItem3rowEv [QtXml]
_ZNK13QStandardItem4dataEi [QtXml]	_ZNK13QStandardItem4typeEv [QtXml]
_ZNK13QStandardItem5childEii [QtXml]	_ZNK13QStandardItem5cloneEv [QtXml]
_ZNK13QStandardItem5flagsEv [QtXml]	_ZNK13QStandardItem5indexEv [QtXml]
_ZNK13QStandardItem5modelEv [QtXml]	_ZNK13QStandardItem5writeER11QDataStream [QtXml]
_ZNK13QStandardItem6columnEv [QtXml]	_ZNK13QStandardItem6parentEv [QtXml]
_ZNK13QStandardItem8rowCountEv [QtXml]	_ZNK13QStandardItemltERKS_ [QtXml]

_ZNK14QItemSelection7indexesEv [QtGui]	_ZNK14QItemSelection8containsERK11QModelIndex [QtGui]
_ZNK15QAbstractButton10autoRepeatEv [QtGui]	_ZNK15QAbstractButton10metaObjectEv [QtGui]
_ZNK15QAbstractButton11isCheckedEv [QtGui]	_ZNK15QAbstractButton13autoExclusiveEv [QtGui]
_ZNK15QAbstractButton15autoRepeatDelayEv [QtXml]	_ZNK15QAbstractButton18autoRepeatIntervalEv [QtXml]
_ZNK15QAbstractButton4iconEv [QtGui]	_ZNK15QAbstractButton4textEv [QtGui]
_ZNK15QAbstractButton5groupEv [QtGui]	_ZNK15QAbstractButton6isDownEv [QtGui]
_ZNK15QAbstractButton7iconSetEv [QtGui]	_ZNK15QAbstractButton8iconSizeEv [QtGui]
_ZNK15QAbstractButton8shortcutEv [QtGui]	_ZNK15QAbstractButton9hitButtonERK6QPoint [QtGui]
_ZNK15QAbstractButton9isCheckedEv [QtGui]	_ZNK15QAbstractSlider10metaObjectEv [QtGui]
_ZNK15QAbstractSlider10singleStepEv [QtGui]	_ZNK15QAbstractSlider11hasTrackingEv [QtGui]
_ZNK15QAbstractSlider11orientationEv [QtGui]	_ZNK15QAbstractSlider12isSliderDownEv [QtGui]
_ZNK15QAbstractSlider12repeatActionEv [QtGui]	_ZNK15QAbstractSlider14sliderPositionEv [QtGui]
_ZNK15QAbstractSlider16invertedControlsEv [QtGui]	_ZNK15QAbstractSlider18invertedAppearanceEv [QtGui]
_ZNK15QAbstractSlider5valueEv [QtGui]	_ZNK15QAbstractSlider7maximumEv [QtGui]
_ZNK15QAbstractSlider7minimumEv [QtGui]	_ZNK15QAbstractSlider8pageStepEv [QtGui]
_ZNK15QTreeWidgetItem4dataEii [QtGui]	_ZNK15QTreeWidgetItem5cloneEv [QtGui]
_ZNK15QTreeWidgetItem5writeER11QDataStream [QtGui]	_ZNK15QTreeWidgetItemltERKS_ [QtGui]
_ZNK16QAbstractSpinBox10isReadOnlyEv [QtGui]	_ZNK16QAbstractSpinBox10metaObjectEv [QtGui]
_ZNK16QAbstractSpinBox11stepEnabledEv [QtGui]	_ZNK16QAbstractSpinBox13buttonSymbolsEv [QtGui]
_ZNK16QAbstractSpinBox13isAcceleratedEv [QtXml]	_ZNK16QAbstractSpinBox14correctionModeEv [QtXml]
_ZNK16QAbstractSpinBox15minimumSizeHintEv [QtGui]	_ZNK16QAbstractSpinBox16specialValueTextEv [QtGui]

_ZNK16QAbstractSpinBox18hasAcceptableInputEv [QtXml]	_ZNK16QAbstractSpinBox4textEv [QtGui]
_ZNK16QAbstractSpinBox5fixupER7QString [QtGui]	_ZNK16QAbstractSpinBox8hasFrameEv [QtGui]
_ZNK16QAbstractSpinBox8lineEditEv [QtGui]	_ZNK16QAbstractSpinBox8sizeHintEv [QtGui]
_ZNK16QAbstractSpinBox8validateER7QStringRi [QtGui]	_ZNK16QAbstractSpinBox8wrappingEv [QtGui]
_ZNK16QAbstractSpinBox9alignmentEv [QtGui]	_ZNK16QStringListModel10metaObjectEv [QtGui]
_ZNK16QStringListModel10stringListEv [QtGui]	_ZNK16QStringListModel4dataERK11QModelIndex [QtGui]
_ZNK16QStringListModel5flagsERK11QModelIndex [QtGui]	_ZNK16QStringListModel8rowCountERK11QModelIndex [QtGui]
_ZNK16QTableWidgetItem4dataEi [QtGui]	_ZNK16QTableWidgetItem5cloneEv [QtGui]
_ZNK16QTableWidgetItem5writeER11QDataStream [QtGui]	_ZNK16QTableWidgetItemltERKS_ [QtGui]
_ZNK17QAbstractItemView10metaObjectEv [QtGui]	_ZNK17QAbstractItemView11dragEnabledEv [QtGui]
_ZNK17QAbstractItemView11indexWidgetERK11QModelIndex [QtGui]	_ZNK17QAbstractItemView11viewOptionsEv [QtGui]
_ZNK17QAbstractItemView12currentIndexEv [QtGui]	_ZNK17QAbstractItemView12dragDropModeEv [QtXml]
_ZNK17QAbstractItemView12editTriggersEv [QtGui]	_ZNK17QAbstractItemView12itemDelegateERK11QModelIndex [QtXml]
_ZNK17QAbstractItemView12itemDelegateEv [QtGui]	_ZNK17QAbstractItemView13hasAutoScrollEv [QtGui]
_ZNK17QAbstractItemView13selectionModeEv [QtGui]	_ZNK17QAbstractItemView13textEditModeEv [QtGui]
_ZNK17QAbstractItemView14selectionModelEv [QtGui]	_ZNK17QAbstractItemView14sizeHintForRowEi [QtGui]
_ZNK17QAbstractItemView15selectedIndexesEv [QtGui]	_ZNK17QAbstractItemView16inputMethodQueryEN2Qt16InputMethodQueryE [QtXml]
_ZNK17QAbstractItemView16selectionCommandERK11QModelIndexPK6QEvent [QtGui]	_ZNK17QAbstractItemView16sizeHintForIndexERK11QModelIndex [QtGui]
_ZNK17QAbstractItemView16tabKeyNavigationEv [QtGui]	_ZNK17QAbstractItemView17dirtyRegionOffsetEv [QtGui]
_ZNK17QAbstractItemView17selectionBehaviorEv [QtGui]	_ZNK17QAbstractItemView17showDropIndicatorEv [QtGui]

_ZNK17QAbstractItemView17sizeHintForColumnEi [QtGui]	_ZNK17QAbstractItemView18itemDelegateForRowEi [QtXml]
_ZNK17QAbstractItemView18verticalScrollModeEv [QtXml]	_ZNK17QAbstractItemView20alternatingRowColorsEv [QtGui]
_ZNK17QAbstractItemView20horizontalScrollModeEv [QtXml]	_ZNK17QAbstractItemView20verticalStepsPerItemEv [QtGui]
_ZNK17QAbstractItemView21dragDropOverwriteModeEv [QtXml]	_ZNK17QAbstractItemView21dropIndicatorPositionEv [QtGui]
_ZNK17QAbstractItemView21itemDelegateForColumnEi [QtXml]	_ZNK17QAbstractItemView22horizontalStepsPerItemEv [QtGui]
_ZNK17QAbstractItemView5modelEv [QtGui]	_ZNK17QAbstractItemView5stateEv [QtGui]
_ZNK17QAbstractItemView8iconSizeEv [QtGui]	_ZNK17QAbstractItemView9rootIndexEv [QtGui]
_ZNK17QDataWidgetMapper10metaObjectEv [QtXml]	_ZNK17QDataWidgetMapper11orientationEv [QtXml]
_ZNK17QDataWidgetMapper12currentIndexEv [QtXml]	_ZNK17QDataWidgetMapper12itemDelegateEv [QtXml]
_ZNK17QDataWidgetMapper12submitPolicyEv [QtXml]	_ZNK17QDataWidgetMapper13mappedSectionEP7QWidget [QtXml]
_ZNK17QDataWidgetMapper14mappedWidgetAtEi [QtXml]	_ZNK17QDataWidgetMapper5modelEv [QtXml]
_ZNK17QDataWidgetMapper9rootIndexEv [QtXml]	_ZNK18QItemEditorFactory12createEditorEN8QVariant4TypeEP7QWidget [QtGui]
_ZNK18QItemEditorFactory17valuePropertyNameEN8QVariant4TypeE [QtGui]	_ZNK18QStandardItemModel10headerDataEiN2Qt11OrientationEi [QtGui]
_ZNK18QStandardItemModel10metaObjectEv [QtGui]	_ZNK18QStandardItemModel11columnCountERK11QModelIndex [QtGui]
_ZNK18QStandardItemModel11hasChildrenERK11QModelIndex [QtGui]	_ZNK18QStandardItemModel13indexFromItemEPK13QStandardItem [QtXml]
_ZNK18QStandardItemModel13itemFromIndexERK11QModelIndex [QtXml]	_ZNK18QStandardItemModel13itemPrototypeEv [QtXml]
_ZNK18QStandardItemModel17invisibleRootItemEv [QtXml]	_ZNK18QStandardItemModel18verticalHeaderItemEi [QtXml]
_ZNK18QStandardItemModel20horizontalHeaderItemEi [QtXml]	_ZNK18QStandardItemModel20supportedDropActionsEv [QtXml]
_ZNK18QStandardItemModel4dataERK11QModelIndex [QtGui]	_ZNK18QStandardItemModel4itemEi [QtXml]

_Znk18QStandardItemModel5flags ERK11QModelIndex [QtGui]	_Znk18QStandardItemModel5index EiiRK11QModelIndex [QtGui]
_Znk18QStandardItemModel6paren tERK11QModelIndex [QtGui]	_Znk18QStandardItemModel8item DataERK11QModelIndex [QtXml]
_Znk18QStandardItemModel8rowC ountERK11QModelIndex [QtGui]	_Znk18QStandardItemModel8sortR oleEv [QtXml]
_Znk18QStandardItemModel9findIt emsERK7QString6QFlagsIN2Qt9Mat chFlagEEi [QtXml]	_Znk19QAbstractProxyModel10hea derDataEiN2Qt11OrientationEi [QtXml]
_Znk19QAbstractProxyModel10met aObjectEv [QtGui]	_Znk19QAbstractProxyModel11sou rceModelEv [QtGui]
_Znk19QAbstractProxyModel20ma pSelectionToSourceERK14QItemSele ction [QtGui]	_Znk19QAbstractProxyModel22ma pSelectionFromSourceERK14QItemS election [QtGui]
_Znk19QAbstractProxyModel4data ERK11QModelIndex [QtXml]	_Znk19QAbstractProxyModel5flags ERK11QModelIndex [QtXml]
_Znk19QAbstractScrollArea10meta ObjectEv [QtGui]	_Znk19QAbstractScrollArea12corne rWidgetEv [QtXml]
_Znk19QAbstractScrollArea15mini mumSizeHintEv [QtGui]	_Znk19QAbstractScrollArea17vertic alScrollBarEv [QtGui]
_Znk19QAbstractScrollArea19horiz ontalScrollBarEv [QtGui]	_Znk19QAbstractScrollArea19maxi mumViewportSizeEv [QtGui]
_Znk19QAbstractScrollArea23vertic alScrollBarPolicyEv [QtGui]	_Znk19QAbstractScrollArea25horiz ontalScrollBarPolicyEv [QtGui]
_Znk19QAbstractScrollArea8sizeHi ntEv [QtGui]	_Znk19QAbstractScrollArea8viewp ortEv [QtGui]
_Znk19QItemSelectionModel10isSel ectedERK11QModelIndex [QtGui]	_Znk19QItemSelectionModel10meta ObjectEv [QtGui]
_Znk19QItemSelectionModel12curr entIndexEv [QtGui]	_Znk19QItemSelectionModel12hasS electionEv [QtXml]
_Znk19QItemSelectionModel12selec tedRowsEi [QtXml]	_Znk19QItemSelectionModel13isRo wSelectedEiRK11QModelIndex [QtGui]
_Znk19QItemSelectionModel15selec tedColumnsEi [QtXml]	_Znk19QItemSelectionModel15selec tedIndexesEv [QtGui]
_Znk19QItemSelectionModel16isCol umnSelectedEiRK11QModelIndex [QtGui]	_Znk19QItemSelectionModel22rowI ntersectsSelectionEiRK11QModelInd ex [QtGui]
_Znk19QItemSelectionModel25colu mnIntersectsSelectionEiRK11QModel Index [QtGui]	_Znk19QItemSelectionModel5mode IEv [QtGui]
_Znk19QItemSelectionModel9selecti onEv [QtGui]	_Znk19QItemSelectionRange10inter sectsERKS_ [QtGui]

_Znk19QItemSelectionRange7index esEv [QtGui]	_Znk19QItemSelectionRange9inters ectERKS_ [QtGui]
_Znk20QAbstractPrintDialog10meta ObjectEv [QtXml]	_Znk20QAbstractPrintDialog10print RangeEv [QtGui]
_Znk20QAbstractPrintDialog14enab ledOptionsEv [QtGui]	_Znk20QAbstractPrintDialog15isOp tionEnabledENS_17PrintDialogOptio nE [QtGui]
_Znk20QAbstractPrintDialog6toPag eEv [QtGui]	_Znk20QAbstractPrintDialog7maxP ageEv [QtGui]
_Znk20QAbstractPrintDialog7minP ageEv [QtGui]	_Znk20QAbstractPrintDialog7printe rEv [QtGui]
_Znk20QAbstractPrintDialog8from PageEv [QtGui]	_Znk21QAbstractItemDelegate10me taObjectEv [QtGui]
_Znk21QAbstractItemDelegate12cre ateEditorEP7QWidgetRK20QStyleOp tionViewItemRK11QModelIndex [QtGui]	_Znk21QAbstractItemDelegate12set ModelDataEP7QWidgetP18QAbstrac tItemModelRK11QModelIndex [QtGui]
_Znk21QAbstractItemDelegate13set EditorDataEP7QWidgetRK11QMode lIndex [QtGui]	_Znk21QAbstractItemDelegate20up dateEditorGeometryEP7QWidgetRK 20QStyleOptionViewItemRK11QMo delIndex [QtGui]
_Znk21QSortFilterProxyModel10filt erRoleEv [QtXml]	_Znk21QSortFilterProxyModel10he aderDataEiN2Qt11OrientationEi [QtGui]
_Znk21QSortFilterProxyModel10me taObjectEv [QtGui]	_Znk21QSortFilterProxyModel11col umnCountERK11QModelIndex [QtGui]
_Znk21QSortFilterProxyModel11has ChildrenERK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel11ma pToSourceERK11QModelIndex [QtGui]
_Znk21QSortFilterProxyModel12can FetchMoreERK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel12filt erRegExpEv [QtGui]
_Znk21QSortFilterProxyModel13ma pFromSourceERK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel15filt erKeyColumnEv [QtGui]
_Znk21QSortFilterProxyModel16filt erAcceptsRowEiRK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel17dy namicSortFilterEv [QtXml]
_Znk21QSortFilterProxyModel19filt erAcceptsColumnEiRK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel19sor tCaseSensitivityEv [QtXml]
_Znk21QSortFilterProxyModel20ma pSelectionToSourceERK14QItemSele ction [QtGui]	_Znk21QSortFilterProxyModel20su pportedDropActionsEv [QtXml]

_Znk21QSortFilterProxyModel21filterCaseSensitivityEv [QtGui]	_Znk21QSortFilterProxyModel22mapSelectionFromSourceERK14QItemSelection [QtGui]
_Znk21QSortFilterProxyModel4dataERK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel4spanERK11QModelIndex [QtGui]
_Znk21QSortFilterProxyModel5budgetERK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel5flagsERK11QModelIndex [QtGui]
_Znk21QSortFilterProxyModel5indexEiRK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel5matchERK11QModelIndexiRK8QVarianti6QFlagsIN2Qt9MatchFlagEE [QtGui]
_Znk21QSortFilterProxyModel6parentERK11QModelIndex [QtGui]	_Znk21QSortFilterProxyModel8lessThanERK11QModelIndexS2_ [QtGui]
_Znk21QSortFilterProxyModel8mimeDataERK5QListI11QModelIndexE [QtGui]	_Znk21QSortFilterProxyModel8rowCountERK11QModelIndex [QtGui]
_Znk21QSortFilterProxyModel8sortRoleEv [QtXml]	_Znk21QSortFilterProxyModel9mimeTypesEv [QtXml]
_Znk24QAbstractPageSetupDialog10metaObjectEv [QtXml]	_Znk26QAbstractGraphicsShapeItem10opaqueAreaEv [QtXml]
_Znk26QAbstractGraphicsShapeItem12isObscuredByEPK13QGraphicsItem [QtXml]	_Znk26QAbstractGraphicsShapeItem3penEv [QtXml]
_Znk26QAbstractGraphicsShapeItem5brushEv [QtXml]	_Znk27QAbstractTextDocumentLayout10metaObjectEv [QtGui]
_Znk27QAbstractTextDocumentLayout11paintDeviceEv [QtGui]	_Znk27QAbstractTextDocumentLayout16handlerForObjectEi [QtGui]
_Znk27QAbstractTextDocumentLayout8anchorAtERK7QPointF [QtGui]	_Znk27QAbstractTextDocumentLayout8documentEv [QtGui]
_Znk9QDirModel10headerDataEiN2Qt11OrientationEi [QtGui]	_Znk9QDirModel10isReadOnlyEv [QtGui]
_Znk9QDirModel10metaObjectEv [QtGui]	_Znk9QDirModel11columnCountERK11QModelIndex [QtGui]
_Znk9QDirModel11hasChildrenERK11QModelIndex [QtGui]	_Znk9QDirModel11nameFiltersEv [QtGui]
_Znk9QDirModel12iconProviderEv [QtGui]	_Znk9QDirModel14lazyChildCountEv [QtGui]
_Znk9QDirModel15resolveSymlinksEv [QtGui]	_Znk9QDirModel20supportedDropActionsEv [QtGui]
_Znk9QDirModel4dataERK11QModelIndex [QtGui]	_Znk9QDirModel5flagsERK11QModelIndex [QtGui]

_Znk9QDirModel5indexERK7QStri ngi [QtGui]	_Znk9QDirModel5indexEiiRK11QM odelIndex [QtGui]
_Znk9QDirModel5isDirERK11QM odelIndex [QtGui]	_Znk9QDirModel6filterEv [QtGui]
_Znk9QDirModel6parentERK11QM odelIndex [QtGui]	_Znk9QDirModel7sortingEv [QtGui]
_Znk9QDirModel8fileIconERK11Q ModelIndex [QtGui]	_Znk9QDirModel8fileInfoERK11Q ModelIndex [QtGui]
_Znk9QDirModel8fileNameERK11 QModelIndex [QtGui]	_Znk9QDirModel8filePathERK11Q ModelIndex [QtGui]
_Znk9QDirModel8mimeDataERK5 QList11QModelIndexE [QtGui]	_Znk9QDirModel8rowCountERK11 QModelIndex [QtGui]
_Znk9QDirModel9mimeTypesEv [QtGui]	_Znk9QTreeView10indexAboveER K11QModelIndex [QtGui]
_Znk9QTreeView10indexBelowERK 11QModelIndex [QtGui]	_Znk9QTreeView10isAnimatedEv [QtXml]
_Znk9QTreeView10isExpandedERK 11QModelIndex [QtGui]	_Znk9QTreeView10metaObjectEv [QtGui]
_Znk9QTreeView10visualRectERK1 1QModelIndex [QtGui]	_Znk9QTreeView11columnWidthEi [QtGui]
_Znk9QTreeView11indentationEv [QtGui]	_Znk9QTreeView11isRowHiddenEi RK11QModelIndex [QtGui]
_Znk9QTreeView12drawBranchesE P8QPainterRK5QRectRK11QModelIn dex [QtGui]	_Znk9QTreeView13isIndexHiddenE RK11QModelIndex [QtGui]
_Znk9QTreeView14isColumnHidde nEi [QtGui]	_Znk9QTreeView14verticalOffsetEv [QtGui]
_Znk9QTreeView15itemsExpandabl eEv [QtGui]	_Znk9QTreeView15rootIsDecorated Ev [QtGui]
_Znk9QTreeView15selectedIndexes Ev [QtGui]	_Znk9QTreeView16horizontalOffset Ev [QtGui]
_Znk9QTreeView16indexRowSizeH intERK11QModelIndex [QtGui]	_Znk9QTreeView16isSortingEnable dEv [QtXml]
_Znk9QTreeView17sizeHintForCol umnEi [QtGui]	_Znk9QTreeView17uniformRowHei ghtsEv [QtGui]
_Znk9QTreeView19allColumnsSho wFocusEv [QtXml]	_Znk9QTreeView22columnViewpor tPositionEi [QtGui]
_Znk9QTreeView24visualRegionFor SelectionERK14QItemSelection [QtGui]	_Znk9QTreeView6headerEv [QtGui]
_Znk9QTreeView7drawRowEP8QP ainterRK20QStyleOptionViewItemR K11QModelIndex [QtGui]	_Znk9QTreeView7indexAtERK6QP oint [QtGui]

_Znk9QTreeView8columnAtEi [QtGui]	_Znk9QTreeView8drawTreeEP8QPainterRK7QRegion [QtXml]
_Zls6QDebugRK19QItemSelectionRange [QtGui]	_ZlsR11QDataStreamRK13QStandardItem [QtXml]
_ZlsR11QDataStreamRK15QTreeViewWidgetItem [QtGui]	_ZlsR11QDataStreamRK16QTableWidgetItem [QtGui]
_ZrsR11QDataStreamRK13QStandardItem [QtXml]	_ZrsR11QDataStreamRK15QTreeViewWidgetItem [QtGui]
_ZrsR11QDataStreamRK16QTableWidgetItem [QtGui]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Models and Views specified in Table 18-258, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-258 libQtGui - Qt4 Models and Views Deprecated Function Interfaces

_ZN21QAbstractItemDelegate10elideTextERK12QFontMetricsiN2Qt13TextElideModeERK7QString [QtGui]	_Znk11QTreeViewWidget5itemsEPK9QMimeData [QtGui]

18.5.10 Qt4 Dialogs

18.5.10.1 Class data for QDialog

The virtual table for the QDialog class is described by Table 18-259

Table 18-259 Primary vtable for QDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDialog
vfunc[0]:	QDialog::metaObject() const
vfunc[1]:	QDialog::qt_metacast(char const*)
vfunc[2]:	QDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDialog::~~QDialog()
vfunc[4]:	QDialog::~~QDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)

vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)

vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QDialog class is described by Table 18-260

Table 18-260 typeinfo for QDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDialog
basetype:	typeinfo for QWidget

18.5.10.2 Class data for QColorDialog

The virtual table for the QColorDialog class is described by Table 18-261

Table 18-261 Primary vtable for QColorDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QColorDialog
vfunc[0]:	QColorDialog::metaObject() const
vfunc[1]:	QColorDialog::qt_metacast(char const*)
vfunc[2]:	QColorDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QColorDialog::~~QColorDialog()
vfunc[4]:	QColorDialog::~QColorDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)

vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)

vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QColorDialog class is described by Table 18-262

Table 18-262 typeinfo for QColorDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QColorDialog
basetype:	typeinfo for QDialog

18.5.10.3 Class data for QFontDialog

The virtual table for the QFontDialog class is described by Table 18-263

Table 18-263 Primary vtable for QFontDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFontDialog
vfunc[0]:	QFontDialog::metaObject() const
vfunc[1]:	QFontDialog::qt_metacast(char const*)
vfunc[2]:	QFontDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFontDialog::~~QFontDialog()
vfunc[4]:	QFontDialog::~~QFontDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QFontDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)

vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)

vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QFontDialog class is described by Table 18-264

Table 18-264 typeinfo for QFontDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFontDialog
basetype:	typeinfo for QDialog

18.5.10.4 Class data for QMessageBox

The virtual table for the QMessageBox class is described by Table 18-265

Table 18-265 Primary vtable for QMessageBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMessageBox
vfunc[0]:	QMessageBox::metaObject() const
vfunc[1]:	QMessageBox::qt_metacast(char const*)
vfunc[2]:	QMessageBox::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMessageBox::~~QMessageBox()
vfunc[4]:	QMessageBox::~QMessageBox()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QMessageBox::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)

vfunc[23]:	QMessageBox::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QMessageBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QMessageBox::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QMessageBox::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QMessageBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const

vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QMessageBox class is described by Table 18-266

Table 18-266 typeinfo for QMessageBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMessageBox
basetype:	typeinfo for QDialog

18.5.10.5 Class data for QProgressDialog

The virtual table for the QProgressDialog class is described by Table 18-267

Table 18-267 Primary vtable for QProgressDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QProgressDialog
vfunc[0]:	QProgressDialog::metaObject() const
vfunc[1]:	QProgressDialog::qt_metacast(char const*)
vfunc[2]:	QProgressDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QProgressDialog::~QProgressDialog()
vfunc[4]:	QProgressDialog::~QProgressDialog()
vfunc[5]:	QWidget::event(QEvent*)

vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QProgressDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QProgressDialog::resizeEvent(QResizeEvent*)

vfunc[32]:	QProgressDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QProgressDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QProgressDialog::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QProgressDialog class is described by Table 18-268

Table 18-268 typeinfo for QProgressDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QProgressDialog
basetype:	typeinfo for QDialog

18.5.10.6 Class data for QErrorMessage

The virtual table for the QErrorMessage class is described by Table 18-269

Table 18-269 Primary vtable for QErrorMessage

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QErrorMessage
vfunc[0]:	QErrorMessage::metaObject() const
vfunc[1]:	QErrorMessage::qt_metacast(char const*)
vfunc[2]:	QErrorMessage::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QErrorMessage::~QErrorMessage()
vfunc[4]:	QErrorMessage::~QErrorMessage()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const

vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)

vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QErrorMessage::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QErrorMessage class is described by Table 18-270

Table 18-270 typeinfo for QErrorMessage

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QErrorMessage
basetype	typeinfo for QDialog

18.5.10.7 Class data for QFileDialog

The virtual table for the QFileDialog class is described by Table 18-271

Table 18-271 Primary vtable for QFileDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFileDialog
vfunc[0]:	QFileDialog::metaObject() const
vfunc[1]:	QFileDialog::qt_metacast(char const*)

vfunc[2]:	QFileDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFileDialog::~~QFileDialog()
vfunc[4]:	QFileDialog::~~QFileDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)

vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QFileDialog::done(int)
vfunc[55]:	QFileDialog::accept()

vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QFileDialog class is described by Table 18-272

Table 18-272 typeinfo for QFileDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFileDialog
basetype:	typeinfo for QDialog

18.5.10.8 Class data for QPrintDialog

The virtual table for the QPrintDialog class is described by Table 18-273

Table 18-273 Primary vtable for QPrintDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPrintDialog
vfunc[0]:	QPrintDialog::metaObject() const
vfunc[1]:	QPrintDialog::qt_metacast(char const*)
vfunc[2]:	QPrintDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QPrintDialog::~~QPrintDialog()
vfunc[4]:	QPrintDialog::~QPrintDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const

vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)

vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()
vfunc[57]:	QPrintDialog::exec()

The Run Time Type Information for the QPrintDialog class is described by Table 18-274

Table 18-274 typeinfo for QPrintDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPrintDialog
basetype:	typeinfo for QAbstractPrintDialog

18.5.10.9 Class data for QPageSetupDialog

The virtual table for the QPageSetupDialog class is described by Table 18-275

Table 18-275 Primary vtable for QPageSetupDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPageSetupDialog
vfunc[0]:	QDialog::metaObject() const
vfunc[1]:	QDialog::qt_metacast(char const*)

vfunc[2]:	QDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QPageSetupDialog::~QPageSetupDialog()
vfunc[4]:	NULL or QPageSetupDialog::~QPageSetupDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)

vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)

vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()
vfunc[57]:	QPageSetupDialog::exec()

The Run Time Type Information for the QPageSetupDialog class is described by Table 18-276

Table 18-276 typeinfo for QPageSetupDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPageSetupDialog
basetype:	typeinfo for QAbstractPageSetupDialog

18.5.10.10 Class data for QInputDialog

The virtual table for the QInputDialog class is described by Table 18-277

Table 18-277 Primary vtable for QInputDialog

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QInputDialog
vfunc[0]:	QInputDialog::metaObject() const
vfunc[1]:	QInputDialog::qt_metacast(char const*)
vfunc[2]:	QInputDialog::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QInputDialog::~~QInputDialog()
vfunc[4]:	QInputDialog::~~QInputDialog()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QDialog::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const

vfunc[13]:	QDialog::setVisible(bool)
vfunc[14]:	QDialog::sizeHint() const
vfunc[15]:	QDialog::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QDialog::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDialog::resizeEvent(QResizeEvent*)
vfunc[32]:	QDialog::closeEvent(QCloseEvent*)
vfunc[33]:	QDialog::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)

vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QDialog::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDialog::done(int)
vfunc[55]:	QDialog::accept()
vfunc[56]:	QDialog::reject()

The Run Time Type Information for the QDialog class is described by Table 18-278

Table 18-278 typeinfo for QDialog

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDialog
basetype:	typeinfo for QDialog

18.5.10.11 Class data for QFileIconProvider

The virtual table for the QFileIconProvider class is described by Table 18-279

Table 18-279 Primary vtable for QFileIconProvider

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QFileIconProvider
vfunc[0]:	QFileIconProvider::~~QFileIconProvider()
vfunc[1]:	QFileIconProvider::~~QFileIconProvider()
vfunc[2]:	QFileIconProvider::icon(QFileIconProvider::IconType) const
vfunc[3]:	QFileIconProvider::icon(QFileInfo const&) const
vfunc[4]:	QFileIconProvider::type(QFileInfo const&) const

The Run Time Type Information for the QFileIconProvider class is described by Table 18-280

Table 18-280 typeinfo for QFileIconProvider

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QFileIconProvider

18.5.10.12 Interfaces for Qt4 Dialogs

An LSB conforming implementation shall provide the generic functions for Qt4 Dialogs specified in Table 18-281, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-281 libQtGui - Qt4 Dialogs Function Interfaces

_ZN11QFileDialog10selectFileERK7QString [QtGui]	_ZN11QFileDialog10setFiltersERK11QStringList [QtGui]
_ZN11QFileDialog10setHistoryERK11QStringList [QtGui]	_ZN11QFileDialog11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN11QFileDialog11qt_metacastEPKc [QtGui]	_ZN11QFileDialog11setFileModeENS_8FileModeE [QtGui]
_ZN11QFileDialog11setReadOnlyEb [QtGui]	_ZN11QFileDialog11setViewModeENS_8ViewModeE [QtGui]
_ZN11QFileDialog12selectFilterERK7QString [QtGui]	_ZN11QFileDialog12setDirectoryERK7QString [QtGui]
_ZN11QFileDialog12setLabelTextENS_11DialogLabelERK7QString [QtGui]	_ZN11QFileDialog13filesSelectedERK11QStringList [QtGui]
_ZN11QFileDialog13setAcceptModeENS_10AcceptModeE [QtGui]	_ZN11QFileDialog14currentChangedERK7QString [QtGui]

_ZN11QFileDialog15getOpenFileNameEP7QWidgetRK7QStringS4_S4_PS2_6QFlagsINS_6OptionEE [QtGui]	_ZN11QFileDialog15getSaveFileNameEP7QWidgetRK7QStringS4_S4_PS2_6QFlagsINS_6OptionEE [QtGui]
_ZN11QFileDialog15setIconProviderEP17QFileIconProvider [QtGui]	_ZN11QFileDialog15setItemDelegateEP21QAbstractItemDelegate [QtGui]
_ZN11QFileDialog16getOpenNamesEP7QWidgetRK7QStringS4_S4_PS2_6QFlagsINS_6OptionEE [QtGui]	_ZN11QFileDialog16setDefaultSuffixERK7QString [QtGui]
_ZN11QFileDialog18setResolveSymLinksEb [QtGui]	_ZN11QFileDialog19setConfirmOverwriteEb [QtGui]
_ZN11QFileDialog20getExistingDirectoryEP7QWidgetRK7QStringS4_6QFlagsINS_6OptionEE [QtGui]	_ZN11QFileDialog4doneEi [QtGui]
_ZN11QFileDialog6acceptEv [QtGui]	_ZN11QFileDialog9setFilterERK7QString [QtGui]
_ZN11QFileDialogC1EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN11QFileDialogC1EP7QWidgetRK7QStringS4_S4_ [QtGui]
_ZN11QFileDialogC1ERK15QFileDialogArgs [QtGui]	_ZN11QFileDialogC2EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN11QFileDialogC2EP7QWidgetRK7QStringS4_S4_ [QtGui]	_ZN11QFileDialogC2ERK15QFileDialogArgs [QtGui]
_ZN11QFileDialogD0Ev [QtGui]	_ZN11QFileDialogD1Ev [QtGui]
_ZN11QFileDialogD2Ev [QtGui]	_ZN11QFontDialog11eventFilterEP7QObjectP6QEvent [QtGui]
_ZN11QFontDialog7getFontEPbP7QWidget [QtGui]	_ZN11QFontDialog7getFontEPbRK5QFontP7QWidget [QtGui]
_ZN11QFontDialog7getFontEPbRK5QFontP7QWidgetRK7QString [QtGui]	_ZN11QFontDialogD0Ev [QtGui]
_ZN11QFontDialogD1Ev [QtGui]	_ZN11QFontDialogD2Ev [QtGui]
_ZN11QMessageBox10closeEventEP11QCloseEvent [QtGui]	_ZN11QMessageBox11changeEventEP6QEvent [QtGui]
_ZN11QMessageBox11informationEP7QWidgetRK7QStringS4_6QFlagsINS_14StandardButtonEES6_ [QtGui]	_ZN11QMessageBox11informationEP7QWidgetRK7QStringS4_S4_S4_S4_ii [QtGui]
_ZN11QMessageBox11informationEP7QWidgetRK7QStringS4_iii [QtGui]	_ZN11QMessageBox11qt_metacallEN11QMetaObject4CalleiPPv [QtGui]
_ZN11QMessageBox11qt_metacastEPKc [QtGui]	_ZN11QMessageBox11resizeEventEP12QResizeEvent [QtGui]
_ZN11QMessageBox12removeButtonEP15QAbstractButton [QtGui]	_ZN11QMessageBox12standardIconENS_4IconE [QtGui]

_ZN11QMessageBox12standardIconENS_4IconEN2Qt8GUIStyleE [QtGui]	_ZN11QMessageBox13keyPressEventEP9QKeyEvent [QtGui]
_ZN11QMessageBox13setButtonTextEiRK7QString [QtGui]	_ZN11QMessageBox13setIconPixmapERK7QPixmap [QtGui]
_ZN11QMessageBox13setTextFormatEN2Qt10TextFormatE [QtGui]	_ZN11QMessageBox14setWindowTitleERK7QString [QtXml]
_ZN11QMessageBox15setDetailedTextERK7QString [QtXml]	_ZN11QMessageBox15setEscapeButtonEP15QAbstractButton [QtXml]
_ZN11QMessageBox16setDefaultButtonEP11QPushButton [QtXml]	_ZN11QMessageBox17setWindowModalityEN2Qt14WindowModalityE [QtXml]
_ZN11QMessageBox18setInformativeTextERK7QString [QtXml]	_ZN11QMessageBox18setStandardButtonsE6QFlagsINS_14StandardButtonEE [QtXml]
ZN11QMessageBox5aboutEP7QWidgetRK7QStringS4 [QtGui]	_ZN11QMessageBox7aboutQtEP7QWidgetRK7QString [QtGui]
_ZN11QMessageBox7setIconENS_4IconE [QtGui]	_ZN11QMessageBox7setTextERK7QString [QtGui]
_ZN11QMessageBox7warningEP7QWidgetRK7QStringS4_6QFlagsINS_14StandardButtonEES6_ [QtXml]	_ZN11QMessageBox7warningEP7QWidgetRK7QStringS4_S4_S4_ii [QtGui]
_ZN11QMessageBox7warningEP7QWidgetRK7QStringS4_iii [QtGui]	_ZN11QMessageBox8criticalEP7QWidgetRK7QStringS4_6QFlagsINS_14StandardButtonEES6_ [QtXml]
_ZN11QMessageBox8criticalEP7QWidgetRK7QStringS4_S4_S4_ii [QtGui]	_ZN11QMessageBox8criticalEP7QWidgetRK7QStringS4_iii [QtGui]
_ZN11QMessageBox8questionEP7QWidgetRK7QStringS4_6QFlagsINS_14StandardButtonEES6_ [QtXml]	_ZN11QMessageBox8questionEP7QWidgetRK7QStringS4_S4_S4_ii [QtGui]
_ZN11QMessageBox8questionEP7QWidgetRK7QStringS4_iii [QtGui]	_ZN11QMessageBox9addButtonENS_14StandardButtonE [QtXml]
_ZN11QMessageBox9addButtonEP15QAbstractButtonNS_10ButtonRoleE [QtXml]	_ZN11QMessageBox9addButtonERK7QStringNS_10ButtonRoleE [QtXml]
_ZN11QMessageBox9showEventEP10QShowEvent [QtGui]	_ZN11QMessageBoxC1ENS_4IconERK7QStringS3_6QFlagsINS_14StandardButtonEEP7QWidgetS4_IN2Qt10WindowTypeEE [QtXml]
_ZN11QMessageBoxC1EP7QWidget [QtGui]	_ZN11QMessageBoxC1EP7QWidgetPKc [QtGui]
_ZN11QMessageBoxC1ERK7QStringS2_NS_4IconEiiiP7QWidget6QFlagsINS2Qt10WindowTypeEE [QtGui]	_ZN11QMessageBoxC1ERK7QStringS2_NS_4IconEiiiP7QWidgetPKcb6QFlagsINS2Qt10WindowTypeEE [QtGui]

_ZN11QMessageBoxC2ENS_4IconERK7QStringS3_6QFlagsINS_14StandardButtonEEP7QWidgetS4_IN2Qt10WindowTypeEE [QtXml]	_ZN11QMessageBoxC2EP7QWidget [QtGui]
_ZN11QMessageBoxC2EP7QWidgetPKc [QtGui]	_ZN11QMessageBoxC2ERK7QStringS2_NS_4IconEiiiP7QWidget6QFlagsINS2Qt10WindowTypeEE [QtGui]
_ZN11QMessageBoxC2ERK7QStringS2_NS_4IconEiiiP7QWidgetPKcb6QFlagsINS2Qt10WindowTypeEE [QtGui]	_ZN11QMessageBoxD0Ev [QtGui]
_ZN11QMessageBoxD1Ev [QtGui]	_ZN11QMessageBoxD2Ev [QtGui]
_ZN12QColorDialog11customColorEi [QtGui]	_ZN12QColorDialog11customCountEv [QtGui]
_ZN12QColorDialog14setCustomColorEij [QtGui]	_ZN12QColorDialog16setStandardColorEij [QtGui]
_ZN12QColorDialog7getRgbaEjPbP7QWidget [QtGui]	_ZN12QColorDialog8getColorERK6QColorP7QWidget [QtGui]
_ZN12QColorDialogD0Ev [QtGui]	_ZN12QColorDialogD1Ev [QtGui]
_ZN12QColorDialogD2Ev [QtGui]	_ZN12QInputDialog10getIntegerEP7QWidgetRK7QStringS4_iiiiPb6QFlagsINS2Qt10WindowTypeEE [QtGui]
_ZN12QInputDialog7getItemEP7QWidgetRK7QStringS4_RK11QStringListibPb6QFlagsINS2Qt10WindowTypeEE [QtGui]	_ZN12QInputDialog7getTextEP7QWidgetRK7QStringS4_N9QLineEdit8EchoModeES4_Pb6QFlagsINS2Qt10WindowTypeEE [QtGui]
_ZN12QInputDialog9getDoubleEP7QWidgetRK7QStringS4_dddPb6QFlagsINS2Qt10WindowTypeEE [QtGui]	_ZN12QInputDialogD0Ev [QtGui]
_ZN12QInputDialogD1Ev [QtGui]	_ZN12QInputDialogD2Ev [QtGui]
_ZN12QPrintDialog10setPrinterEP8QPrinterb [QtGui]	_ZN12QPrintDialog11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN12QPrintDialog11qt_metacastEPKc [QtGui]	_ZN12QPrintDialog4execEv [QtGui]
_ZN12QPrintDialog9addButtonEP11QPushButton [QtGui]	_ZN12QPrintDialogC1EP8QPrinterP7QWidget [QtGui]
_ZN12QPrintDialogC2EP8QPrinterP7QWidget [QtGui]	_ZN12QPrintDialogD0Ev [QtGui]
_ZN12QPrintDialogD1Ev [QtGui]	_ZN12QPrintDialogD2Ev [QtGui]
_ZN13QErrorMessage11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN13QErrorMessage11qt_metacastEPKc [QtGui]
_ZN13QErrorMessage11showMessageERK7QString [QtGui]	_ZN13QErrorMessage4doneEi [QtGui]

_ZN13QErrorMessage9qtHandlerEv [QtGui]	_ZN13QErrorMessageC1EP7QWidge t [QtGui]
_ZN13QErrorMessageC2EP7QWidge t [QtGui]	_ZN13QErrorMessageD0Ev [QtGui]
_ZN13QErrorMessageD1Ev [QtGui]	_ZN13QErrorMessageD2Ev [QtGui]
_ZN15QProgressDialog10closeEvent EP11QCloseEvent [QtGui]	_ZN15QProgressDialog10setMaximu mEi [QtGui]
_ZN15QProgressDialog10setMinimu mEi [QtGui]	_ZN15QProgressDialog11changeEve ntEP6QEvent [QtGui]
_ZN15QProgressDialog11qt_metacal lEN11QMetaObject4CallEiPPv [QtGui]	_ZN15QProgressDialog11qt_metacas tEPKc [QtGui]
_ZN15QProgressDialog11resizeEven tEP12QResizeEvent [QtGui]	_ZN15QProgressDialog12setAutoClo seEb [QtGui]
_ZN15QProgressDialog12setAutoRes etEb [QtGui]	_ZN15QProgressDialog12setLabelTe xtERK7QString [QtGui]
_ZN15QProgressDialog15setCancelB uttonEP11QPushButton [QtGui]	_ZN15QProgressDialog18setMinimu mDurationEi [QtGui]
_ZN15QProgressDialog19setCancelB uttonTextERK7QString [QtGui]	_ZN15QProgressDialog5resetEv [QtGui]
_ZN15QProgressDialog6cancelEv [QtGui]	_ZN15QProgressDialog6setBarEP12 QProgressBar [QtGui]
_ZN15QProgressDialog8canceledEv [QtGui]	_ZN15QProgressDialog8setLabelEP6 QLabel [QtGui]
_ZN15QProgressDialog8setRangeEii [QtGui]	_ZN15QProgressDialog8setValueEi [QtGui]
_ZN15QProgressDialog9forceShowE v [QtGui]	_ZN15QProgressDialog9showEventE P10QShowEvent [QtGui]
_ZN15QProgressDialogC1EP7QWid get6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN15QProgressDialogC1ERK7QStri ngS2_iiP7QWidget6QFlagsIN2Qt10 WindowTypeEE [QtGui]
_ZN15QProgressDialogC2EP7QWid get6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN15QProgressDialogC2ERK7QStri ngS2_iiP7QWidget6QFlagsIN2Qt10 WindowTypeEE [QtGui]
_ZN15QProgressDialogD0Ev [QtGui]	_ZN15QProgressDialogD1Ev [QtGui]
_ZN15QProgressDialogD2Ev [QtGui]	_ZN16QPageSetupDialog11qt_metac allEN11QMetaObject4CallEiPPv [QtXml]
_ZN16QPageSetupDialog11qt_metac astEPKc [QtXml]	_ZN16QPageSetupDialog4execEv [QtGui]
_ZN16QPageSetupDialogC1EP8QPri nterP7QWidget [QtGui]	_ZN16QPageSetupDialogC2EP8QPri nterP7QWidget [QtGui]

_ZN17QFileIconProviderC1Ev [QtGui]	_ZN17QFileIconProviderC2Ev [QtGui]
_ZN17QFileIconProviderD0Ev [QtGui]	_ZN17QFileIconProviderD1Ev [QtGui]
_ZN17QFileIconProviderD2Ev [QtGui]	_ZN7QDialog10closeEventEP11QCloseEvent [QtGui]
_ZN7QDialog10setVisibleEb [QtGui]	_ZN7QDialog11eventFilterEP7QObjectP6QEvent [QtGui]
_ZN7QDialog11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN7QDialog11qt_metacastEPKc [QtGui]
_ZN7QDialog11resizeEventEP12QResizeEvent [QtGui]	_ZN7QDialog12setExtensionEP7QWidget [QtGui]
_ZN7QDialog13keyPressEventEP9QKeyEvent [QtGui]	_ZN7QDialog13showExtensionEb [QtGui]
_ZN7QDialog14adjustPositionEP7QWidget [QtGui]	_ZN7QDialog14setOrientationEN2Qt11OrientationE [QtGui]
_ZN7QDialog16contextMenuEventEP17QContextMenuEvent [QtGui]	_ZN7QDialog18setSizeGripEnabledEb [QtGui]
_ZN7QDialog4doneEi [QtGui]	_ZN7QDialog4execEv [QtGui]
_ZN7QDialog6acceptEv [QtGui]	_ZN7QDialog6rejectEv [QtGui]
_ZN7QDialog8acceptedEv [QtGui]	_ZN7QDialog8finishedEi [QtGui]
_ZN7QDialog8rejectedEv [QtGui]	_ZN7QDialog8setModalEb [QtGui]
_ZN7QDialog9setResultEi [QtGui]	_ZN7QDialog9showEventEP10QShowEvent [QtGui]
_ZN7QDialogC1EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN7QDialogC1EP7QWidgetPKcb6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN7QDialogC2EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN7QDialogC2EP7QWidgetPKcb6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN7QDialogD0Ev [QtGui]	_ZN7QDialogD1Ev [QtGui]
_ZN7QDialogD2Ev [QtGui]	_ZNK11QFileDialog10acceptModeEv [QtGui]
_ZNK11QFileDialog10isReadOnlyEv [QtGui]	_ZNK11QFileDialog10metaObjectEv [QtGui]
_ZNK11QFileDialog12iconProviderEv [QtGui]	_ZNK11QFileDialog12itemDelegateEv [QtGui]
_ZNK11QFileDialog12selectedFileEv [QtGui]	_ZNK11QFileDialog13defaultSuffixEv [QtGui]
_ZNK11QFileDialog13selectedFilesEv [QtGui]	_ZNK11QFileDialog14selectedFilterEv [QtGui]

_ZNK11QFileDialog15resolveSymlinksEv [QtGui]	_ZNK11QFileDialog16confirmOverwriteEv [QtGui]
_ZNK11QFileDialog7filtersEv [QtGui]	_ZNK11QFileDialog7historyEv [QtGui]
_ZNK11QFileDialog8fileModeEv [QtGui]	_ZNK11QFileDialog8viewModeEv [QtGui]
_ZNK11QFileDialog9directoryEv [QtGui]	_ZNK11QFileDialog9labelTextENS_11DialogLabelE [QtGui]
_ZNK11QMessageBox10buttonTextEi [QtGui]	_ZNK11QMessageBox10iconPixmapEv [QtGui]
_ZNK11QMessageBox10metaObjectEv [QtGui]	_ZNK11QMessageBox10textFormatEv [QtGui]
_ZNK11QMessageBox12detailedTextEv [QtXml]	_ZNK11QMessageBox12escapeButtonEv [QtXml]
_ZNK11QMessageBox13clickedButtonEv [QtXml]	_ZNK11QMessageBox13defaultButtonEv [QtXml]
_ZNK11QMessageBox14standardButtonEP15QAbstractButton [QtXml]	_ZNK11QMessageBox15informativeTextEv [QtXml]
_ZNK11QMessageBox15standardButtonsEv [QtXml]	_ZNK11QMessageBox4iconEv [QtGui]
_ZNK11QMessageBox4textEv [QtGui]	_ZNK11QMessageBox6buttonENS_14StandardButtonE [QtXml]
_ZNK11QMessageBox8sizeHintEv [QtGui]	_ZNK12QPrintDialog10metaObjectEv [QtGui]
_ZNK12QPrintDialog7printerEv [QtGui]	_ZNK13QErrorMessage10metaObjectEv [QtGui]
_ZNK15QProgressDialog10metaObjectEv [QtGui]	_ZNK15QProgressDialog11wasCanceledEv [QtGui]
_ZNK15QProgressDialog15minimumDurationEv [QtGui]	_ZNK15QProgressDialog5valueEv [QtGui]
_ZNK15QProgressDialog7maximumEv [QtGui]	_ZNK15QProgressDialog7minimumEv [QtGui]
_ZNK15QProgressDialog8sizeHintEv [QtGui]	_ZNK15QProgressDialog9autoCloseEv [QtGui]
_ZNK15QProgressDialog9autoResetEv [QtGui]	_ZNK15QProgressDialog9labelTextEv [QtGui]
_ZNK16QPageSetupDialog10metaObjectEv [QtXml]	_ZNK17QFileIconProvider4iconENS_8IconTypeE [QtGui]
_ZNK17QFileIconProvider4iconERK9QFileInfo [QtGui]	_ZNK17QFileIconProvider4typeERK9QFileInfo [QtGui]
_ZNK7QDialog10metaObjectEv [QtGui]	_ZNK7QDialog11orientationEv [QtGui]

_ZNK7QDialog15minimumSizeHintEv [QtGui]	_ZNK7QDialog17isSizeGripEnabledEv [QtGui]
_ZNK7QDialog6resultEv [QtGui]	_ZNK7QDialog8sizeHintEv [QtGui]
_ZNK7QDialog9extensionEv [QtGui]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Dialogs specified in Table 18-282, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-282 libQtGui - Qt4 Dialogs Deprecated Function Interfaces

_ZN11QMessageBox13setButtonTextEiRK7QString [QtGui]	_ZN7QDialog12setExtensionEP7QWi dget [QtGui]
_ZN7QDialog13showExtensionEb [QtGui]	_ZN7QDialog14setOrientationEN2Qt 11OrientationE [QtGui]
_ZNK11QMessageBox10buttonTextEi [QtGui]	_ZNK7QDialog11orientationEv [QtGui]
_ZNK7QDialog9extensionEv [QtGui]	

18.5.11 Qt4 Date and Time

18.5.11.1 Class data for QDateTimeEdit

The virtual table for the QDateTimeEdit class is described by Table 18-283

Table 18-283 Primary vtable for QDateTimeEdit

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDateTimeEdit
vfunc[0]:	QDateTimeEdit::metaObject() const
vfunc[1]:	QDateTimeEdit::qt_metacast(char const*)
vfunc[2]:	QDateTimeEdit::qt_metacall(QMeta Object::Call, int, void**)
vfunc[3]:	NULL or QDateTimeEdit::~~QDateTimeEdit()
vfunc[4]:	NULL or QDateTimeEdit::~~QDateTimeEdit()
vfunc[5]:	QDateTimeEdit::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSpinBox::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QDateTimeEdit::sizeHint() const
vfunc[15]:	QAbstractSpinBox::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractSpinBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractSpinBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractSpinBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QDateTimeEdit::wheelEvent(QWheelEvent*)
vfunc[23]:	QDateTimeEdit::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractSpinBox::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QDateTimeEdit::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractSpinBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractSpinBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractSpinBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QAbstractSpinBox::closeEvent(QCloseEvent*)

vfunc[33]:	QAbstractSpinBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QAbstractSpinBox::showEvent(QShowEvent*)
vfunc[41]:	QAbstractSpinBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSpinBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QDateTimeEdit::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDateTimeEdit::validate(QString&, int&) const
vfunc[55]:	QDateTimeEdit::fixup(QString&) const
vfunc[56]:	QDateTimeEdit::stepBy(int)

vfunc[57]:	QDateTimeEdit::clear()
vfunc[58]:	QDateTimeEdit::stepEnabled() const
vfunc[59]:	QDateTimeEdit::dateTimeFromText(QString const&) const
vfunc[60]:	QDateTimeEdit::textFromDateTime(QDateTime const&) const

The Run Time Type Information for the QDateTimeEdit class is described by Table 18-284

Table 18-284 typeinfo for QDateTimeEdit

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDateTimeEdit
basetype:	typeinfo for QAbstractSpinBox

18.5.11.2 Class data for QTimeEdit

The virtual table for the QTimeEdit class is described by Table 18-285

Table 18-285 Primary vtable for QTimeEdit

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTimeEdit
vfunc[0]:	QTimeEdit::metaObject() const
vfunc[1]:	QTimeEdit::qt_metacast(char const*)
vfunc[2]:	QTimeEdit::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QTimeEdit::~~QTimeEdit()
vfunc[4]:	NULL or QTimeEdit::~~QTimeEdit()
vfunc[5]:	QDateTimeEdit::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSpinBox::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const

vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QDateTimeEdit::sizeHint() const
vfunc[15]:	QAbstractSpinBox::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractSpinBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractSpinBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractSpinBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QDateTimeEdit::wheelEvent(QWheelEvent*)
vfunc[23]:	QDateTimeEdit::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractSpinBox::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QDateTimeEdit::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractSpinBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractSpinBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractSpinBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QAbstractSpinBox::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractSpinBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)

vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QAbstractSpinBox::showEvent(QShowEvent*)
vfunc[41]:	QAbstractSpinBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSpinBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QDateTimeEdit::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDateTimeEdit::validate(QString&, int&) const
vfunc[55]:	QDateTimeEdit::fixup(QString&) const
vfunc[56]:	QDateTimeEdit::stepBy(int)
vfunc[57]:	QDateTimeEdit::clear()
vfunc[58]:	QDateTimeEdit::stepEnabled() const
vfunc[59]:	QDateTimeEdit::dateTimeFromText(QString const&) const
vfunc[60]:	QDateTimeEdit::textFromDateTime(QDateTime const&) const

The Run Time Type Information for the QTimeEdit class is described by Table 18-286

Table 18-286 typeinfo for QTimeEdit

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTimeEdit
basetype:	typeinfo for QDateTimeEdit

18.5.11.3 Class data for QDateTimeEdit

The virtual table for the QDateTimeEdit class is described by Table 18-287

Table 18-287 Primary vtable for QDateTimeEdit

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDateTimeEdit
vfunc[0]:	QDateTimeEdit::metaObject() const
vfunc[1]:	QDateTimeEdit::qt_metacast(char const*)
vfunc[2]:	QDateTimeEdit::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QDateTimeEdit::~~QDateTimeEdit()
vfunc[4]:	NULL or QDateTimeEdit::~~QDateTimeEdit()
vfunc[5]:	QDateTimeEdit::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSpinBox::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QDateTimeEdit::sizeHint() const
vfunc[15]:	QAbstractSpinBox::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const

vfunc[18]:	QAbstractSpinBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractSpinBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractSpinBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QDateTimeEdit::wheelEvent(QWheelEvent*)
vfunc[23]:	QDateTimeEdit::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractSpinBox::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QDateTimeEdit::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractSpinBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractSpinBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractSpinBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QAbstractSpinBox::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractSpinBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)

vfunc[40]:	QAbstractSpinBox::showEvent(QShowEvent*)
vfunc[41]:	QAbstractSpinBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSpinBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QDateTimeEdit::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDateTimeEdit::validate(QString&, int&) const
vfunc[55]:	QDateTimeEdit::fixup(QString&) const
vfunc[56]:	QDateTimeEdit::stepBy(int)
vfunc[57]:	QDateTimeEdit::clear()
vfunc[58]:	QDateTimeEdit::stepEnabled() const
vfunc[59]:	QDateTimeEdit::dateTimeFromText(QString const&) const
vfunc[60]:	QDateTimeEdit::textFromDateTime(QDateTime const&) const

The Run Time Type Information for the QDateTimeEdit class is described by Table 18-288

Table 18-288 typeinfo for QDateTimeEdit

Base Vtable	vtable for __cxxabiv1::__si_class_type_info

Name	typeinfo name for QDateEdit
basetype:	typeinfo for QDateTimeEdit

18.5.11.4 Interfaces for Qt4 Date and Time

An LSB conforming implementation shall provide the generic functions for Qt4 Date and Time specified in Table 18-289, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-289 libQtGui - Qt4 Date and Time Function Interfaces

_ZN13QDateTimeEdit10paintEventEP11QPaintEvent [QtXml]	_ZN13QDateTimeEdit10wheelEventEP11QWheelEvent [QtGui]
_ZN13QDateTimeEdit11dateChangedERK5QDate [QtGui]	_ZN13QDateTimeEdit11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN13QDateTimeEdit11qt_metacastEPKc [QtGui]	_ZN13QDateTimeEdit11setDateTimeERK9QDateTime [QtGui]
_ZN13QDateTimeEdit11timeChangedERK5QTime [QtGui]	_ZN13QDateTimeEdit12focusInEventEP11QFocusEvent [QtGui]
ZN13QDateTimeEdit12setDateRangeERK5QDateS2 [QtGui]	_ZN13QDateTimeEdit12setTimeRangeERK5QTimeS2_ [QtGui]
_ZN13QDateTimeEdit13keyPressEventEP9QKeyEvent [QtGui]	_ZN13QDateTimeEdit14setMaximumDateERK5QDate [QtGui]
_ZN13QDateTimeEdit14setMaximumTimeERK5QTime [QtGui]	_ZN13QDateTimeEdit14setMinimumDateERK5QDate [QtGui]
_ZN13QDateTimeEdit14setMinimumTimeERK5QTime [QtGui]	_ZN13QDateTimeEdit15dateTimeChangedERK9QDateTime [QtGui]
_ZN13QDateTimeEdit15mousePressEventEP11QMouseEvent [QtXml]	_ZN13QDateTimeEdit16clearMaximumDateEv [QtGui]
_ZN13QDateTimeEdit16clearMaximumTimeEv [QtGui]	_ZN13QDateTimeEdit16clearMinimumDateEv [QtGui]
_ZN13QDateTimeEdit16clearMinimumTimeEv [QtGui]	_ZN13QDateTimeEdit16setCalendarPopupEb [QtXml]
_ZN13QDateTimeEdit16setDisplayFormatERK7QString [QtGui]	_ZN13QDateTimeEdit17setCurrentSectionENS_7SectionE [QtGui]
_ZN13QDateTimeEdit18focusNextPrevChildEb [QtGui]	_ZN13QDateTimeEdit18setSelectedSectionENS_7SectionE [QtXml]
_ZN13QDateTimeEdit5clearEv [QtGui]	_ZN13QDateTimeEdit5eventEP6QEvent [QtGui]
_ZN13QDateTimeEdit6stepByEi [QtGui]	_ZN13QDateTimeEdit7setDateERK5QDate [QtGui]
_ZN13QDateTimeEdit7setTimeERK5QTime [QtGui]	_ZN13QDateTimeEditC1EP7QWidget [QtGui]

_ZN13QDateTimeEditC1ERK5QDateP7QWidget [QtGui]	_ZN13QDateTimeEditC1ERK5QTimeP7QWidget [QtGui]
_ZN13QDateTimeEditC1ERK9QDateP7QWidget [QtGui]	_ZN13QDateTimeEditC2EP7QWidget [QtGui]
_ZN13QDateTimeEditC2ERK5QDateP7QWidget [QtGui]	_ZN13QDateTimeEditC2ERK5QTimeP7QWidget [QtGui]
_ZN13QDateTimeEditC2ERK9QDateP7QWidget [QtGui]	_ZN9QDateEdit11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QDateEdit11qt_metacastEPKc [QtGui]	_ZN9QDateEditC1EP7QWidget [QtGui]
_ZN9QDateEditC1ERK5QDateP7QWidget [QtGui]	_ZN9QDateEditC2EP7QWidget [QtGui]
_ZN9QDateEditC2ERK5QDateP7QWidget [QtGui]	_ZN9QTimeEdit11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QTimeEdit11qt_metacastEPKc [QtGui]	_ZN9QTimeEditC1EP7QWidget [QtGui]
_ZN9QTimeEditC1ERK5QTimeP7QWidget [QtGui]	_ZN9QTimeEditC2EP7QWidget [QtGui]
_ZN9QTimeEditC2ERK5QTimeP7QWidget [QtGui]	_ZNK13QDateTimeEdit10metaObjectEv [QtGui]
_ZNK13QDateTimeEdit11maximumDateEv [QtGui]	_ZNK13QDateTimeEdit11maximumTimeEv [QtGui]
_ZNK13QDateTimeEdit11minimumDateEv [QtGui]	_ZNK13QDateTimeEdit11minimumTimeEv [QtGui]
_ZNK13QDateTimeEdit11sectionTextENS_7SectionE [QtGui]	_ZNK13QDateTimeEdit11stepEnabledEv [QtGui]
_ZNK13QDateTimeEdit13calendarPopupEv [QtGui]	_ZNK13QDateTimeEdit13displayFormatEv [QtGui]
_ZNK13QDateTimeEdit14currentSectionEv [QtGui]	_ZNK13QDateTimeEdit16dateTimeFromTextERK7QString [QtGui]
_ZNK13QDateTimeEdit16textFromDateTimeERK9QDateTime [QtGui]	_ZNK13QDateTimeEdit17displayedSectionsEv [QtGui]
_ZNK13QDateTimeEdit4dateEv [QtGui]	_ZNK13QDateTimeEdit4timeEv [QtGui]
_ZNK13QDateTimeEdit5fixupER7QString [QtGui]	_ZNK13QDateTimeEdit8dateTimeEv [QtGui]
_ZNK13QDateTimeEdit8sizeHintEv [QtGui]	_ZNK13QDateTimeEdit8validateER7QStringRi [QtGui]
_ZNK9QDateEdit10metaObjectEv [QtGui]	_ZNK9QTimeEdit10metaObjectEv [QtGui]

18.5.12 Qt4 Miscellaneous

18.5.12.1 Class data for QRubberBand

The virtual table for the QRubberBand class is described by Table 18-290

Table 18-290 Primary vtable for QRubberBand

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QRubberBand
vfunc[0]:	QRubberBand::metaObject() const
vfunc[1]:	QRubberBand::qt_metacast(char const*)
vfunc[2]:	QRubberBand::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QRubberBand::~~QRubberBand()
vfunc[4]:	QRubberBand::~QRubberBand()
vfunc[5]:	QRubberBand::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)

vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QRubberBand::paintEvent(QPaintEvent*)
vfunc[30]:	QRubberBand::moveEvent(QMoveEvent*)
vfunc[31]:	QRubberBand::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QRubberBand::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QRubberBand::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const

vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QRubberBand class is described by Table 18-291

Table 18-291 typeinfo for QRubberBand

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QRubberBand
basetype:	typeinfo for QWidget

18.5.12.2 Class data for QShortcut

The virtual table for the QShortcut class is described by Table 18-292

Table 18-292 Primary vtable for QShortcut

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QShortcut
vfunc[0]:	QShortcut::metaObject() const
vfunc[1]:	QShortcut::qt_metacast(char const*)
vfunc[2]:	QShortcut::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QShortcut::~~QShortcut()
vfunc[4]:	QShortcut::~~QShortcut()
vfunc[5]:	QShortcut::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QShortcut class is described by Table 18-293

Table 18-293 typeinfo for QShortcut

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QShortcut
basetype:	typeinfo for QObject

18.5.12.3 Class data for QSplashScreen

The virtual table for the QSplashScreen class is described by Table 18-294

Table 18-294 Primary vtable for QSplashScreen

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSplashScreen
vfunc[0]:	QSplashScreen::metaObject() const
vfunc[1]:	QSplashScreen::qt_metacast(char const*)
vfunc[2]:	QSplashScreen::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSplashScreen::~~QSplashScreen()
vfunc[4]:	QSplashScreen::~~QSplashScreen()
vfunc[5]:	QSplashScreen::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)

vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QSplashScreen::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)

vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QSplashScreen::drawContents(QPainter*)

The Run Time Type Information for the QSplashScreen class is described by Table 18-295

Table 18-295 typeinfo for QSplashScreen

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSplashScreen
basetype:	typeinfo for QWidget

18.5.12.4 Class data for QUndoCommand

The virtual table for the QUndoCommand class is described by Table 18-296

Table 18-296 Primary vtable for QUndoCommand

Base Offset	0
Virtual Base Offset	0

RTTI	typeinfo for QUndoCommand
vfunc[0]:	QUndoCommand::~~QUndoCommand()
vfunc[1]:	QUndoCommand::~~QUndoCommand()
vfunc[2]:	QUndoCommand::undo()
vfunc[3]:	QUndoCommand::redo()
vfunc[4]:	QUndoCommand::id() const
vfunc[5]:	QUndoCommand::mergeWith(QUndoCommand const*)

18.5.12.5 Class data for QUndoStack

The virtual table for the QUndoStack class is described by Table 18-297

Table 18-297 Primary vtable for QUndoStack

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QUndoStack
vfunc[0]:	QUndoStack::metaObject() const
vfunc[1]:	QUndoStack::qt_metacast(char const*)
vfunc[2]:	QUndoStack::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QUndoStack::~~QUndoStack()
vfunc[4]:	QUndoStack::~~QUndoStack()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

18.5.12.6 Interfaces for Qt4 Miscellaneous

An LSB conforming implementation shall provide the generic functions for Qt4 Miscellaneous specified in Table 18-298, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-298 libQtGui - Qt4 Miscellaneous Function Interfaces

_ZN10QUndoGroup11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN10QUndoGroup11qt_metacastEPKc [QtXml]
_ZN10QUndoGroup11removeStackEP10QUndoStack [QtXml]	_ZN10QUndoGroup12cleanChangedEb [QtXml]
_ZN10QUndoGroup12indexChangedEi [QtXml]	_ZN10QUndoGroup14canRedoChangedEb [QtXml]
_ZN10QUndoGroup14canUndoChangedEb [QtXml]	_ZN10QUndoGroup14setActiveStackEP10QUndoStack [QtXml]
_ZN10QUndoGroup15redoTextChangedERK7QString [QtXml]	_ZN10QUndoGroup15undoTextChangedERK7QString [QtXml]
_ZN10QUndoGroup18activeStackChangedEP10QUndoStack [QtXml]	_ZN10QUndoGroup4redoEv [QtXml]
_ZN10QUndoGroup4undoEv [QtXml]	_ZN10QUndoGroup8addStackEP10QUndoStack [QtXml]
_ZN10QUndoGroupC1EP7QObject [QtXml]	_ZN10QUndoGroupC2EP7QObject [QtXml]
_ZN10QUndoStack10beginMacroERK7QString [QtXml]	_ZN10QUndoStack11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN10QUndoStack11qt_metacastEPKc [QtXml]	_ZN10QUndoStack12cleanChangedEb [QtXml]
_ZN10QUndoStack12indexChangedEi [QtXml]	_ZN10QUndoStack14canRedoChangedEb [QtXml]
_ZN10QUndoStack14canUndoChangedEb [QtXml]	_ZN10QUndoStack15redoTextChangedERK7QString [QtXml]
_ZN10QUndoStack15undoTextChangedERK7QString [QtXml]	_ZN10QUndoStack4pushEP12UndoCommand [QtXml]
_ZN10QUndoStack4redoEv [QtXml]	_ZN10QUndoStack4undoEv [QtXml]
_ZN10QUndoStack5clearEv [QtXml]	_ZN10QUndoStack8endMacroEv [QtXml]
_ZN10QUndoStack8setCleanEv [QtXml]	_ZN10QUndoStack8setIndexEi [QtXml]
_ZN10QUndoStack9setActiveEb [QtXml]	_ZN10QUndoStackC1EP7QObject [QtXml]
_ZN10QUndoStackC2EP7QObject [QtXml]	_ZN10QUndoStackD0Ev [QtXml]
_ZN10QUndoStackD1Ev [QtXml]	_ZN10QUndoStackD2Ev [QtXml]
_ZN11QRubberBand10paintEventEP11QPaintEvent [QtGui]	_ZN11QRubberBand11changeEventEP6QEvent [QtGui]
_ZN11QRubberBand11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QRubberBand11qt_metacastEPKc [QtGui]

_ZN11QRubberBand11resizeEventEP12QResizeEvent [QtGui]	_ZN11QRubberBand11setGeometryERK5QRect [QtGui]
_ZN11QRubberBand5eventEP6QEvent [QtGui]	_ZN11QRubberBand9moveEventEP10QMoveEvent [QtGui]
_ZN11QRubberBand9showEventEP10QShowEvent [QtGui]	_ZN11QRubberBandC1ENS_5ShapeEP7QWidget [QtGui]
_ZN11QRubberBandC2ENS_5ShapeEP7QWidget [QtGui]	_ZN11QRubberBandD0Ev [QtGui]
_ZN11QRubberBandD1Ev [QtGui]	_ZN11QRubberBandD2Ev [QtGui]
_ZN12QKeySequence10fromStringERK7QStringNS_14SequenceFormatE [QtGui]	_ZN12QKeySequence11keyBindingsENS_11StandardKeyE [QtXml]
_ZN12QKeySequence8mnemonicERK7QString [QtGui]	_ZN12QKeySequenceC1ENS_11StandardKeyE [QtXml]
_ZN12QKeySequenceC1ERK7QString [QtGui]	_ZN12QKeySequenceC1ERKS_ [QtGui]
_ZN12QKeySequenceC1Eiiii [QtGui]	_ZN12QKeySequenceC1Ev [QtGui]
_ZN12QKeySequenceC2ENS_11StandardKeyE [QtXml]	_ZN12QKeySequenceC2ERK7QString [QtGui]
ZN12QKeySequenceC2ERKS [QtGui]	_ZN12QKeySequenceC2Eiiii [QtGui]
_ZN12QKeySequenceC2Ev [QtGui]	_ZN12QKeySequenceD1Ev [QtGui]
_ZN12QKeySequenceD2Ev [QtGui]	_ZN12QKeySequenceeaSERKS_ [QtGui]
_ZN12QUndoCommand4redoEv [QtXml]	_ZN12QUndoCommand4undoEv [QtXml]
_ZN12QUndoCommand7setTextERK7QString [QtXml]	_ZN12QUndoCommand9mergeWithEPKS_ [QtXml]
ZN12QUndoCommandC1EPS [QtXml]	_ZN12QUndoCommandC1ERK7QStringPS_ [QtXml]
ZN12QUndoCommandC2EPS [QtXml]	_ZN12QUndoCommandC2ERK7QStringPS_ [QtXml]
_ZN12QUndoCommandD0Ev [QtXml]	_ZN12QUndoCommandD1Ev [QtXml]
_ZN12QUndoCommandD2Ev [QtXml]	_ZN13QSplashScreen11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN13QSplashScreen11qt_metacastEPKc [QtGui]	_ZN13QSplashScreen11showMessageERK7QStringiRK6QColor [QtGui]
_ZN13QSplashScreen12clearMessageEv [QtGui]	_ZN13QSplashScreen12drawContentsEP8QPainter [QtGui]
_ZN13QSplashScreen14messageChangedERK7QString [QtGui]	_ZN13QSplashScreen15mousePressEventEP11QMouseEvent [QtGui]

_ZN13QSplashScreen5eventEP6QEvent [QtGui]	_ZN13QSplashScreen6finishEP7QWidget [QtGui]
_ZN13QSplashScreen7repaintEv [QtGui]	_ZN13QSplashScreen9setPixmapERK7QPixmap [QtGui]
_ZN13QSplashScreenC1EP7QWidgetRK7QPixmap6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN13QSplashScreenC1ERK7QPixmap6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN13QSplashScreenC2EP7QWidgetRK7QPixmap6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN13QSplashScreenC2ERK7QPixmap6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN13QSplashScreenD0Ev [QtGui]	_ZN13QSplashScreenD1Ev [QtGui]
_ZN13QSplashScreenD2Ev [QtGui]	_ZN7QCursor3posEv [QtGui]
_ZN7QCursor6setPosEii [QtGui]	_ZN7QCursor8setShapeEN2Qt11CursorShapeE [QtGui]
_ZN7QCursor9x11ScreenEv [QtGui]	_ZN7QCursorC1EN2Qt11CursorShapeE [QtGui]
_ZN7QCursorC1ERK7QBitmapS2_ii [QtGui]	_ZN7QCursorC1ERK7QPixmapii [QtGui]
ZN7QCursorC1ERKS [QtGui]	_ZN7QCursorC1Em [QtGui]
_ZN7QCursorC1Ev [QtGui]	_ZN7QCursorC2EN2Qt11CursorShapeE [QtGui]
_ZN7QCursorC2ERK7QBitmapS2_ii [QtGui]	_ZN7QCursorC2ERK7QPixmapii [QtGui]
ZN7QCursorC2ERKS [QtGui]	_ZN7QCursorC2Em [QtGui]
_ZN7QCursorC2Ev [QtGui]	_ZN7QCursorD1Ev [QtGui]
_ZN7QCursorD2Ev [QtGui]	_ZN7QCursoraSERKS_ [QtGui]
_ZN9QShortcut10setContextEN2Qt15ShortcutContextE [QtGui]	_ZN9QShortcut10setEnabledEb [QtGui]
_ZN9QShortcut11qt_metacallEN11QMetaObject4CalleiPPv [QtGui]	_ZN9QShortcut11qt_metacastEPKc [QtGui]
_ZN9QShortcut12setWhatsThisERK7QString [QtGui]	_ZN9QShortcut13setAutoRepeatEb [QtGui]
_ZN9QShortcut20activatedAmbiguouslyEv [QtGui]	_ZN9QShortcut5eventEP6QEvent [QtGui]
_ZN9QShortcut6setKeyERK12QKeySequence [QtGui]	_ZN9QShortcut7contextEv [QtGui]
_ZN9QShortcut9activatedEv [QtGui]	_ZN9QShortcutC1EP7QWidget [QtGui]
_ZN9QShortcutC1ERK12QKeySequenceP7QWidgetPKcS6_N2Qt15ShortcutContextE [QtGui]	_ZN9QShortcutC2EP7QWidget [QtGui]

_ZN9QShortcutC2ERK12QKeySequenceP7QWidgetPKcS6_N2Qt15ShortcutContextE [QtGui]	_ZN9QShortcutD0Ev [QtGui]
_ZN9QShortcutD1Ev [QtGui]	_ZN9QShortcutD2Ev [QtGui]
_ZN9QUndoView11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN9QUndoView11qt_metacastEPKc [QtXml]
_ZN9QUndoView12setCleanIconERK5QIcon [QtXml]	_ZN9QUndoView13setEmptyLabelERK7QString [QtXml]
_ZN9QUndoView8setGroupEP10QUndoGroup [QtXml]	_ZN9QUndoView8setStackEP10QUndoStack [QtXml]
_ZN9QUndoViewC1EP10QUndoGroupP7QWidget [QtXml]	_ZN9QUndoViewC1EP10QUndoStackP7QWidget [QtXml]
_ZN9QUndoViewC1EP7QWidget [QtXml]	_ZN9QUndoViewC2EP10QUndoGroupP7QWidget [QtXml]
_ZN9QUndoViewC2EP10QUndoStackP7QWidget [QtXml]	_ZN9QUndoViewC2EP7QWidget [QtXml]
_ZN9QUndoViewD0Ev [QtXml]	_ZN9QUndoViewD1Ev [QtXml]
_ZN9QUndoViewD2Ev [QtXml]	_ZNK10QUndoGroup10metaObjectEv [QtXml]
_ZNK10QUndoGroup11activeStackEv [QtXml]	_ZNK10QUndoGroup16createRedoActionEP7QObjectRK7QString [QtXml]
_ZNK10QUndoGroup16createUndoActionEP7QObjectRK7QString [QtXml]	_ZNK10QUndoGroup6stacksEv [QtXml]
_ZNK10QUndoGroup7canRedoEv [QtXml]	_ZNK10QUndoGroup7canUndoEv [QtXml]
_ZNK10QUndoGroup7isCleanEv [QtXml]	_ZNK10QUndoGroup8redoTextEv [QtXml]
_ZNK10QUndoGroup8undoTextEv [QtXml]	_ZNK10QUndoStack10cleanIndexEv [QtXml]
_ZNK10QUndoStack10metaObjectEv [QtXml]	_ZNK10QUndoStack16createRedoActionEP7QObjectRK7QString [QtXml]
_ZNK10QUndoStack16createUndoActionEP7QObjectRK7QString [QtXml]	_ZNK10QUndoStack4textEi [QtXml]
_ZNK10QUndoStack5countEv [QtXml]	_ZNK10QUndoStack5indexEv [QtXml]
_ZNK10QUndoStack7canRedoEv [QtXml]	_ZNK10QUndoStack7canUndoEv [QtXml]
_ZNK10QUndoStack7isCleanEv [QtXml]	_ZNK10QUndoStack8isActiveEv [QtXml]

_Znk10QUndoStack8redoTextEv [QtXml]	_Znk10QUndoStack8undoTextEv [QtXml]
_Znk11QRubberBand10metaObjectEv [QtGui]	_Znk11QRubberBand5shapeEv [QtGui]
_Znk12QKeySequence10isDetachedEv [QtGui]	_Znk12QKeySequence5countEv [QtGui]
_Znk12QKeySequence7isEmptyEv [QtGui]	_Znk12QKeySequence7matchesERKS_ [QtGui]
_Znk12QKeySequence8toStringENS_14SequenceFormatE [QtGui]	_Znk12QKeySequencecv7QStringEv [QtGui]
_Znk12QKeySequencecv8QVariantEv [QtGui]	_Znk12QKeySequencecvEv [QtGui]
Znk12QKeySequenceeqERKS [QtGui]	_Znk12QKeySequenceixEj [QtGui]
Znk12QKeySequencecvtERKS [QtGui]	_Znk12QUndoCommand2idEv [QtXml]
_Znk12QUndoCommand4textEv [QtXml]	_Znk13QSplashScreen10metaObjectEv [QtGui]
_Znk13QSplashScreen6pixmapEv [QtGui]	_Znk7QCursor4maskEv [QtGui]
_Znk7QCursor5shapeEv [QtGui]	_Znk7QCursor6bitmapEv [QtGui]
_Znk7QCursor6handleEv [QtGui]	_Znk7QCursor6pixmapEv [QtGui]
_Znk7QCursor7hotSpotEv [QtGui]	_Znk7QCursorcv8QVariantEv [QtGui]
_Znk9QShortcut10autoRepeatEv [QtXml]	_Znk9QShortcut10metaObjectEv [QtGui]
_Znk9QShortcut2idEv [QtGui]	_Znk9QShortcut3keyEv [QtGui]
_Znk9QShortcut9isEnabledEv [QtGui]	_Znk9QShortcut9whatsThisEv [QtGui]
_Znk9QUndoView10emptyLabelEv [QtXml]	_Znk9QUndoView10metaObjectEv [QtXml]
_Znk9QUndoView5groupEv [QtXml]	_Znk9QUndoView5stackEv [QtXml]
_Znk9QUndoView9cleanIconEv [QtXml]	_Zls6QDebugRK12QKeySequence [QtGui]
_ZlsR11QDataStreamRK12QKeySequence [QtGui]	_ZlsR11QDataStreamRK7QCursor [QtGui]
_ZrsR11QDataStreamR12QKeySequence [QtGui]	_ZrsR11QDataStreamR7QCursor [QtGui]

18.5.13 Qt4 Paint Device

18.5.13.1 Class data for QPaintDevice

The virtual table for the QPaintDevice class is described by Table 18-299

Table 18-299 Primary vtable for QPaintDevice

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPaintDevice
vfunc[0]:	QPaintDevice::~~QPaintDevice()
vfunc[1]:	QPaintDevice::~~QPaintDevice()
vfunc[2]:	NULL or QPaintDevice::devType() const
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	QPaintDevice::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QPaintDevice class is described by Table 18-300

Table 18-300 typeinfo for QPaintDevice

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QPaintDevice

18.5.13.2 Class data for QPixmap

The virtual table for the QPixmap class is described by Table 18-301

Table 18-301 Primary vtable for QPixmap

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPixmap
vfunc[0]:	QPixmap::~~QPixmap()
vfunc[1]:	QPixmap::~~QPixmap()
vfunc[2]:	QPixmap::devType() const
vfunc[3]:	QPixmap::paintEngine() const
vfunc[4]:	QPixmap::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QPixmap class is described by Table 18-302

Table 18-302 typeinfo for QPixmap

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPixmap
basetype:	typeinfo for QPaintDevice

18.5.13.3 Class data for QPicture

The virtual table for the QPicture class is described by Table 18-303

Table 18-303 Primary vtable for QPicture

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPicture
vfunc[0]:	QPicture::~~QPicture()
vfunc[1]:	QPicture::~~QPicture()
vfunc[2]:	QPicture::devType() const
vfunc[3]:	QPicture::paintEngine() const
vfunc[4]:	QPicture::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[5]:	QPicture::setData(char const*, unsigned int)

The Run Time Type Information for the QPicture class is described by Table 18-304

Table 18-304 typeinfo for QPicture

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPicture
basetype:	typeinfo for QPaintDevice

18.5.13.4 Class data for QBitmap

The virtual table for the QBitmap class is described by Table 18-305

Table 18-305 Primary vtable for QBitmap

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QBitmap
vfunc[0]:	QBitmap::~~QBitmap()
vfunc[1]:	QBitmap::~~QBitmap()

vfunc[2]:	QPixmap::devType() const
vfunc[3]:	QPixmap::paintEngine() const
vfunc[4]:	QPixmap::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QPixmap class is described by Table 18-306

Table 18-306 typeinfo for QPixmap

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPixmap
basetype:	typeinfo for QPixmap

18.5.13.5 Class data for QPictureFormatInterface

The virtual table for the QPictureFormatInterface class is described by Table 18-307

Table 18-307 Primary vtable for QPictureFormatInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPictureFormatInterface
vfunc[0]:	NULL or QPictureFormatInterface::~~QPictureFormatInterface()
vfunc[1]:	NULL or QPictureFormatInterface::~~QPictureFormatInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual

The Run Time Type Information for the QPictureFormatInterface class is described by Table 18-308

Table 18-308 typeinfo for QPictureFormatInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPictureFormatInterface
basetype:	typeinfo for QFactoryInterface

18.5.13.6 Class data for QPainter

The virtual table for the QPainter class is described by Table 18-309

Table 18-309 Primary vtable for QPainter

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPainter
vfunc[0]:	QPainter::~~QPainter()
vfunc[1]:	QPainter::~~QPainter()
vfunc[2]:	QPainter::devType() const
vfunc[3]:	QPainter::paintEngine() const
vfunc[4]:	QPainter::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QPainter class is described by Table 18-310

Table 18-310 typeinfo for QPainter

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPainter
basetype:	typeinfo for QPaintDevice

18.5.13.7 Class data for QPrintEngine

The virtual table for the QPrintEngine class is described by Table 18-311

Table 18-311 Primary vtable for QPrintEngine

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPrintEngine
vfunc[0]:	NULL or QPrintEngine::~~QPrintEngine()
vfunc[1]:	NULL or QPrintEngine::~~QPrintEngine()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual
vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual

The Run Time Type Information for the QPrintEngine class is described by Table 18-312

Table 18-312 typeid for QPrintEngine

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeid name for QPrintEngine

18.5.13.8 Interfaces for Qt4 Paint Device

An LSB conforming implementation shall provide the generic functions for Qt4 Paint Device specified in Table 18-313, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-313 libQtGui - Qt4 Paint Device Function Interfaces

_ZN10QPictureIO10setPictureERK8QPicture [QtGui]	_ZN10QPictureIO10setQualityEi [QtGui]
_ZN10QPictureIO11setFileNameERK7QString [QtGui]	_ZN10QPictureIO11setIODeviceEP9QIODevice [QtGui]
_ZN10QPictureIO12inputFormatsEv [QtGui]	_ZN10QPictureIO13outputFormatsEv [QtGui]
_ZN10QPictureIO13pictureFormatEP9QIODevice [QtGui]	_ZN10QPictureIO13pictureFormatERK7QString [QtGui]
_ZN10QPictureIO13setParametersEPKc [QtGui]	_ZN10QPictureIO14setDescriptionERK7QString [QtGui]
_ZN10QPictureIO15defineIOHandlerEPKcS1_S1_PFvPS_ES4_ [QtGui]	_ZN10QPictureIO4readEv [QtGui]
_ZN10QPictureIO5writeEv [QtGui]	_ZN10QPictureIO8setGammaEf [QtGui]
_ZN10QPictureIO9setFormatEPKc [QtGui]	_ZN10QPictureIO9setStatusEi [QtGui]
_ZN10QPictureIOC1EP9QIODevicePKc [QtGui]	_ZN10QPictureIOC1ERK7QStringPKc [QtGui]
_ZN10QPictureIOC1Ev [QtGui]	_ZN10QPictureIOC2EP9QIODevicePKc [QtGui]
_ZN10QPictureIOC2ERK7QStringPKc [QtGui]	_ZN10QPictureIOC2Ev [QtGui]
_ZN10QPictureIOD1Ev [QtGui]	_ZN10QPictureIOD2Ev [QtGui]
_ZN12QPaintDevice10x11AppDpiXEi [QtGui]	_ZN12QPaintDevice10x11AppDpiYEi [QtGui]
_ZN12QPaintDevice11x11AppCellsEi [QtGui]	_ZN12QPaintDevice11x11AppDepthEi [QtGui]
_ZN12QPaintDevice12x11AppScreenEv [QtGui]	_ZN12QPaintDevice12x11AppVisualEi [QtGui]

_ZN12QPaintDevice13x11AppDisplayEv [QtGui]	_ZN12QPaintDevice13x11SetAppDpiXEii [QtGui]
_ZN12QPaintDevice13x11SetAppDpiYEii [QtGui]	_ZN12QPaintDevice14x11AppColorMapEi [QtGui]
_ZN12QPaintDevice16x11AppRootWindowEi [QtGui]	_ZN12QPaintDevice19x11AppDefaultVisualEi [QtGui]
_ZN12QPaintDevice21x11AppDefaultColormapEi [QtGui]	_ZN12QPaintDeviceC1Ev [QtGui]
_ZN12QPaintDeviceC2Ev [QtGui]	_ZN12QPaintDeviceD0Ev [QtGui]
_ZN12QPaintDeviceD1Ev [QtGui]	_ZN12QPaintDeviceD2Ev [QtGui]
_ZN12QPixmapCache10cacheLimitEv [LSB]	_ZN12QPixmapCache13setCacheLimitEi [QtGui]
_ZN12QPixmapCache4findERK7QString [QtGui]	_ZN12QPixmapCache4findERK7QStringR7QPixmap [QtGui]
_ZN12QPixmapCache5clearEv [QtGui]	_ZN12QPixmapCache6insertERK7QStringR7QPixmap [QtGui]
_ZN12QPixmapCache6removeERK7QString [QtGui]	_ZN7QBitmap8fromDataERK5QSizePKhN6QImage6FormatE [QtGui]
_ZN7QBitmap9fromImageERK6QImage6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]	_ZN7QBitmapC1ERK5QSize [QtGui]
_ZN7QBitmapC1ERK5QSizePKhb [QtGui]	_ZN7QBitmapC1ERK7QPixmap [QtGui]
_ZN7QBitmapC1ERK7QStringPKc [QtGui]	_ZN7QBitmapC1Eii [QtGui]
_ZN7QBitmapC1EiiPKhb [QtGui]	_ZN7QBitmapC1Ev [QtGui]
_ZN7QBitmapC2ERK5QSize [QtGui]	_ZN7QBitmapC2ERK5QSizePKhb [QtGui]
_ZN7QBitmapC2ERK7QPixmap [QtGui]	_ZN7QBitmapC2ERK7QStringPKc [QtGui]
_ZN7QBitmapC2Eii [QtGui]	_ZN7QBitmapC2EiiPKhb [QtGui]
_ZN7QBitmapC2Ev [QtGui]	_ZN7QBitmapD0Ev [QtGui]
_ZN7QBitmapD1Ev [QtGui]	_ZN7QBitmapD2Ev [QtGui]
_ZN7QBitmapaSERK7QPixmap [QtGui]	_ZN7QPixmap10grabWidgetEP7QWidgetRK5QRect [QtGui]
_ZN7QPixmap10grabWindowEmiiii [QtGui]	_ZN7QPixmap10trueMatrixERK7QMatrixii [QtGui]
_ZN7QPixmap12defaultDepthEv [QtGui]	_ZN7QPixmap12loadFromDataEPKhjPKc6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]

_ZN7QPixmap12loadFromDataEPKh jPKcNS_9ColorModeE [QtGui]	_ZN7QPixmap12x11SetScreenEi [QtGui]
ZN7QPixmap15setAlphaChannelE RKS [QtGui]	_ZN7QPixmap16convertFromImage ERK6QImageNS_9ColorModeE [QtGui]
_ZN7QPixmap19x11SetDefaultScree nEi [LSB]	_ZN7QPixmap4fillEPK7QWidgetRK 6QPoint [QtGui]
_ZN7QPixmap4fillERK6QColor [QtGui]	_ZN7QPixmap4loadERK7QStringPK c6QFlagsIN2Qt19ImageConversionFl agEE [QtGui]
_ZN7QPixmap4loadERK7QStringPK cNS_9ColorModeE [QtGui]	_ZN7QPixmap6detachEv [QtGui]
_ZN7QPixmap7setMaskERK7QBitm ap [QtGui]	_ZN7QPixmap9fromImageERK6QIm age6QFlagsIN2Qt19ImageConversio nFlagEE [QtGui]
_ZN7QPixmapC1EPKPKc [QtGui]	_ZN7QPixmapC1ERK5QSize [QtGui]
_ZN7QPixmapC1ERK6QImage [QtGui]	_ZN7QPixmapC1ERK7QStringPKc6 QFlagsIN2Qt19ImageConversionFla gEE [QtGui]
_ZN7QPixmapC1ERK7QStringPKcN S_9ColorModeE [QtGui]	_ZN7QPixmapC1ERKS_ [QtGui]
_ZN7QPixmapC1Eii [QtGui]	_ZN7QPixmapC1Ev [QtGui]
_ZN7QPixmapC2EPKPKc [QtGui]	_ZN7QPixmapC2ERK5QSize [QtGui]
_ZN7QPixmapC2ERK6QImage [QtGui]	_ZN7QPixmapC2ERK7QStringPKc6 QFlagsIN2Qt19ImageConversionFla gEE [QtGui]
_ZN7QPixmapC2ERK7QStringPKcN S_9ColorModeE [QtGui]	_ZN7QPixmapC2ERKS_ [QtGui]
_ZN7QPixmapC2Eii [QtGui]	_ZN7QPixmapC2Ev [QtGui]
_ZN7QPixmapD0Ev [QtGui]	_ZN7QPixmapD1Ev [QtGui]
_ZN7QPixmapD2Ev [QtGui]	_ZN7QPixmapasERK6QImage [QtGui]
ZN7QPixmapasERKS [QtGui]	_ZN8QPicture12inputFormatsEv [QtGui]
_ZN8QPicture13outputFormatsEv [QtGui]	_ZN8QPicture13pictureFormatERK7 QString [QtGui]
_ZN8QPicture15inputFormatListEv [QtGui]	_ZN8QPicture15setBoundingRectER K5QRect [QtGui]
_ZN8QPicture16outputFormatListEv [QtGui]	_ZN8QPicture4loadEP9QIODeviceP Kc [QtGui]
_ZN8QPicture4loadERK7QStringPKc [QtGui]	_ZN8QPicture4playEP8QPainter [QtGui]

_ZN8QPicture4saveEP9QIODevicePKc [QtGui]	_ZN8QPicture4saveERK7QStringPKc [QtGui]
_ZN8QPicture6detachEv [QtGui]	_ZN8QPicture7setDataEPKcj [QtGui]
ZN8QPictureC1ERKS [QtGui]	_ZN8QPictureC1Ei [QtGui]
ZN8QPictureC2ERKS [QtGui]	_ZN8QPictureC2Ei [QtGui]
_ZN8QPictureD0Ev [QtGui]	_ZN8QPictureD1Ev [QtGui]
_ZN8QPictureD2Ev [QtGui]	_ZN8QPictureaSERKS_ [QtGui]
_ZN8QPrinter10setCreatorERK7QString [QtGui]	_ZN8QPrinter10setDocNameERK7QString [QtGui]
_ZN8QPrinter10setEnginesEP12QPrintEngineP12QPaintEngine [QtGui]	_ZN8QPrinter11setFullPageEb [QtGui]
_ZN8QPrinter11setPageSizeENS_8PageSizeE [QtGui]	_ZN8QPrinter12setColorModeENS_9ColorModeE [QtGui]
_ZN8QPrinter12setNumCopiesEi [QtGui]	_ZN8QPrinter12setPageOrderENS_9PageOrderE [QtGui]
_ZN8QPrinter13setPrintRangeENS_10PrintRangeE [QtGui]	_ZN8QPrinter13setResolutionEi [QtGui]
_ZN8QPrinter14setOrientationENS_11OrientationE [QtGui]	_ZN8QPrinter14setPaperSourceENS_11PaperSourceE [QtGui]
_ZN8QPrinter14setPrinterNameERK7QString [QtGui]	_ZN8QPrinter15setOutputFormatENS_12OutputFormatE [QtGui]
_ZN8QPrinter15setOutputToFileEb [QtGui]	_ZN8QPrinter15setPrintProgramERK7QString [QtGui]
_ZN8QPrinter16setCollateCopiesEb [QtGui]	_ZN8QPrinter16setOptionEnabledENS_13PrinterOptionEb [QtGui]
_ZN8QPrinter17setOutputFileNameERK7QString [QtGui]	_ZN8QPrinter22setDoubleSidedPrintingEb [QtGui]
_ZN8QPrinter23setCollateCopiesEnabledEb [QtGui]	_ZN8QPrinter23setFontEmbeddingEnabledEb [QtGui]
_ZN8QPrinter25setPrinterSelectionOptionERK7QString [QtGui]	_ZN8QPrinter5abortEv [QtGui]
_ZN8QPrinter5setupEP7QWidget [QtGui]	_ZN8QPrinter7newPageEv [QtGui]
_ZN8QPrinter9setFromToEii [QtGui]	_ZN8QPrinter9setMinMaxEii [QtGui]
_ZN8QPrinterC1ENS_11PrinterModeE [QtGui]	_ZN8QPrinterC2ENS_11PrinterModeE [QtGui]
_ZN8QPrinterD0Ev [QtGui]	_ZN8QPrinterD1Ev [QtGui]
_ZN8QPrinterD2Ev [QtGui]	_ZNK10QPictureIO10parametersEv [QtGui]

_Znk10QPictureIO11descriptionEv [QtGui]	_Znk10QPictureIO5gammaEv [QtGui]
_Znk10QPictureIO6formatEv [QtGui]	_Znk10QPictureIO6statusEv [QtGui]
_Znk10QPictureIO7pictureEv [QtGui]	_Znk10QPictureIO7qualityEv [QtGui]
_Znk10QPictureIO8fileNameEv [QtGui]	_Znk10QPictureIO8ioDeviceEv [QtGui]
_Znk12QPaintDevice10x11DisplayEv [QtGui]	_Znk12QPaintDevice11x11ColorMapEv [QtGui]
_Znk12QPaintDevice16x11DefaultVisualEv [QtGui]	_Znk12QPaintDevice18x11DefaultColorMapEv [QtGui]
_Znk12QPaintDevice6metricENS_17PaintDeviceMetricE [QtGui]	_Znk12QPaintDevice8x11CellsEv [QtGui]
_Znk12QPaintDevice8x11DepthEv [QtGui]	_Znk12QPaintDevice9x11ScreenEv [QtGui]
_Znk12QPaintDevice9x11VisualEv [QtGui]	_Znk7QBitmap11transformedERK7QMatrix [QtGui]
_Znk7QBitmapcv8QVariantEv [QtGui]	_Znk7QPixmap10isDetachedEv [QtGui]
_Znk7QPixmap11paintEngineEv [QtGui]	_Znk7QPixmap11transformedERK7QMatrixN2Qt18TransformationModeE [QtGui]
_Znk7QPixmap12alphaChannelEv [QtGui]	_Znk7QPixmap12serialNumberEv [QtGui]
_Znk7QPixmap13scaledToWidthEiN2Qt18TransformationModeE [QtGui]	_Znk7QPixmap14scaledToHeightEiN2Qt18TransformationModeE [QtGui]
_Znk7QPixmap15hasAlphaChannelEv [QtGui]	_Znk7QPixmap16x11PictureHandleEv [QtGui]
_Znk7QPixmap19createHeuristicMaskEv [QtGui]	_Znk7QPixmap19createMaskFromColorERK6QColor [QtGui]
_Znk7QPixmap4copyERK5QRect [QtGui]	_Znk7QPixmap4maskEv [QtGui]
_Znk7QPixmap4rectEv [QtGui]	_Znk7QPixmap4saveEP9QIODevicePKci [QtGui]
_Znk7QPixmap4saveERK7QStringPKci [QtGui]	_Znk7QPixmap4sizeEv [QtGui]
_Znk7QPixmap5depthEv [QtGui]	_Znk7QPixmap5widthEv [QtGui]
_Znk7QPixmap6handleEv [QtGui]	_Znk7QPixmap6heightEv [QtGui]
_Znk7QPixmap6isNullEv [QtGui]	_Znk7QPixmap6metricEN12QPaintDevice17PaintDeviceMetricE [QtGui]

_Znk7QPixmap6scaledERK5QSizeN 2Qt15AspectRatioModeENS3_18Tran sformationModeE [QtGui]	_Znk7QPixmap7devTypeEv [LSB]
_Znk7QPixmap7toImageEv [QtGui]	_Znk7QPixmap7x11InfoEv [QtGui]
_Znk7QPixmap8hasAlphaEv [QtGui]	_Znk7QPixmappcv8QVariantEv [QtGui]
_Znk8QPicture10isDetachedEv [QtGui]	_Znk8QPicture11paintEngineEv [QtGui]
_Znk8QPicture12boundingRectEv [QtGui]	_Znk8QPicture4dataEv [QtGui]
_Znk8QPicture4sizeEv [QtGui]	_Znk8QPicture6isNullEv [QtGui]
_Znk8QPicture6metricEN12QPaint Device17PaintDeviceMetricE [QtGui]	_Znk8QPicture7devTypeEv [LSB]
_Znk8QPrinter10printRangeEv [QtGui]	_Znk8QPrinter10resolutionEv [QtGui]
_Znk8QPrinter11orientationEv [QtGui]	_Znk8QPrinter11paintEngineEv [QtGui]
_Znk8QPrinter11paperSourceEv [QtGui]	_Znk8QPrinter11printEngineEv [QtGui]
_Znk8QPrinter11printerNameEv [QtGui]	_Znk8QPrinter12outputFormatEv [QtGui]
_Znk8QPrinter12printProgramEv [QtGui]	_Znk8QPrinter12printerStateEv [QtGui]
_Znk8QPrinter13collateCopiesEv [QtGui]	_Znk8QPrinter14outputFileNameEv [QtGui]
_Znk8QPrinter15isOptionEnabledE NS_13PrinterOptionE [QtGui]	_Znk8QPrinter19doubleSidedPrinti ngEv [QtXml]
_Znk8QPrinter20collateCopiesEnabl edEv [QtGui]	_Znk8QPrinter20fontEmbeddingEn abledEv [QtGui]
_Znk8QPrinter20supportedResoluti onsEv [QtGui]	_Znk8QPrinter22printerSelectionOp tionEv [QtGui]
_Znk8QPrinter6metricEN12QPaint Device17PaintDeviceMetricE [QtGui]	_Znk8QPrinter6toPageEv [QtGui]
_Znk8QPrinter7creatorEv [QtGui]	_Znk8QPrinter7devTypeEv [LSB]
_Znk8QPrinter7docNameEv [QtGui]	_Znk8QPrinter7maxPageEv [QtGui]
_Znk8QPrinter7minPageEv [QtGui]	_Znk8QPrinter8fromPageEv [QtGui]
_Znk8QPrinter8fullPageEv [QtGui]	_Znk8QPrinter8pageRectEv [QtGui]
_Znk8QPrinter8pageSizeEv [QtGui]	_Znk8QPrinter9colorModeEv [QtGui]

_ZNK8QPrinter9numCopiesEv [QtGui]	_ZNK8QPrinter9pageOrderEv [QtGui]
_ZNK8QPrinter9paperRectEv [QtGui]	_ZlsR11QDataStreamRK7QPixmap [QtGui]
_ZlsR11QDataStreamRK8QPicture [QtGui]	_ZrsR11QDataStreamR7QPixmap [QtGui]
_ZrsR11QDataStreamR8QPicture [QtGui]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Paint Device specified in Table 18-314, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-314 libQtGui - Qt4 Paint Device Deprecated Function Interfaces

_ZN12QPixmapCache4findERK7QSt ring [QtGui]	_ZN7QPixmap12x11SetScreenEi [QtGui]
_ZN7QPixmap19x11SetDefaultScreenEi [LSB]	_ZNK7QPixmap16x11PictureHandleEv [QtGui]
_ZNK7QPixmap6handleEv [QtGui]	_ZNK7QPixmap7x11InfoEv [QtGui]

18.5.14 Qt4 Multimedia

18.5.14.1 Class data for QImage

The virtual table for the QImage class is described by Table 18-315

Table 18-315 Primary vtable for QImage

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QImage
vfunc[0]:	QImage::~~QImage()
vfunc[1]:	QImage::~~QImage()
vfunc[2]:	QImage::devType() const
vfunc[3]:	QImage::paintEngine() const
vfunc[4]:	QImage::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QImage class is described by Table 18-316

Table 18-316 typeinfo for QImage

Base Vtable	vtable for __cxxabiv1::__si_class_type_info

Name	typeinfo name for QImage
basetype:	typeinfo for QPainterDevice

18.5.14.2 Class data for QImageIOHandler

The virtual table for the QImageIOHandler class is described by Table 18-317

Table 18-317 Primary vtable for QImageIOHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QImageIOHandler
vfunc[0]:	QImageIOHandler::~~QImageIOHandler()
vfunc[1]:	QImageIOHandler::~~QImageIOHandler()
vfunc[2]:	QImageIOHandler::name() const
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	QImageIOHandler::write(QImage const&)
vfunc[6]:	QImageIOHandler::option(QImageIOHandler::ImageOption) const
vfunc[7]:	QImageIOHandler::setOption(QImageIOHandler::ImageOption, QVariant const&)
vfunc[8]:	QImageIOHandler::supportsOption(QImageIOHandler::ImageOption) const
vfunc[9]:	QImageIOHandler::jumpToNextImage()
vfunc[10]:	QImageIOHandler::jumpToImage(int)
vfunc[11]:	QImageIOHandler::loopCount() const
vfunc[12]:	QImageIOHandler::imageCount() const
vfunc[13]:	QImageIOHandler::nextImageDelay() const
vfunc[14]:	QImageIOHandler::currentImageNumber() const
vfunc[15]:	QImageIOHandler::currentImageRect() const

The Run Time Type Information for the QImageIOHandler class is described by Table 18-318

Table 18-318 typeinfo for QImageIOHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QImageIOHandler

18.5.14.3 Class data for QImageIOHandlerFactoryInterface

The virtual table for the QImageIOHandlerFactoryInterface class is described by Table 18-319

Table 18-319 Primary vtable for QImageIOHandlerFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QImageIOHandlerFactoryInterface
vfunc[0]:	NULL or QImageIOHandlerFactoryInterface::~~ QImageIOHandlerFactoryInterface()
vfunc[1]:	NULL or QImageIOHandlerFactoryInterface::~~ QImageIOHandlerFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QImageIOHandlerFactoryInterface class is described by Table 18-320

Table 18-320 typeinfo for QImageIOHandlerFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QImageIOHandlerFactoryInterface
basetype:	typeinfo for QFactoryInterface

18.5.14.4 Class data for QMovie

The virtual table for the QMovie class is described by Table 18-321

Table 18-321 Primary vtable for QMovie

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMovie

vfunc[0]:	QMovie::metaObject() const
vfunc[1]:	QMovie::qt_metacast(char const*)
vfunc[2]:	QMovie::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMovie::~~QMovie()
vfunc[4]:	QMovie::~~QMovie()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QMovie class is described by Table 18-322

Table 18-322 typeinfo for QMovie

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMovie
basetype:	typeinfo for QObject

18.5.14.5 Class data for QSound

The virtual table for the QSound class is described by Table 18-323

Table 18-323 Primary vtable for QSound

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSound
vfunc[0]:	QSound::metaObject() const
vfunc[1]:	QSound::qt_metacast(char const*)
vfunc[2]:	QSound::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSound::~~QSound()
vfunc[4]:	QSound::~~QSound()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QSound class is described by Table 18-324

Table 18-324 typeinfo for QSound

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSound
basetype:	typeinfo for QObject

18.5.14.6 Interfaces for Qt4 Multimedia

An LSB conforming implementation shall provide the generic functions for Qt4 Multimedia specified in Table 18-325, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-325 libQtGui - Qt4 Multimedia Function Interfaces

_ZN12QImageReader10setQualityEi [QtXml]	_ZN12QImageReader11imageFormatEP9QIODevice [QtGui]
_ZN12QImageReader11imageFormatERK7QString [QtGui]	_ZN12QImageReader11jumpToImageEi [QtGui]
_ZN12QImageReader11setClipRectERK5QRect [QtGui]	_ZN12QImageReader11setFileNameERK7QString [QtGui]
_ZN12QImageReader13setScaledSizeERK5QSize [QtGui]	_ZN12QImageReader15jumpToNextImageEv [QtGui]
_ZN12QImageReader17setScaledClipRectERK5QRect [QtGui]	_ZN12QImageReader18setBackgroundColorsERK6QColor [QtGui]
_ZN12QImageReader21supportedImageFormatsEv [QtGui]	_ZN12QImageReader4readEP6QImage [QtXml]
_ZN12QImageReader4readEv [QtGui]	_ZN12QImageReader9setDeviceEP9QIODevice [QtGui]
_ZN12QImageReader9setFormatERK10QByteArray [QtGui]	_ZN12QImageReaderC1EP9QIODeviceRK10QByteArray [QtGui]
_ZN12QImageReaderC1ERK7QStringRK10QByteArray [QtGui]	_ZN12QImageReaderC1Ev [QtGui]

_ZN12QImageReaderC2EP9QIODeviceRK10QByteArray [QtGui]	_ZN12QImageReaderC2ERK7QStringRK10QByteArray [QtGui]
_ZN12QImageReaderC2Ev [QtGui]	_ZN12QImageReaderD1Ev [QtGui]
_ZN12QImageReaderD2Ev [QtGui]	_ZN12QImageWriter10setQualityEi [QtGui]
_ZN12QImageWriter11setFileNameERK7QString [QtGui]	_ZN12QImageWriter14setDescriptionERK7QString [QtGui]
_ZN12QImageWriter21supportedImageFormatsEv [QtGui]	_ZN12QImageWriter5writeERK6QImage [QtGui]
ZN12QImageWriter7setTextERK7QStringS2 [QtGui]	_ZN12QImageWriter8setGammaEf [QtGui]
_ZN12QImageWriter9setDeviceEP9QIODevice [QtGui]	_ZN12QImageWriter9setFormatERK10QByteArray [QtGui]
_ZN12QImageWriterC1EP9QIODeviceRK10QByteArray [QtGui]	_ZN12QImageWriterC1ERK7QStringRK10QByteArray [QtGui]
_ZN12QImageWriterC1Ev [QtGui]	_ZN12QImageWriterC2EP9QIODeviceRK10QByteArray [QtGui]
_ZN12QImageWriterC2ERK7QStringRK10QByteArray [QtGui]	_ZN12QImageWriterC2Ev [QtGui]
_ZN12QImageWriterD1Ev [QtGui]	_ZN12QImageWriterD2Ev [QtGui]
_ZN15QImageIOHandler11jumpToImageEi [QtGui]	_ZN15QImageIOHandler15jumpToNextImageEv [QtGui]
_ZN15QImageIOHandler5writeERK6QImage [QtGui]	_ZN15QImageIOHandler9setDeviceEP9QIODevice [QtGui]
_ZN15QImageIOHandler9setFormatERK10QByteArray [QtGui]	_ZN15QImageIOHandler9setOptionENS_11ImageOptionERK8QVariant [QtGui]
_ZN15QImageIOHandlerC1Ev [QtGui]	_ZN15QImageIOHandlerC2Ev [QtGui]
_ZN15QImageIOHandlerD0Ev [QtGui]	_ZN15QImageIOHandlerD1Ev [QtGui]
_ZN15QImageIOHandlerD2Ev [QtGui]	_ZN6QImage10trueMatrixERK7QMatrixii [QtGui]
_ZN6QImage12invertPixelsENS_10InvertModeE [QtGui]	_ZN6QImage12loadFromDataEPKhipKc [QtGui]
_ZN6QImage12setNumColorsEi [QtGui]	_ZN6QImage13setColorTableE7QVectorIjE [QtGui]
_ZN6QImage14setAlphaBufferEb [QtGui]	_ZN6QImage15setAlphaChannelERKS_ [QtGui]
_ZN6QImage16setDotsPerMeterXEi [QtGui]	_ZN6QImage16setDotsPerMeterYEi [QtGui]
_ZN6QImage4bitsEv [QtGui]	_ZN6QImage4fillEj [QtGui]

_ZN6QImage4loadEP9QIODevicePKc [QtGui]	_ZN6QImage4loadERK7QStringPKc [QtGui]
_ZN6QImage6createERK5QSizeiiNS_6EndianE [QtGui]	_ZN6QImage6createEiiiiNS_6EndianE [QtGui]
_ZN6QImage6detachEv [QtGui]	_ZN6QImage7setTextEPKcS1_RK7QString [QtGui]
ZN6QImage7setTextERK7QStringS2 [QtGui]	_ZN6QImage8fromDataEPKhiPKc [QtGui]
_ZN6QImage8scanLineEi [QtGui]	_ZN6QImage8setColorEij [QtGui]
_ZN6QImage8setPixelEij [QtGui]	_ZN6QImage9jumpTableEv [QtGui]
_ZN6QImage9setOffsetERK6QPoint [QtGui]	_ZN6QImageC1EPKPKc [QtGui]
ZN6QImageC1EPKcS1 [QtGui]	_ZN6QImageC1EPKhiiNS_6FormatE [QtXml]
_ZN6QImageC1EPkhiiNS_6FormatE [QtGui]	_ZN6QImageC1EPkhiiPKjiNS_6EndianE [QtGui]
_ZN6QImageC1ERK5QSizeNS_6FormatE [QtGui]	_ZN6QImageC1ERK5QSizeiiNS_6EndianE [QtGui]
_ZN6QImageC1ERK7QStringPKc [QtGui]	_ZN6QImageC1ERKS_ [QtGui]
_ZN6QImageC1EiiNS_6FormatE [QtGui]	_ZN6QImageC1EiiiiNS_6EndianE [QtGui]
_ZN6QImageC1Ev [QtGui]	_ZN6QImageC2EPKPKc [QtGui]
ZN6QImageC2EPKcS1 [QtGui]	_ZN6QImageC2EPKhiiNS_6FormatE [QtXml]
_ZN6QImageC2EPkhiiNS_6FormatE [QtGui]	_ZN6QImageC2EPkhiiPKjiNS_6EndianE [QtGui]
_ZN6QImageC2ERK5QSizeNS_6FormatE [QtGui]	_ZN6QImageC2ERK5QSizeiiNS_6EndianE [QtGui]
_ZN6QImageC2ERK7QStringPKc [QtGui]	_ZN6QImageC2ERKS_ [QtGui]
_ZN6QImageC2EiiNS_6FormatE [QtGui]	_ZN6QImageC2EiiiiNS_6EndianE [QtGui]
_ZN6QImageC2Ev [QtGui]	_ZN6QImageD0Ev [QtGui]
_ZN6QImageD1Ev [QtGui]	_ZN6QImageD2Ev [QtGui]
ZN6QImageaSERKS [QtGui]	_ZN6QMovie10scaledSizeEv [QtGui]
_ZN6QMovie11jumpToFrameEi [QtGui]	_ZN6QMovie11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN6QMovie11qt_metacastEPKc [QtGui]	_ZN6QMovie11setFileNameERK7QString [QtGui]

_ZN6QMovie12frameChangedEi [QtGui]	_ZN6QMovie12setCacheModeENS_9 CacheModeE [QtGui]
_ZN6QMovie12stateChangedENS_10 MovieStateE [QtGui]	_ZN6QMovie13setScaledSizeERK5QSize [QtGui]
_ZN6QMovie15jumpToNextFrameEv [QtGui]	_ZN6QMovie16supportedFormatsEv [QtGui]
_ZN6QMovie18setBackgroundColor ERK6QColor [QtGui]	_ZN6QMovie4stopEv [QtGui]
_ZN6QMovie5errorEN12QImageReader16ImageReaderErrorE [QtGui]	_ZN6QMovie5startEv [QtGui]
_ZN6QMovie7resizedERK5QSize [QtGui]	_ZN6QMovie7startedEv [QtGui]
_ZN6QMovie7updatedERK5QRect [QtGui]	_ZN6QMovie8finishedEv [QtGui]
_ZN6QMovie8setSpeedEi [QtGui]	_ZN6QMovie9cacheModeEv [QtGui]
_ZN6QMovie9setDeviceEP9QIODevice [QtGui]	_ZN6QMovie9setFormatERK10QByteArray [QtGui]
_ZN6QMovie9setPausedEb [QtGui]	_ZN6QMovieC1EP7QObject [QtGui]
_ZN6QMovieC1EP9QIODeviceRK10QByteArrayP7QObject [QtGui]	_ZN6QMovieC1ERK7QStringRK10QByteArrayP7QObject [QtGui]
_ZN6QMovieC2EP7QObject [QtGui]	_ZN6QMovieC2EP9QIODeviceRK10QByteArrayP7QObject [QtGui]
_ZN6QMovieC2ERK7QStringRK10QByteArrayP7QObject [QtGui]	_ZN6QMovieD0Ev [QtGui]
_ZN6QMovieD1Ev [QtGui]	_ZN6QMovieD2Ev [QtGui]
_ZN6QSound11isAvailableEv [QtGui]	_ZN6QSound11qt_metacallEN11QMetaObject4CalleiPPv [QtGui]
_ZN6QSound11qt_metacastEPKc [QtGui]	_ZN6QSound4playERK7QString [QtGui]
_ZN6QSound4playEv [QtGui]	_ZN6QSound4stopEv [QtGui]
_ZN6QSound8setLoopsEi [QtGui]	_ZN6QSoundC1ERK7QStringP7QObject [QtGui]
_ZN6QSoundC1ERK7QStringP7QObjectPKc [QtGui]	_ZN6QSoundC2ERK7QStringP7QObject [QtGui]
_ZN6QSoundC2ERK7QStringP7QObjectPKc [QtGui]	_ZN6QSoundD0Ev [QtGui]
_ZN6QSoundD1Ev [QtGui]	_ZN6QSoundD2Ev [QtGui]
_ZNK12QImageReader10imageCountEv [QtGui]	_ZNK12QImageReader10scaledSizeEv [QtGui]
_ZNK12QImageReader11errorStringEv [QtGui]	_ZNK12QImageReader14nextImageDelayEv [QtGui]

_ZNK12QImageReader14scaledClipRectEv [QtGui]	_ZNK12QImageReader14supportsOptionEN15QImageIOHandler11ImageOptionE [QtXml]
_ZNK12QImageReader15backgroundColorEv [QtGui]	_ZNK12QImageReader16currentImageRectEv [QtGui]
_ZNK12QImageReader17supportsAnimationEv [QtGui]	_ZNK12QImageReader18currentImageNumberEv [QtGui]
_ZNK12QImageReader4sizeEv [QtGui]	_ZNK12QImageReader4textERK7QString [QtGui]
_ZNK12QImageReader5errorEv [QtGui]	_ZNK12QImageReader6deviceEv [QtGui]
_ZNK12QImageReader6formatEv [QtGui]	_ZNK12QImageReader7canReadEv [QtGui]
_ZNK12QImageReader7qualityEv [QtXml]	_ZNK12QImageReader8clipRectEv [QtGui]
_ZNK12QImageReader8fileNameEv [QtGui]	_ZNK12QImageReader8textKeysEv [QtGui]
_ZNK12QImageReader9loopCountEv [QtGui]	_ZNK12QImageWriter11descriptionEv [QtGui]
_ZNK12QImageWriter11errorStringEv [QtGui]	_ZNK12QImageWriter14supportsOptionEN15QImageIOHandler11ImageOptionE [QtXml]
_ZNK12QImageWriter5errorEv [QtGui]	_ZNK12QImageWriter5gammaEv [QtGui]
_ZNK12QImageWriter6deviceEv [QtGui]	_ZNK12QImageWriter6formatEv [QtGui]
_ZNK12QImageWriter7qualityEv [QtGui]	_ZNK12QImageWriter8canWriteEv [QtGui]
_ZNK12QImageWriter8fileNameEv [QtGui]	_ZNK15QImageIOHandler10imageCountEv [QtGui]
_ZNK15QImageIOHandler14nextImageDelayEv [QtGui]	_ZNK15QImageIOHandler14supportOptionENS_11ImageOptionE [QtGui]
_ZNK15QImageIOHandler16currentImageRectEv [QtGui]	_ZNK15QImageIOHandler18currentImageNumberEv [QtGui]
_ZNK15QImageIOHandler4nameEv [QtGui]	_ZNK15QImageIOHandler6deviceEv [QtGui]
_ZNK15QImageIOHandler6formatEv [QtGui]	_ZNK15QImageIOHandler6optionENS_11ImageOptionE [QtGui]
_ZNK15QImageIOHandler9loopCountEv [QtGui]	_ZNK15QImageIOHandler9setFormatERK10QByteArray [QtGui]
_ZNK6QImage10colorTableEv [QtGui]	_ZNK6QImage10isDetachedEv [QtGui]

_ZNK6QImage10pixelIndexEii [QtGui]	_ZNK6QImage10rgbSwappedEv [QtGui]
_ZNK6QImage11isGrayscaleEv [QtGui]	_ZNK6QImage11paintEngineEv [QtGui]
_ZNK6QImage11transformedERK7QMatrixN2Qt18TransformationModeE [QtGui]	_ZNK6QImage12alphaChannelEv [QtGui]
_ZNK6QImage12bytesPerLineEv [QtGui]	_ZNK6QImage12convertDepthEi6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]
_ZNK6QImage12serialNumberEv [QtGui]	_ZNK6QImage13dotsPerMeterXEiv [QtGui]
_ZNK6QImage13dotsPerMeterYEiv [QtGui]	_ZNK6QImage13scaledToWidthEiN2Qt18TransformationModeE [QtGui]
_ZNK6QImage13textLanguagesEv [QtGui]	_ZNK6QImage14hasAlphaBufferEv [QtGui]
_ZNK6QImage14scaledToHeightEiN2Qt18TransformationModeE [QtGui]	_ZNK6QImage15convertBitOrderENS_6EndianE [QtGui]
_ZNK6QImage15convertToFormatENS_6FormatE6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]	_ZNK6QImage15convertToFormatENS_6FormatERK7QVectorIjE6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]
_ZNK6QImage15createAlphaMaskE6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]	_ZNK6QImage15hasAlphaChannelEv [QtGui]
_ZNK6QImage19createHeuristicMaskEb [QtGui]	_ZNK6QImage23convertDepthWithPaletteEiPji6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]
_ZNK6QImage4bitsEv [QtGui]	_ZNK6QImage4copyERK5QRect [QtGui]
_ZNK6QImage4rectEv [QtGui]	_ZNK6QImage4saveEP9QIODevicePKci [QtGui]
_ZNK6QImage4saveERK7QStringPKci [QtGui]	_ZNK6QImage4sizeEv [QtGui]
ZNK6QImage4textEPKcS1 [QtGui]	_ZNK6QImage4textERK17QImageTextKeyLang [QtGui]
_ZNK6QImage4textERK7QString [QtGui]	_ZNK6QImage5colorEi [QtGui]
_ZNK6QImage5depthEv [QtGui]	_ZNK6QImage5pixelEii [QtGui]
_ZNK6QImage5validEii [QtGui]	_ZNK6QImage5widthEv [QtGui]
_ZNK6QImage6formatEv [QtGui]	_ZNK6QImage6heightEv [QtGui]
_ZNK6QImage6isNullEv [QtGui]	_ZNK6QImage6metricEN12QPaintDevice17PaintDeviceMetricE [QtGui]

_Znk6QImage6offsetEv [QtGui]	_Znk6QImage6scaledERK5QSizeN2Qt15AspectRatioModeENS3_18TransformationModeE [QtGui]
_Znk6QImage7allGrayEv [QtGui]	_Znk6QImage7devTypeEv [LSB]
_Znk6QImage8mirroredEbb [QtGui]	_Znk6QImage8numBytesEv [QtGui]
_Znk6QImage8scanLineEi [QtGui]	_Znk6QImage8textKeysEv [QtGui]
_Znk6QImage8textListEv [QtGui]	_Znk6QImage9jumpTableEv [QtGui]
_Znk6QImage9numColorsEv [QtGui]	_Znk6QImagecv8QVariantEv [QtGui]
Znk6QImageeqERKS [QtGui]	_Znk6QImageeneERKS_ [QtGui]
_Znk6QMovie10frameCountEv [QtGui]	_Znk6QMovie10metaObjectEv [QtGui]
_Znk6QMovie12currentImageEv [QtGui]	_Znk6QMovie13currentPixmapEv [QtGui]
_Znk6QMovie14nextFrameDelayEv [QtGui]	_Znk6QMovie15backgroundColorEv [QtGui]
_Znk6QMovie18currentFrameNumberEv [QtGui]	_Znk6QMovie5speedEv [QtGui]
_Znk6QMovie5stateEv [QtGui]	_Znk6QMovie6deviceEv [QtGui]
_Znk6QMovie6formatEv [QtGui]	_Znk6QMovie7isValidEv [QtGui]
_Znk6QMovie8fileNameEv [QtGui]	_Znk6QMovie9cacheModeEv [QtXml]
_Znk6QMovie9frameRectEv [QtGui]	_Znk6QMovie9loopCountEv [QtGui]
_Znk6QSound10isFinishedEv [QtGui]	_Znk6QSound10metaObjectEv [QtGui]
_Znk6QSound14loopsRemainingEv [QtGui]	_Znk6QSound5loopsEv [QtGui]
_Znk6QSound8fileNameEv [QtGui]	_ZlsR11QDataStreamRK6QImage [QtGui]
_ZrsR11QDataStreamR6QImage [QtGui]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Multimedia specified in Table 18-326, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-326 libQtGui - Qt4 Multimedia Deprecated Function Interfaces

_ZN12QImageWriter14setDescriptionERK7QString [QtGui]	_ZN6QImage7setTextEPKcS1_RK7QString [QtGui]
_ZNK12QImageWriter11descriptionEv [QtGui]	_ZNK15QImageIOHandler4nameEv [QtGui]
ZNK6QImage4textEPKcS1 [QtGui]	_ZNK6QImage4textERK17QImageTextKeyLang [QtGui]
_ZNK6QImage8textListEv [QtGui]	

18.5.15 Qt4 Layouts

18.5.15.1 Class data for QLayoutItem

The virtual table for the QLayoutItem class is described by Table 18-327

Table 18-327 Primary vtable for QLayoutItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QLayoutItem
vfunc[0]:	QLayoutItem::~~QLayoutItem()
vfunc[1]:	QLayoutItem::~~QLayoutItem()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual
vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual
vfunc[8]:	__cxa_pure_virtual
vfunc[9]:	QLayoutItem::hasHeightForWidth() const
vfunc[10]:	QLayoutItem::heightForWidth(int) const
vfunc[11]:	QLayoutItem::minimumHeightForWidth(int) const
vfunc[12]:	QLayoutItem::invalidate()
vfunc[13]:	QLayoutItem::widget()
vfunc[14]:	QLayoutItem::layout()
vfunc[15]:	QLayoutItem::spacerItem()

The Run Time Type Information for the QLayoutItem class is described by Table 18-328

Table 18-328 typeinfo for QLayoutItem

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QLayoutItem

18.5.15.2 Class data for QSpacerItem

The virtual table for the QSpacerItem class is described by Table 18-329

Table 18-329 Primary vtable for QSpacerItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSpacerItem
vfunc[0]:	NULL or QSpacerItem::~~QSpacerItem()
vfunc[1]:	NULL or QSpacerItem::~~QSpacerItem()
vfunc[2]:	QSpacerItem::sizeHint() const
vfunc[3]:	QSpacerItem::minimumSize() const
vfunc[4]:	QSpacerItem::maximumSize() const
vfunc[5]:	QSpacerItem::expandingDirections() const
vfunc[6]:	QSpacerItem::setGeometry(QRect const&)
vfunc[7]:	QSpacerItem::geometry() const
vfunc[8]:	QSpacerItem::isEmpty() const
vfunc[9]:	QLayoutItem::hasHeightForWidth() const
vfunc[10]:	QLayoutItem::heightForWidth(int) const
vfunc[11]:	QLayoutItem::minimumHeightForW idth(int) const
vfunc[12]:	QLayoutItem::invalidate()
vfunc[13]:	QLayoutItem::widget()
vfunc[14]:	QLayoutItem::layout()
vfunc[15]:	QSpacerItem::spacerItem()

The Run Time Type Information for the QSpacerItem class is described by Table 18-330

Table 18-330 typeinfo for QSpacerItem

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSpacerItem
basetype:	typeinfo for QLayoutItem

18.5.15.3 Class data for QWidgetItem

The virtual table for the QWidgetItem class is described by Table 18-331

Table 18-331 Primary vtable for QWidgetItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWidgetItem
vfunc[0]:	NULL or QWidgetItem::~~QWidgetItem()
vfunc[1]:	NULL or QWidgetItem::~~QWidgetItem()
vfunc[2]:	QWidgetItem::sizeHint() const
vfunc[3]:	QWidgetItem::minimumSize() const
vfunc[4]:	QWidgetItem::maximumSize() const
vfunc[5]:	QWidgetItem::expandingDirections() const
vfunc[6]:	QWidgetItem::setGeometry(QRect const&)
vfunc[7]:	QWidgetItem::geometry() const
vfunc[8]:	QWidgetItem::isEmpty() const
vfunc[9]:	QWidgetItem::hasHeightForWidth() const
vfunc[10]:	QWidgetItem::heightForWidth(int) const
vfunc[11]:	QLayoutItem::minimumHeightForW idth(int) const
vfunc[12]:	QLayoutItem::invalidate()
vfunc[13]:	QWidgetItem::widget()
vfunc[14]:	QLayoutItem::layout()
vfunc[15]:	QLayoutItem::spacerItem()

The Run Time Type Information for the QWidgetItem class is described by Table 18-332

Table 18-332 typeinfo for QWidgetItem

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWidgetItem
basetype:	typeinfo for QLayoutItem

18.5.15.4 Class data for QLayout

The virtual table for the QLayout class is described by Table 18-333

Table 18-333 Primary vtable for QLayout

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QLayout
vfunc[0]:	QLayout::metaObject() const
vfunc[1]:	QLayout::qt_metacast(char const*)
vfunc[2]:	QLayout::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QLayout::~~QLayout()
vfunc[4]:	QLayout::~~QLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QLayout::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QLayout::invalidate()
vfunc[13]:	QLayout::geometry() const
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	QLayout::expandingDirections() const
vfunc[16]:	QLayout::minimumSize() const
vfunc[17]:	QLayout::maximumSize() const
vfunc[18]:	__cxa_pure_virtual
vfunc[19]:	__cxa_pure_virtual
vfunc[20]:	__cxa_pure_virtual

vfunc[21]:	QLayout::indexOf(QWidget*) const
vfunc[22]:	__cxa_pure_virtual
vfunc[23]:	QLayout::isEmpty() const
vfunc[24]:	QLayout::layout()

The Run Time Type Information for the QLayout class is described by Table 18-334

Table 18-334 typeinfo for QLayout

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2 2050
Name	typeinfo name for QLayout	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QLayoutItem	

18.5.15.5 Class data for QGridLayout

The virtual table for the QGridLayout class is described by Table 18-335

Table 18-335 Primary vtable for QGridLayout

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGridLayout
vfunc[0]:	QGridLayout::metaObject() const
vfunc[1]:	QGridLayout::qt_metacast(char const*)
vfunc[2]:	QGridLayout::qt_metacall(QMetaObj ect::Call, int, void**)
vfunc[3]:	QGridLayout::~~QGridLayout()
vfunc[4]:	QGridLayout::~~QGridLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QLayout::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QGridLayout::invalidate()
vfunc[13]:	QLayout::geometry() const
vfunc[14]:	QGridLayout::addItem(QLayoutItem*)
vfunc[15]:	QGridLayout::expandingDirections() const
vfunc[16]:	QGridLayout::minimumSize() const
vfunc[17]:	QGridLayout::maximumSize() const
vfunc[18]:	QGridLayout::setGeometry(QRect const&)
vfunc[19]:	QGridLayout::itemAt(int) const
vfunc[20]:	QGridLayout::takeAt(int)
vfunc[21]:	QLayout::indexOf(QWidget*) const
vfunc[22]:	QGridLayout::count() const
vfunc[23]:	QLayout::isEmpty() const
vfunc[24]:	QLayout::layout()
vfunc[25]:	QGridLayout::sizeHint() const
vfunc[26]:	QGridLayout::hasHeightForWidth() const
vfunc[27]:	QGridLayout::heightForWidth(int) const
vfunc[28]:	QGridLayout::minimumHeightForWidth(int) const

The Run Time Type Information for the QGridLayout class is described by Table 18-336

Table 18-336 typeinfo for QGridLayout

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QGridLayout
basetype:	typeinfo for QLayout

18.5.15.6 Class data for QBoxLayout

The virtual table for the QBoxLayout class is described by Table 18-337

Table 18-337 Primary vtable for QBoxLayout

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QBoxLayout
vfunc[0]:	QBoxLayout::metaObject() const
vfunc[1]:	QBoxLayout::qt_metacast(char const*)
vfunc[2]:	QBoxLayout::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QBoxLayout::~~QBoxLayout()
vfunc[4]:	QBoxLayout::~QBoxLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QLayout::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QBoxLayout::invalidate()
vfunc[13]:	QLayout::geometry() const
vfunc[14]:	QBoxLayout::addItem(QLayoutItem*)
vfunc[15]:	QBoxLayout::expandingDirections() const
vfunc[16]:	QBoxLayout::minimumSize() const
vfunc[17]:	QBoxLayout::maximumSize() const
vfunc[18]:	QBoxLayout::setGeometry(QRect const&)
vfunc[19]:	QBoxLayout::itemAt(int) const
vfunc[20]:	QBoxLayout::takeAt(int)
vfunc[21]:	QLayout::indexOf(QWidget*) const
vfunc[22]:	QBoxLayout::count() const
vfunc[23]:	QLayout::isEmpty() const
vfunc[24]:	QLayout::layout()
vfunc[25]:	QBoxLayout::sizeHint() const
vfunc[26]:	QBoxLayout::hasHeightForWidth() const

vfunc[27]:	QBoxLayout::heightForWidth(int) const
vfunc[28]:	QBoxLayout::minimumHeightForWidth(int) const

The Run Time Type Information for the QBoxLayout class is described by Table 18-338

Table 18-338 typeinfo for QBoxLayout

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QBoxLayout
basetype:	typeinfo for QLayout

18.5.15.7 Class data for QHBoxLayout

The virtual table for the QHBoxLayout class is described by Table 18-339

Table 18-339 Primary vtable for QHBoxLayout

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHBoxLayout
vfunc[0]:	QHBoxLayout::metaObject() const
vfunc[1]:	QHBoxLayout::qt_metacast(char const*)
vfunc[2]:	QHBoxLayout::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QHBoxLayout::~~QHBoxLayout()
vfunc[4]:	QHBoxLayout::~~QHBoxLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QLayout::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QBoxLayout::invalidate()
vfunc[13]:	QLayout::geometry() const

vfunc[14]:	QBoxLayout::addItem(QLayoutItem*)
vfunc[15]:	QBoxLayout::expandingDirections() const
vfunc[16]:	QBoxLayout::minimumSize() const
vfunc[17]:	QBoxLayout::maximumSize() const
vfunc[18]:	QBoxLayout::setGeometry(QRect const&)
vfunc[19]:	QBoxLayout::itemAt(int) const
vfunc[20]:	QBoxLayout::takeAt(int)
vfunc[21]:	QLayout::indexOf(QWidget*) const
vfunc[22]:	QBoxLayout::count() const
vfunc[23]:	QLayout::isEmpty() const
vfunc[24]:	QLayout::layout()
vfunc[25]:	QBoxLayout::sizeHint() const
vfunc[26]:	QBoxLayout::hasHeightForWidth() const
vfunc[27]:	QBoxLayout::heightForWidth(int) const
vfunc[28]:	QBoxLayout::minimumHeightForWidth(int) const

The Run Time Type Information for the QHBoxLayout class is described by Table 18-340

Table 18-340 typeinfo for QHBoxLayout

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHBoxLayout
basetype:	typeinfo for QBoxLayout

18.5.15.8 Class data for QVBoxLayout

The virtual table for the QVBoxLayout class is described by Table 18-341

Table 18-341 Primary vtable for QVBoxLayout

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QVBoxLayout
vfunc[0]:	QVBoxLayout::metaObject() const

vfunc[1]:	QVBoxLayout::qt_metacast(char const*)
vfunc[2]:	QVBoxLayout::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QVBoxLayout::~~VBoxLayout()
vfunc[4]:	QVBoxLayout::~~VBoxLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QLayout::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QBoxLayout::invalidate()
vfunc[13]:	QLayout::geometry() const
vfunc[14]:	QBoxLayout::addItem(QLayoutItem*)
vfunc[15]:	QBoxLayout::expandingDirections() const
vfunc[16]:	QBoxLayout::minimumSize() const
vfunc[17]:	QBoxLayout::maximumSize() const
vfunc[18]:	QBoxLayout::setGeometry(QRect const&)
vfunc[19]:	QBoxLayout::itemAt(int) const
vfunc[20]:	QBoxLayout::takeAt(int)
vfunc[21]:	QLayout::indexOf(QWidget*) const
vfunc[22]:	QBoxLayout::count() const
vfunc[23]:	QLayout::isEmpty() const
vfunc[24]:	QLayout::layout()
vfunc[25]:	QBoxLayout::sizeHint() const
vfunc[26]:	QBoxLayout::hasHeightForWidth() const
vfunc[27]:	QBoxLayout::heightForWidth(int) const
vfunc[28]:	QBoxLayout::minimumHeightForWidth(int) const

The Run Time Type Information for the QVBoxLayout class is described by Table 18-342

Table 18-342 typeinfo for QVBoxLayout

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QVBoxLayout
basetype:	typeinfo for QBoxLayout

18.5.15.9 Class data for QStackedLayout

The virtual table for the QStackedLayout class is described by Table 18-343

Table 18-343 Primary vtable for QStackedLayout

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStackedLayout
vfunc[0]:	QStackedLayout::metaObject() const
vfunc[1]:	QStackedLayout::qt_metacast(char const*)
vfunc[2]:	QStackedLayout::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QStackedLayout::~~QStackedLayout()
vfunc[4]:	QStackedLayout::~~QStackedLayout()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QLayout::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QLayout::invalidate()
vfunc[13]:	QLayout::geometry() const
vfunc[14]:	QStackedLayout::addItem(QLayoutItem*)
vfunc[15]:	QLayout::expandingDirections() const
vfunc[16]:	QStackedLayout::minimumSize() const

vfunc[17]:	QLayout::maximumSize() const
vfunc[18]:	QStackedLayout::setGeometry(QRect const&)
vfunc[19]:	QStackedLayout::itemAt(int) const
vfunc[20]:	QStackedLayout::takeAt(int)
vfunc[21]:	QLayout::indexOf(QWidget*) const
vfunc[22]:	QStackedLayout::count() const
vfunc[23]:	QLayout::isEmpty() const
vfunc[24]:	QLayout::layout()
vfunc[25]:	QStackedLayout::sizeHint() const

The Run Time Type Information for the QStackedLayout class is described by Table 18-344

Table 18-344 typeinfo for QStackedLayout

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStackedLayout
basetype:	typeinfo for QLayout

18.5.15.10 Interfaces for Qt4 Layouts

An LSB conforming implementation shall provide the generic functions for Qt4 Layouts specified in Table 18-345, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-345 libQtGui - Qt4 Layouts Function Interfaces

_ZN10QBoxLayout10addSpacingEi [QtGui]	_ZN10QBoxLayout10addStretchEi [QtGui]
_ZN10QBoxLayout10insertItemEiP11QLayoutItem [QtGui]	_ZN10QBoxLayout10invalidateEv [QtGui]
_ZN10QBoxLayout11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN10QBoxLayout11qt_metacastEPKc [QtGui]
_ZN10QBoxLayout11setGeometryERK5QRect [QtGui]	_ZN10QBoxLayout12insertLayoutEiP7QLayouti [QtGui]
_ZN10QBoxLayout12insertWidgetEiP7QWidgeti6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN10QBoxLayout12setDirectionENS_9DirectionE [QtGui]
_ZN10QBoxLayout13insertSpacingEii [QtGui]	_ZN10QBoxLayout13insertStretchEii [QtGui]
_ZN10QBoxLayout16setStretchFactorEP7QLayouti [QtGui]	_ZN10QBoxLayout16setStretchFactorEP7QWidgeti [QtGui]

_ZN10QBoxLayout6takeAtEi [QtGui]	_ZN10QBoxLayout7addItemEP11QLayoutItem [QtGui]
_ZN10QBoxLayout8addStrutEi [QtGui]	_ZN10QBoxLayout9addLayoutEP7QLayouti [QtGui]
_ZN10QBoxLayout9addWidgetEP7QWidgeti6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN10QBoxLayoutC1ENS_9DirectionEP7QWidget [QtGui]
_ZN10QBoxLayoutC1ENS_9DirectionEiPKc [QtGui]	_ZN10QBoxLayoutC1EP7QLayoutNS_9DirectionEiPKc [QtGui]
_ZN10QBoxLayoutC1EP7QWidgetNS_9DirectionEiPKc [QtGui]	_ZN10QBoxLayoutC2ENS_9DirectionEP7QWidget [QtGui]
_ZN10QBoxLayoutC2ENS_9DirectionEiPKc [QtGui]	_ZN10QBoxLayoutC2EP7QLayoutNS_9DirectionEiPKc [QtGui]
_ZN10QBoxLayoutC2EP7QWidgetNS_9DirectionEiPKc [QtGui]	_ZN10QBoxLayoutD0Ev [QtGui]
_ZN10QBoxLayoutD1Ev [QtGui]	_ZN10QBoxLayoutD2Ev [QtGui]
ZN11QGridLayout10findWidgetEP7QWidgetPiS2 [QtGui]	_ZN11QGridLayout10invalidateEv [QtGui]
_ZN11QGridLayout11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QGridLayout11qt_metacastEPKc [QtGui]
_ZN11QGridLayout11setGeometryERK5QRect [QtGui]	_ZN11QGridLayout13setRowStretchEii [QtGui]
_ZN11QGridLayout15getItemPositionEiPiS0_S0_S0_ [QtGui]	_ZN11QGridLayout15setOriginCornerEN2Qt6CornerE [QtGui]
_ZN11QGridLayout16setColumnStretchEii [QtGui]	_ZN11QGridLayout19setRowMinimumHeightEii [QtGui]
_ZN11QGridLayout21setColumnMinimumWidthEii [QtGui]	_ZN11QGridLayout21setDefaultPositioningEiN2Qt11OrientationE [LSB]
_ZN11QGridLayout6expandEii [QtGui]	_ZN11QGridLayout6takeAtEi [QtGui]
_ZN11QGridLayout7addItemEP11QLayoutItem [QtGui]	_ZN11QGridLayout7addItemEP11QLayoutItemiiii6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN11QGridLayout9addLayoutEP7QLayoutiii6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN11QGridLayout9addLayoutEP7QLayoutiiii6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN11QGridLayout9addWidgetEP7QWidgetiii6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN11QGridLayout9addWidgetEP7QWidgetiiii6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN11QGridLayoutC1EP7QLayoutiPKc [QtGui]	_ZN11QGridLayoutC1EP7QWidget [QtGui]
_ZN11QGridLayoutC1EP7QWidgetiiPKc [QtGui]	_ZN11QGridLayoutC1EiiiPKc [QtGui]

_ZN11QGridLayoutC1Ev [QtGui]	_ZN11QGridLayoutC2EP7QLayoutiiPKc [QtGui]
_ZN11QGridLayoutC2EP7QWidget [QtGui]	_ZN11QGridLayoutC2EP7QWidgetiiPKc [QtGui]
_ZN11QGridLayoutC2EiiiPKc [QtGui]	_ZN11QGridLayoutC2Ev [QtGui]
_ZN11QGridLayoutD0Ev [QtGui]	_ZN11QGridLayoutD1Ev [QtGui]
_ZN11QGridLayoutD2Ev [QtGui]	_ZN11QHBoxLayout11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN11QHBoxLayout11qt_metacastEPKc [QtGui]	_ZN11QHBoxLayoutC1EP7QLayoutiiPKc [QtGui]
_ZN11QHBoxLayoutC1EP7QWidget [QtGui]	_ZN11QHBoxLayoutC1EP7QWidgetiiPKc [QtGui]
_ZN11QHBoxLayoutC1EiPKc [QtGui]	_ZN11QHBoxLayoutC1Ev [QtGui]
_ZN11QHBoxLayoutC2EP7QLayoutiiPKc [QtGui]	_ZN11QHBoxLayoutC2EP7QWidget [QtGui]
_ZN11QHBoxLayoutC2EP7QWidgetiiPKc [QtGui]	_ZN11QHBoxLayoutC2EiPKc [QtGui]
_ZN11QHBoxLayoutC2Ev [QtGui]	_ZN11QHBoxLayoutD0Ev [QtGui]
_ZN11QHBoxLayoutD1Ev [QtGui]	_ZN11QHBoxLayoutD2Ev [QtGui]
_ZN11QLayoutItem10invalidateEv [QtGui]	_ZN11QLayoutItem10spacerItemEv [QtGui]
_ZN11QLayoutItem12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]	_ZN11QLayoutItem6layoutEv [QtGui]
_ZN11QLayoutItem6widgetEv [QtGui]	_ZN11QLayoutItemD0Ev [QtGui]
_ZN11QLayoutItemD1Ev [QtGui]	_ZN11QLayoutItemD2Ev [QtGui]
ZN11QSpacerItem10changeSizeEiiN11QSizePolicy6PolicyES1 [QtGui]	_ZN11QSpacerItem10spacerItemEv [QtGui]
_ZN11QSpacerItem11setGeometryERK5QRect [QtGui]	_ZN11QVBoxLayout11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN11QVBoxLayout11qt_metacastEPKc [QtGui]	_ZN11QVBoxLayoutC1EP7QLayoutiiPKc [QtGui]
_ZN11QVBoxLayoutC1EP7QWidget [QtGui]	_ZN11QVBoxLayoutC1EP7QWidgetiiPKc [QtGui]
_ZN11QVBoxLayoutC1EiPKc [QtGui]	_ZN11QVBoxLayoutC1Ev [QtGui]
_ZN11QVBoxLayoutC2EP7QLayoutiiPKc [QtGui]	_ZN11QVBoxLayoutC2EP7QWidget [QtGui]

_ZN11QVBoxLayoutC2EP7QWidg etIPKc [QtGui]	_ZN11QVBoxLayoutC2EiPKc [QtGui]
_ZN11QVBoxLayoutC2Ev [QtGui]	_ZN11QVBoxLayoutD0Ev [QtGui]
_ZN11QVBoxLayoutD1Ev [QtGui]	_ZN11QVBoxLayoutD2Ev [QtGui]
_ZN11QWidgetItem11setGeometryE RK5QRect [QtGui]	_ZN11QWidgetItem6widgetEv [QtGui]
_ZN14QStackedLayout11qt_metacall EN11QMetaObject4CallEiPPv [QtGui]	_ZN14QStackedLayout11qt_metacast EPKc [QtGui]
_ZN14QStackedLayout11setGeometr yERK5QRect [QtGui]	_ZN14QStackedLayout12insertWidg etEiP7QWidget [QtGui]
_ZN14QStackedLayout13widgetRem ovedEi [QtGui]	_ZN14QStackedLayout14currentCha ngedEi [QtGui]
_ZN14QStackedLayout15setCurrentI ndexEi [QtGui]	_ZN14QStackedLayout16setCurrent WidgetEP7QWidget [QtGui]
_ZN14QStackedLayout6takeAtEi [QtGui]	_ZN14QStackedLayout7addItemEP1 1QLayoutItem [QtGui]
_ZN14QStackedLayout9addWidgetE P7QWidget [QtGui]	_ZN14QStackedLayoutC1EP7QLayo ut [QtGui]
_ZN14QStackedLayoutC1EP7QWidg et [QtGui]	_ZN14QStackedLayoutC1Ev [QtGui]
_ZN14QStackedLayoutC2EP7QLayo ut [QtGui]	_ZN14QStackedLayoutC2EP7QWidg et [QtGui]
_ZN14QStackedLayoutC2Ev [QtGui]	_ZN14QStackedLayoutD0Ev [QtGui]
_ZN14QStackedLayoutD1Ev [QtGui]	_ZN14QStackedLayoutD2Ev [QtGui]
_ZN7QLayout10childEventEP11QCh ildEvent [QtGui]	_ZN7QLayout10invalidateEv [QtGui]
_ZN7QLayout10removeItemEP11QL ayoutItem [QtGui]	_ZN7QLayout10setAutoAddEb [QtGui]
_ZN7QLayout10setEnabledEb [QtGui]	_ZN7QLayout10setMenuBarEP7QWi dget [QtGui]
_ZN7QLayout10setSpacingEi [QtGui]	_ZN7QLayout11qt_metacallEN11Q MetaObject4CallEiPPv [QtGui]
_ZN7QLayout11qt_metacastEPKc [QtGui]	_ZN7QLayout11setGeometryERK5Q Rect [QtGui]
_ZN7QLayout11widgetEventEP6QE vent [LSB]	_ZN7QLayout12removeWidgetEP7Q Widget [QtGui]
_ZN7QLayout12setAlignmentEP7Q Widget6QFlagsIN2Qt13AlignmentFl agEE [QtGui]	_ZN7QLayout12setAlignmentEPS_6 QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN7QLayout14addChildLayoutEPS _ [QtGui]	_ZN7QLayout14addChildWidgetEP7 QWidget [QtGui]

_ZN7QLayout14deleteAllItemsEv [QtGui]	_ZN7QLayout17setSizeConstraintENS_14SizeConstraintE [QtGui]
_ZN7QLayout21closestAcceptableSizeEPK7QWidgetRK5QSize [QtGui]	_ZN7QLayout6freezeEii [QtGui]
_ZN7QLayout6layoutEv [QtGui]	_ZN7QLayout6updateEv [QtGui]
_ZN7QLayout8activateEv [QtGui]	_ZN7QLayout9addWidgetEP7QWidget [QtGui]
_ZN7QLayout9setMarginEi [QtGui]	_ZN7QLayoutC1EP7QWidget [QtGui]
_ZN7QLayoutC1EP7QWidgetiiPKc [QtGui]	_ZN7QLayoutC1EPS_iPKc [QtGui]
_ZN7QLayoutC1EiPKc [QtGui]	_ZN7QLayoutC1Ev [QtGui]
_ZN7QLayoutC2EP7QWidget [QtGui]	_ZN7QLayoutC2EP7QWidgetiiPKc [QtGui]
_ZN7QLayoutC2EPS_iPKc [QtGui]	_ZN7QLayoutC2EiPKc [QtGui]
_ZN7QLayoutC2Ev [QtGui]	_ZN7QLayoutD0Ev [QtGui]
_ZN7QLayoutD1Ev [QtGui]	_ZN7QLayoutD2Ev [QtGui]
_ZNK10QBoxLayout10metaObjectEv [QtGui]	_ZNK10QBoxLayout11maximumSizeEv [QtGui]
_ZNK10QBoxLayout11minimumSizeEv [QtGui]	_ZNK10QBoxLayout14heightForWidthEi [QtGui]
_ZNK10QBoxLayout17hasHeightForWidthEv [QtGui]	_ZNK10QBoxLayout19expandingDirectionsEv [QtGui]
_ZNK10QBoxLayout21minimumHeightForWidthEi [QtGui]	_ZNK10QBoxLayout5countEv [QtGui]
_ZNK10QBoxLayout6itemAtEi [QtGui]	_ZNK10QBoxLayout8sizeHintEv [QtGui]
_ZNK10QBoxLayout9directionEv [QtGui]	_ZNK11QGridLayout10metaObjectEv [QtGui]
_ZNK11QGridLayout10rowStretchEi [QtGui]	_ZNK11QGridLayout11columnCountEv [QtGui]
_ZNK11QGridLayout11maximumSizeEv [QtGui]	_ZNK11QGridLayout11minimumSizeEv [QtGui]
_ZNK11QGridLayout12originCornerEv [QtGui]	_ZNK11QGridLayout13columnStretchEi [QtGui]
_ZNK11QGridLayout14heightForWidthEi [QtGui]	_ZNK11QGridLayout16rowMinimumHeightEi [QtGui]
_ZNK11QGridLayout17hasHeightForWidthEv [QtGui]	_ZNK11QGridLayout18columnMinimumWidthEi [QtGui]
_ZNK11QGridLayout19expandingDirectionsEv [QtGui]	_ZNK11QGridLayout21minimumHeightForWidthEi [QtGui]

_Znk11QGridLayout5countEv [QtGui]	_Znk11QGridLayout6itemAtEi [QtGui]
_Znk11QGridLayout8cellRectEii [QtGui]	_Znk11QGridLayout8rowCountEv [QtGui]
_Znk11QGridLayout8sizeHintEv [QtGui]	_Znk11QHBoxLayout10metaObject Ev [QtGui]
_Znk11QLayoutItem14heightForWidthEi [QtGui]	_Znk11QLayoutItem17hasHeightForWidthEv [QtGui]
_Znk11QLayoutItem21minimumHeightForWidthEi [QtGui]	_Znk11QSizePolicycv8QVariantEv [QtGui]
_Znk11QSpacerItem11maximumSizeEv [QtGui]	_Znk11QSpacerItem11minimumSizeEv [QtGui]
_Znk11QSpacerItem19expandingDirectionsEv [QtGui]	_Znk11QSpacerItem7isEmptyEv [QtGui]
_Znk11QSpacerItem8geometryEv [QtGui]	_Znk11QSpacerItem8sizeHintEv [QtGui]
_Znk11QVBoxLayout10metaObject Ev [QtGui]	_Znk11QWidgetItem11maximumSizeEv [QtGui]
_Znk11QWidgetItem11minimumSizeEv [QtGui]	_Znk11QWidgetItem14heightForWidthEi [QtGui]
_Znk11QWidgetItem17hasHeightForWidthEv [QtGui]	_Znk11QWidgetItem19expandingDirectionsEv [QtGui]
_Znk11QWidgetItem7isEmptyEv [QtGui]	_Znk11QWidgetItem8geometryEv [QtGui]
_Znk11QWidgetItem8sizeHintEv [QtGui]	_Znk14QStackedLayout10metaObjectEv [QtGui]
_Znk14QStackedLayout11minimumSizeEv [QtGui]	_Znk14QStackedLayout12currentIndexEv [QtGui]
_Znk14QStackedLayout13currentWidgetEv [QtGui]	_Znk14QStackedLayout5countEv [QtGui]
_Znk14QStackedLayout6itemAtEi [QtGui]	_Znk14QStackedLayout6widgetEi [QtGui]
_Znk14QStackedLayout8sizeHintEv [QtGui]	_Znk7QLayout10isTopLevelEv [QtGui]
_Znk7QLayout10metaObjectEv [QtGui]	_Znk7QLayout11maximumSizeEv [QtGui]
_Znk7QLayout11minimumSizeEv [QtGui]	_Znk7QLayout12parentWidgetEv [QtGui]
_Znk7QLayout13alignmentRectERK5QRect [QtGui]	_Znk7QLayout13totalSizeHintEv [LSB]
_Znk7QLayout14sizeConstraintEv [QtGui]	_Znk7QLayout16totalMaximumSizeEv [LSB]

_Znk7QLayout16totalMinimumSizeEv [LSB]	_Znk7QLayout19expandingDirectionsEv [QtGui]
_Znk7QLayout19totalHeightForWidthEi [LSB]	_Znk7QLayout6marginEv [QtGui]
_Znk7QLayout7autoAddEv [QtGui]	_Znk7QLayout7indexOfEP7QWidget [QtGui]
_Znk7QLayout7isEmptyEv [QtGui]	_Znk7QLayout7menuBarEv [QtGui]
_Znk7QLayout7spacingEv [QtGui]	_Znk7QLayout8geometryEv [QtGui]
_Znk7QLayout9isEnabledEv [QtGui]	_ZlsR11QDataStreamRK11QSizePolicy [QtXml]
_ZrsR11QDataStreamRK11QSizePolicy [QtXml]	

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Layouts specified in Table 18-346, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-346 libQtGui - Qt4 Layouts Deprecated Function Interfaces

_Zn7QLayout9setMarginEi [QtGui]	

18.5.16 Qt4 Style

18.5.16.1 Class data for QStyle

The virtual table for the QStyle class is described by Table 18-347

Table 18-347 Primary vtable for QStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStyle
vfunc[0]:	QStyle::metaObject() const
vfunc[1]:	QStyle::qt_metacast(char const*)
vfunc[2]:	QStyle::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QStyle::~QStyle()
vfunc[4]:	QStyle::~QStyle()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QStyle::polish(QWidget*)
vfunc[13]:	QStyle::unpolish(QWidget*)
vfunc[14]:	QStyle::polish(QApplication*)
vfunc[15]:	QStyle::unpolish(QApplication*)
vfunc[16]:	QStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const
vfunc[19]:	QStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QStyle::standardPalette() const
vfunc[22]:	__cxa_pure_virtual
vfunc[23]:	__cxa_pure_virtual
vfunc[24]:	__cxa_pure_virtual
vfunc[25]:	__cxa_pure_virtual
vfunc[26]:	__cxa_pure_virtual
vfunc[27]:	__cxa_pure_virtual
vfunc[28]:	__cxa_pure_virtual
vfunc[29]:	__cxa_pure_virtual
vfunc[30]:	__cxa_pure_virtual
vfunc[31]:	__cxa_pure_virtual
vfunc[32]:	__cxa_pure_virtual

The Run Time Type Information for the QStyle class is described by Table 18-348

Table 18-348 typeinfo for QStyle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStyle
basetype:	typeinfo for QObject

18.5.16.2 Class data for QCommonStyle

The virtual table for the QCommonStyle class is described by Table 18-349

Table 18-349 Primary vtable for QCommonStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCommonStyle
vfunc[0]:	QCommonStyle::metaObject() const
vfunc[1]:	QCommonStyle::qt_metacast(char const*)
vfunc[2]:	QCommonStyle::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QCommonStyle::~~QCommonStyle()
vfunc[4]:	QCommonStyle::~~QCommonStyle()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QStyle::polish(QWidget*)
vfunc[13]:	QStyle::unpolish(QWidget*)
vfunc[14]:	QStyle::polish(QApplication*)
vfunc[15]:	QStyle::unpolish(QApplication*)
vfunc[16]:	QStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const

vfunc[19]:	QStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QStyle::standardPalette() const
vfunc[22]:	QCommonStyle::drawPrimitive(QStyle ::PrimitiveElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[23]:	QCommonStyle::drawControl(QStyle ::ControlElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[24]:	QCommonStyle::subElementRect(QS tyle::SubElement, QStyleOption const*, QWidget const*) const
vfunc[25]:	QCommonStyle::drawComplexContr ol(QStyle::ComplexControl, QStyleOptionComplex const*, QPainter*, QWidget const*) const
vfunc[26]:	QCommonStyle::hitTestComplexCon trol(QStyle::ComplexControl, QStyleOptionComplex const*, QPoint const&, QWidget const*) const
vfunc[27]:	QCommonStyle::subControlRect(QSt yle::ComplexControl, QStyleOptionComplex const*, QStyle::SubControl, QWidget const*) const
vfunc[28]:	QCommonStyle::pixelMetric(QStyle:: PixelMetric, QStyleOption const*, QWidget const*) const
vfunc[29]:	QCommonStyle::sizeFromContents(QStyle::ContentsType, QStyleOption const*, QSize const&, QWidget const*) const
vfunc[30]:	QCommonStyle::styleHint(QStyle::St yleHint, QStyleOption const*, QWidget const*, QStyleHintReturn*) const
vfunc[31]:	QCommonStyle::standardPixmap(QS tyle::StandardPixmap, QStyleOption const*, QWidget const*) const

vfunc[32]:	QCommonStyle::generatedIconPixmap(QIcon::Mode, QPixmap const&, QStyleOption const*) const

The Run Time Type Information for the QCommonStyle class is described by Table 18-350

Table 18-350 typeinfo for QCommonStyle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QCommonStyle
basetype:	typeinfo for QStyle

18.5.16.3 Class data for QMotifStyle

The virtual table for the QMotifStyle class is described by Table 18-351

Table 18-351 Primary vtable for QMotifStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMotifStyle
vfunc[0]:	QMotifStyle::metaObject() const
vfunc[1]:	QMotifStyle::qt_metacast(char const*)
vfunc[2]:	QMotifStyle::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMotifStyle::~QMotifStyle()
vfunc[4]:	QMotifStyle::~QMotifStyle()
vfunc[5]:	QMotifStyle::event(QEvent*)
vfunc[6]:	QMotifStyle::eventFilter(QObject*, QEvent*)
vfunc[7]:	QMotifStyle::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QMotifStyle::polish(QWidget*)
vfunc[13]:	QMotifStyle::unpolish(QWidget*)
vfunc[14]:	QMotifStyle::polish(QApplication*)

vfunc[15]:	QMotifStyle::unpolish(QApplication*)
vfunc[16]:	QMotifStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const
vfunc[19]:	QStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QMotifStyle::standardPalette() const
vfunc[22]:	QMotifStyle::drawPrimitive(QStyle::PrimitiveElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[23]:	QMotifStyle::drawControl(QStyle::ControlElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[24]:	QMotifStyle::subElementRect(QStyle::SubElement, QStyleOption const*, QWidget const*) const
vfunc[25]:	QMotifStyle::drawComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPainter*, QWidget const*) const
vfunc[26]:	QCommonStyle::hitTestComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPoint const&, QWidget const*) const
vfunc[27]:	QMotifStyle::subControlRect(QStyle::ComplexControl, QStyleOptionComplex const*, QStyle::SubControl, QWidget const*) const
vfunc[28]:	QMotifStyle::pixelMetric(QStyle::PixelMetric, QStyleOption const*, QWidget const*) const
vfunc[29]:	QMotifStyle::sizeFromContents(QStyle::ContentsType, QStyleOption

	const*, QSize const&, QWidget const*) const
vfunc[30]:	QMotifStyle::styleHint(QStyle::Style Hint, QStyleOption const*, QWidget const*, QStyleHintReturn*) const
vfunc[31]:	QMotifStyle::standardPixmap(QStyle::StandardPixmap, QStyleOption const*, QWidget const*) const
vfunc[32]:	QCommonStyle::generatedIconPixmap(QIcon::Mode, QPixmap const&, QStyleOption const*) const

The Run Time Type Information for the QMotifStyle class is described by Table 18-352

Table 18-352 typeinfo for QMotifStyle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMotifStyle
basetype:	typeinfo for QCommonStyle

18.5.16.4 Class data for QStyleFactoryInterface

The virtual table for the QStyleFactoryInterface class is described by Table 18-353

Table 18-353 Primary vtable for QStyleFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStyleFactoryInterface
vfunc[0]:	NULL or QStyleFactoryInterface::~~QStyleFactoryInterface()
vfunc[1]:	NULL or QStyleFactoryInterface::~~QStyleFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QStyleFactoryInterface class is described by Table 18-354

Table 18-354 typeinfo for QStyleFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStyleFactoryInterface

basetype:	typeinfo for QFactoryInterface

18.5.16.5 Class data for QWindowsStyle

The virtual table for the QWindowsStyle class is described by Table 18-355

Table 18-355 Primary vtable for QWindowsStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWindowsStyle
vfunc[0]:	QWindowsStyle::metaObject() const
vfunc[1]:	QWindowsStyle::qt_metacast(char const*)
vfunc[2]:	QWindowsStyle::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QWindowsStyle::~~QWindowsStyle()
vfunc[4]:	QWindowsStyle::~QWindowsStyle()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QWindowsStyle::eventFilter(QObject*, QEvent*)
vfunc[7]:	QWindowsStyle::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWindowsStyle::polish(QWidget*)
vfunc[13]:	QWindowsStyle::unpolish(QWidget*)
vfunc[14]:	QWindowsStyle::polish(QApplication*)
vfunc[15]:	QWindowsStyle::unpolish(QApplication*)
vfunc[16]:	QWindowsStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const

vfunc[19]:	QStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QStyle::standardPalette() const
vfunc[22]:	QWindowsStyle::drawPrimitive(QSt yle::PrimitiveElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[23]:	QWindowsStyle::drawControl(QStyl e::ControlElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[24]:	QWindowsStyle::subElementRect(QS tyle::SubElement, QStyleOption const*, QWidget const*) const
vfunc[25]:	QWindowsStyle::drawComplexCont rol(QStyle::ComplexControl, QStyleOptionComplex const*, QPainter*, QWidget const*) const
vfunc[26]:	QCommonStyle::hitTestComplexCon trol(QStyle::ComplexControl, QStyleOptionComplex const*, QPoint const&, QWidget const*) const
vfunc[27]:	QCommonStyle::subControlRect(QSt yle::ComplexControl, QStyleOptionComplex const*, QStyle::SubControl, QWidget const*) const
vfunc[28]:	QWindowsStyle::pixelMetric(QStyle:: PixelMetric, QStyleOption const*, QWidget const*) const
vfunc[29]:	QWindowsStyle::sizeFromContents(QStyle::ContentsType, QStyleOption const*, QSize const&, QWidget const*) const
vfunc[30]:	QWindowsStyle::styleHint(QStyle::St yleHint, QStyleOption const*, QWidget const*, QStyleHintReturn*) const
vfunc[31]:	QWindowsStyle::standardPixmap(Q Style::StandardPixmap,

	QStyleOption const*, QWidget const*) const
vfunc[32]:	QCommonStyle::generatedIconPixmap(QIcon::Mode, QPixmap const&, QStyleOption const*) const

The Run Time Type Information for the QWindowsStyle class is described by Table 18-356

Table 18-356 typeinfo for QWindowsStyle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWindowsStyle
basetype:	typeinfo for QCommonStyle

18.5.16.6 Class data for QCDEStyle

The virtual table for the QCDEStyle class is described by Table 18-357

Table 18-357 Primary vtable for QCDEStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCDEStyle
vfunc[0]:	QCDEStyle::metaObject() const
vfunc[1]:	QCDEStyle::qt_metacast(char const*)
vfunc[2]:	QCDEStyle::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QCDEStyle::~~QCDEStyle()
vfunc[4]:	QCDEStyle::~~QCDEStyle()
vfunc[5]:	QMotifStyle::event(QEvent*)
vfunc[6]:	QMotifStyle::eventFilter(QObject*, QEvent*)
vfunc[7]:	QMotifStyle::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QMotifStyle::polish(QWidget*)
vfunc[13]:	QMotifStyle::unpolish(QWidget*)
vfunc[14]:	QMotifStyle::polish(QApplication*)

vfunc[15]:	QMotifStyle::unpolish(QApplication*)
vfunc[16]:	QMotifStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const
vfunc[19]:	QStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QCDEStyle::standardPalette() const
vfunc[22]:	QCDEStyle::drawPrimitive(QStyle::PrimitiveElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[23]:	QCDEStyle::drawControl(QStyle::ControlElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[24]:	QMotifStyle::subElementRect(QStyle::SubElement, QStyleOption const*, QWidget const*) const
vfunc[25]:	QMotifStyle::drawComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPainter*, QWidget const*) const
vfunc[26]:	QCommonStyle::hitTestComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPoint const&, QWidget const*) const
vfunc[27]:	QMotifStyle::subControlRect(QStyle::ComplexControl, QStyleOptionComplex const*, QStyle::SubControl, QWidget const*) const
vfunc[28]:	QCDEStyle::pixelMetric(QStyle::PixelMetric, QStyleOption const*, QWidget const*) const
vfunc[29]:	QMotifStyle::sizeFromContents(QStyle::ContentsType, QStyleOption

	const*, QSize const&, QWidget const*) const
vfunc[30]:	QMotifStyle::styleHint(QStyle::Style Hint, QStyleOption const*, QWidget const*, QStyleHintReturn*) const
vfunc[31]:	QMotifStyle::standardPixmap(QStyle::StandardPixmap, QStyleOption const*, QWidget const*) const
vfunc[32]:	QCommonStyle::generatedIconPixmap(QIcon::Mode, QPixmap const&, QStyleOption const*) const

The Run Time Type Information for the QCDEStyle class is described by Table 18-358

Table 18-358 typeinfo for QCDEStyle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QCDEStyle
basetype:	typeinfo for QMotifStyle

18.5.16.7 Class data for QPlastiqueStyle

The virtual table for the QPlastiqueStyle class is described by Table 18-359

Table 18-359 Primary vtable for QPlastiqueStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPlastiqueStyle
vfunc[0]:	QPlastiqueStyle::metaObject() const
vfunc[1]:	QPlastiqueStyle::qt_metacast(char const*)
vfunc[2]:	QPlastiqueStyle::qt_metacall(QMeta Object::Call, int, void**)
vfunc[3]:	QPlastiqueStyle::~~QPlastiqueStyle()
vfunc[4]:	QPlastiqueStyle::~~QPlastiqueStyle()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QPlastiqueStyle::eventFilter(QObject*, QEvent*)
vfunc[7]:	QPlastiqueStyle::timerEvent(QTimer Event*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)

vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QPlastiqueStyle::polish(QWidget*)
vfunc[13]:	QPlastiqueStyle::unpolish(QWidget*)
vfunc[14]:	QPlastiqueStyle::polish(QApplication*)
vfunc[15]:	QPlastiqueStyle::unpolish(QApplication*)
vfunc[16]:	QPlastiqueStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const
vfunc[19]:	QStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QPlastiqueStyle::standardPalette() const
vfunc[22]:	QPlastiqueStyle::drawPrimitive(QStyle::PrimitiveElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[23]:	QPlastiqueStyle::drawControl(QStyle::ControlElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[24]:	QPlastiqueStyle::subElementRect(QStyle::SubElement, QStyleOption const*, QWidget const*) const
vfunc[25]:	QPlastiqueStyle::drawComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPainter*, QWidget const*) const
vfunc[26]:	QPlastiqueStyle::hitTestComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPoint const&, QWidget const*) const

vfunc[27]:	QPlastiqueStyle::subControlRect(QStyle::ComplexControl, QStyleOptionComplex const*, QStyle::SubControl, QWidget const*) const
vfunc[28]:	QPlastiqueStyle::pixelMetric(QStyle::PixelMetric, QStyleOption const*, QWidget const*) const
vfunc[29]:	QPlastiqueStyle::sizeFromContents(QStyle::ContentsType, QStyleOption const*, QSize const&, QWidget const*) const
vfunc[30]:	QPlastiqueStyle::styleHint(QStyle::StyleHint, QStyleOption const*, QWidget const*, QStyleHintReturn*) const
vfunc[31]:	QWindowsStyle::standardPixmap(QStyle::StandardPixmap, QStyleOption const*, QWidget const*) const
vfunc[32]:	QCommonStyle::generatedIconPixmap(QIcon::Mode, QPixmap const&, QStyleOption const*) const

The Run Time Type Information for the QPlastiqueStyle class is described by Table 18-360

Table 18-360 typeinfo for QPlastiqueStyle

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPlastiqueStyle
basetype:	typeinfo for QWindowsStyle

18.5.16.8 Class data for QCleanlooksStyle

The virtual table for the QCleanlooksStyle class is described by Table 18-361

Table 18-361 Primary vtable for QCleanlooksStyle

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCleanlooksStyle
vfunc[0]:	QCleanlooksStyle::metaObject() const
vfunc[1]:	QCleanlooksStyle::qt_metacast(char const*)
vfunc[2]:	QCleanlooksStyle::qt_metacall(QMetaObject::Call, int, void**)

vfunc[3]:	QCleanlooksStyle::~~QCleanlooksStyle()
vfunc[4]:	QCleanlooksStyle::~~QCleanlooksStyle()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QWindowsStyle::eventFilter(QObject*, QEvent*)
vfunc[7]:	QWindowsStyle::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QCleanlooksStyle::polish(QWidget*)
vfunc[13]:	QCleanlooksStyle::unpolish(QWidget*)
vfunc[14]:	QCleanlooksStyle::polish(QApplication*)
vfunc[15]:	QCleanlooksStyle::unpolish(QApplication*)
vfunc[16]:	QCleanlooksStyle::polish(QPalette&)
vfunc[17]:	QStyle::itemTextRect(QFontMetrics const&, QRect const&, int, bool, QString const&) const
vfunc[18]:	QCleanlooksStyle::itemPixmapRect(QRect const&, int, QPixmap const&) const
vfunc[19]:	QCleanlooksStyle::drawItemText(QPainter*, QRect const&, int, QPalette const&, bool, QString const&, QPalette::ColorRole) const
vfunc[20]:	QCleanlooksStyle::drawItemPixmap(QPainter*, QRect const&, int, QPixmap const&) const
vfunc[21]:	QCleanlooksStyle::standardPalette() const
vfunc[22]:	QCleanlooksStyle::drawPrimitive(QStyle::PrimitiveElement, QStyleOption const*, QPainter*, QWidget const*) const
vfunc[23]:	QCleanlooksStyle::drawControl(QStyle::ControlElement, QStyleOption

	const*, QPainter*, QWidget const*) const
vfunc[24]:	QCleanlooksStyle::subElementRect(QStyle::SubElement, QStyleOption const*, QWidget const*) const
vfunc[25]:	QCleanlooksStyle::drawComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPainter*, QWidget const*) const
vfunc[26]:	QCleanlooksStyle::hitTestComplexControl(QStyle::ComplexControl, QStyleOptionComplex const*, QPoint const&, QWidget const*) const
vfunc[27]:	QCleanlooksStyle::subControlRect(QStyle::ComplexControl, QStyleOptionComplex const*, QStyle::SubControl, QWidget const*) const
vfunc[28]:	QCleanlooksStyle::pixelMetric(QStyle::PixelMetric, QStyleOption const*, QWidget const*) const
vfunc[29]:	QCleanlooksStyle::sizeFromContents(QStyle::ContentsType, QStyleOption const*, QSize const&, QWidget const*) const
vfunc[30]:	QCleanlooksStyle::styleHint(QStyle::StyleHint, QStyleOption const*, QWidget const*, QStyleHintReturn*) const
vfunc[31]:	QCleanlooksStyle::standardPixmap(QStyle::StandardPixmap, QStyleOption const*, QWidget const*) const
vfunc[32]:	QCleanlooksStyle::generatedIconPixmap(QIcon::Mode, QPixmap const&, QStyleOption const*) const

18.5.16.9 Interfaces for Qt4 Style

An LSB conforming implementation shall provide the generic functions for Qt4 Style specified in Table 18-362, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-362 libQtGui - Qt4 Style Function Interfaces

_ZN11QMotifStyle10timerEventEP11QTimerEvent [QtGui]	_ZN11QMotifStyle11eventFilterEP7QObjectP6QEvent [QtGui]

_ZN11QMotifStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QMotifStyle11qt_metacastEPKc [QtGui]
_ZN11QMotifStyle21setUseHighlightColorsEb [QtGui]	_ZN11QMotifStyle5eventEP6QEvent [QtGui]
_ZN11QMotifStyle6polishEP12QApplication [QtGui]	_ZN11QMotifStyle6polishEP7QWidget [QtGui]
_ZN11QMotifStyle6polishER8QPalette [QtGui]	_ZN11QMotifStyle8unpolishEP12QApplication [QtGui]
_ZN11QMotifStyle8unpolishEP7QWidget [QtGui]	_ZN11QMotifStyleC1Eb [QtGui]
_ZN11QMotifStyleC2Eb [QtGui]	_ZN11QMotifStyleD0Ev [QtGui]
_ZN11QMotifStyleD1Ev [QtGui]	_ZN11QMotifStyleD2Ev [QtGui]
_ZN12QCommonStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN12QCommonStyle11qt_metacastEPKc [QtGui]
_ZN12QCommonStyleC1Ev [QtGui]	_ZN12QCommonStyleC2Ev [QtGui]
_ZN12QCommonStyleD0Ev [QtGui]	_ZN12QCommonStyleD1Ev [QtGui]
_ZN12QCommonStyleD2Ev [QtGui]	_ZN12QStyleOption4initEPK7QWidget [QtGui]
ZN12QStyleOptionC1ERKS [QtGui]	_ZN12QStyleOptionC1Eii [QtGui]
ZN12QStyleOptionC2ERKS [QtGui]	_ZN12QStyleOptionC2Eii [QtGui]
_ZN12QStyleOptionD1Ev [QtGui]	_ZN12QStyleOptionD2Ev [QtGui]
ZN12QStyleOptionaSERKS [QtGui]	_ZN13QStyleFactory4keysEv [QtGui]
_ZN13QStyleFactory6createERK7QStyle [QtGui]	_ZN13QWindowsStyle10timerEventEP11QTimerEvent [QtGui]
_ZN13QWindowsStyle11eventFilterEP7QObjectP6QEvent [QtGui]	_ZN13QWindowsStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN13QWindowsStyle11qt_metacastEPKc [QtGui]	_ZN13QWindowsStyle6polishEP12QApplication [QtGui]
_ZN13QWindowsStyle6polishEP7QWidget [QtGui]	_ZN13QWindowsStyle6polishER8QPalette [QtGui]
_ZN13QWindowsStyle8unpolishEP12QApplication [QtGui]	_ZN13QWindowsStyle8unpolishEP7QWidget [QtGui]
_ZN13QWindowsStyleC1Ev [QtGui]	_ZN13QWindowsStyleC2Ev [QtGui]
_ZN13QWindowsStyleD0Ev [QtGui]	_ZN13QWindowsStyleD1Ev [QtGui]
_ZN13QWindowsStyleD2Ev [QtGui]	_ZN15QPlastiqueStyle10timerEventEP11QTimerEvent [QtGui]

_ZN15QPlastiqueStyle11eventFilterEP7QObjectP6QEvent [QtGui]	_ZN15QPlastiqueStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN15QPlastiqueStyle11qt_metacastEPKc [QtGui]	_ZN15QPlastiqueStyle6polishEP12QApplication [QtGui]
_ZN15QPlastiqueStyle6polishEP7QWidget [QtGui]	_ZN15QPlastiqueStyle6polishER8QPalette [QtGui]
_ZN15QPlastiqueStyle8unpolishEP12QApplication [QtGui]	_ZN15QPlastiqueStyle8unpolishEP7QWidget [QtGui]
_ZN15QPlastiqueStyleC1Ev [QtGui]	_ZN15QPlastiqueStyleC2Ev [QtGui]
_ZN15QPlastiqueStyleD0Ev [QtGui]	_ZN15QPlastiqueStyleD1Ev [QtGui]
_ZN15QPlastiqueStyleD2Ev [QtGui]	_ZN15QStyleOptionTabC1Ei [QtGui]
_ZN15QStyleOptionTabC1Ev [QtGui]	_ZN15QStyleOptionTabC2Ei [QtGui]
_ZN15QStyleOptionTabC2Ev [QtGui]	_ZN16QCleanlooksStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN16QCleanlooksStyle11qt_metacastEPKc [QtXml]	_ZN16QCleanlooksStyle6polishEP12QApplication [QtXml]
_ZN16QCleanlooksStyle6polishEP7QWidget [QtXml]	_ZN16QCleanlooksStyle6polishER8QPalette [QtXml]
_ZN16QCleanlooksStyle8unpolishEP12QApplication [QtXml]	_ZN16QCleanlooksStyle8unpolishEP7QWidget [QtXml]
_ZN16QCleanlooksStyleC1Ev [QtXml]	_ZN16QCleanlooksStyleC2Ev [QtXml]
_ZN16QCleanlooksStyleD0Ev [QtXml]	_ZN16QCleanlooksStyleD1Ev [QtXml]
_ZN16QCleanlooksStyleD2Ev [QtXml]	_ZN16QStyleHintReturnC1Eii [QtGui]
_ZN16QStyleHintReturnC2Eii [QtGui]	_ZN16QStyleHintReturnD1Ev [QtGui]
_ZN16QStyleHintReturnD2Ev [QtGui]	_ZN17QStyleOptionFrameC1Ei [QtGui]
_ZN17QStyleOptionFrameC1Ev [QtGui]	_ZN17QStyleOptionFrameC2Ei [QtGui]
_ZN17QStyleOptionFrameC2Ev [QtGui]	_ZN17QStyleOptionTabV2C1ERK15QStyleOptionTab [QtGui]
_ZN17QStyleOptionTabV2C1Ei [QtGui]	_ZN17QStyleOptionTabV2C1Ev [QtGui]
_ZN17QStyleOptionTabV2C2ERK15QStyleOptionTab [QtGui]	_ZN17QStyleOptionTabV2C2Ei [QtGui]

_ZN17QStyleOptionTabV2C2Ev [QtGui]	_ZN17QStyleOptionTabV2aSERK15 QStyleOptionTab [QtGui]
_ZN18QStyleOptionButtonC1Ei [QtGui]	_ZN18QStyleOptionButtonC1Ev [QtGui]
_ZN18QStyleOptionButtonC2Ei [QtGui]	_ZN18QStyleOptionButtonC2Ev [QtGui]
_ZN18QStyleOptionHeaderC1Ei [QtGui]	_ZN18QStyleOptionHeaderC1Ev [QtGui]
_ZN18QStyleOptionHeaderC2Ei [QtGui]	_ZN18QStyleOptionHeaderC2Ev [QtGui]
_ZN18QStyleOptionSliderC1Ei [QtGui]	_ZN18QStyleOptionSliderC1Ev [QtGui]
_ZN18QStyleOptionSliderC2Ei [QtGui]	_ZN18QStyleOptionSliderC2Ev [QtGui]
_ZN19QStyleOptionComplexC1Eii [QtGui]	_ZN19QStyleOptionComplexC2Eii [QtGui]
_ZN19QStyleOptionFrameV2C1ERK 17QStyleOptionFrame [QtGui]	_ZN19QStyleOptionFrameV2C1Ei [QtGui]
_ZN19QStyleOptionFrameV2C1Ev [QtGui]	_ZN19QStyleOptionFrameV2C2ERK 17QStyleOptionFrame [QtGui]
_ZN19QStyleOptionFrameV2C2Ei [QtGui]	_ZN19QStyleOptionFrameV2C2Ev [QtGui]
_ZN19QStyleOptionFrameV2aSERK1 7QStyleOptionFrame [QtGui]	_ZN19QStyleOptionSpinBoxC1Ei [QtGui]
_ZN19QStyleOptionSpinBoxC1Ev [QtGui]	_ZN19QStyleOptionSpinBoxC2Ei [QtGui]
_ZN19QStyleOptionSpinBoxC2Ev [QtGui]	_ZN19QStyleOptionToolBarC1Ei [QtGui]
_ZN19QStyleOptionToolBarC1Ev [QtGui]	_ZN19QStyleOptionToolBarC2Ei [QtGui]
_ZN19QStyleOptionToolBarC2Ev [QtGui]	_ZN19QStyleOptionToolBoxC1Ei [QtGui]
_ZN19QStyleOptionToolBoxC1Ev [QtGui]	_ZN19QStyleOptionToolBoxC2Ei [QtGui]
_ZN19QStyleOptionToolBoxC2Ev [QtGui]	_ZN20QStyleHintReturnMaskC1Ev [QtGui]
_ZN20QStyleHintReturnMaskC2Ev [QtGui]	_ZN20QStyleOptionComboBoxC1Ei [QtGui]
_ZN20QStyleOptionComboBoxC1Ev [QtGui]	_ZN20QStyleOptionComboBoxC2Ei [QtGui]
_ZN20QStyleOptionComboBoxC2Ev [QtGui]	_ZN20QStyleOptionGroupBoxC1Ei [QtGui]

_ZN20QStyleOptionGroupBoxC1Ev [QtGui]	_ZN20QStyleOptionGroupBoxC2Ei [QtGui]
_ZN20QStyleOptionGroupBoxC2Ev [QtGui]	_ZN20QStyleOptionMenuItemC1Ei [QtGui]
_ZN20QStyleOptionMenuItemC1Ev [QtGui]	_ZN20QStyleOptionMenuItemC2Ei [QtGui]
_ZN20QStyleOptionMenuItemC2Ev [QtGui]	_ZN20QStyleOptionTitleBarC1Ei [QtGui]
_ZN20QStyleOptionTitleBarC1Ev [QtGui]	_ZN20QStyleOptionTitleBarC2Ei [QtGui]
_ZN20QStyleOptionTitleBarC2Ev [QtGui]	_ZN20QStyleOptionViewItemC1Ei [QtGui]
_ZN20QStyleOptionViewItemC1Ev [QtGui]	_ZN20QStyleOptionViewItemC2Ei [QtGui]
_ZN20QStyleOptionViewItemC2Ev [QtGui]	_ZN21QStyleOptionFocusRectC1Ei [QtGui]
_ZN21QStyleOptionFocusRectC1Ev [QtGui]	_ZN21QStyleOptionFocusRectC2Ei [QtGui]
_ZN21QStyleOptionFocusRectC2Ev [QtGui]	_ZN22QStyleOptionDockWidgetC1Ei [QtGui]
_ZN22QStyleOptionDockWidgetC1Ev [QtGui]	_ZN22QStyleOptionDockWidgetC2Ei [QtGui]
_ZN22QStyleOptionDockWidgetC2Ev [QtGui]	_ZN22QStyleOptionQ3ListViewC1Ei [QtGui]
_ZN22QStyleOptionQ3ListViewC1Ev [QtGui]	_ZN22QStyleOptionQ3ListViewC2Ei [QtGui]
_ZN22QStyleOptionQ3ListViewC2Ev [QtGui]	_ZN22QStyleOptionRubberBandC1Ei [QtGui]
_ZN22QStyleOptionRubberBandC1Ev [QtGui]	_ZN22QStyleOptionRubberBandC2Ei [QtGui]
_ZN22QStyleOptionRubberBandC2Ev [QtGui]	_ZN22QStyleOptionTabBarBaseC1Ei [QtGui]
_ZN22QStyleOptionTabBarBaseC1Ev [QtGui]	_ZN22QStyleOptionTabBarBaseC2Ei [QtGui]
_ZN22QStyleOptionTabBarBaseC2Ev [QtGui]	_ZN22QStyleOptionToolButtonC1Ei [QtGui]
_ZN22QStyleOptionToolButtonC1Ev [QtGui]	_ZN22QStyleOptionToolButtonC2Ei [QtGui]
_ZN22QStyleOptionToolButtonC2Ev [QtGui]	_ZN22QStyleOptionViewItemV2C1E RK20QStyleOptionViewItem [QtXml]

_ZN22QStyleOptionViewItemV2C1Ei [QtXml]	_ZN22QStyleOptionViewItemV2C1Ev [QtXml]
_ZN22QStyleOptionViewItemV2C2ERK20QStyleOptionViewItem [QtXml]	_ZN22QStyleOptionViewItemV2C2Ei [QtXml]
_ZN22QStyleOptionViewItemV2C2Ev [QtXml]	_ZN22QStyleOptionViewItemV2aSERK20QStyleOptionViewItem [QtXml]
_ZN23QStyleOptionProgressBarC1Ei [QtGui]	_ZN23QStyleOptionProgressBarC1Ev [QtGui]
_ZN23QStyleOptionProgressBarC2Ei [QtGui]	_ZN23QStyleOptionProgressBarC2Ev [QtGui]
_ZN24QStyleOptionGraphicsItemC1Ei [QtXml]	_ZN24QStyleOptionGraphicsItemC1Ev [QtXml]
_ZN24QStyleOptionGraphicsItemC2Ei [QtXml]	_ZN24QStyleOptionGraphicsItemC2Ev [QtXml]
_ZN24QStyleOptionQ3DockWindowC1Ei [QtGui]	_ZN24QStyleOptionQ3DockWindowC1Ev [QtGui]
_ZN24QStyleOptionQ3DockWindowC2Ei [QtGui]	_ZN24QStyleOptionQ3DockWindowC2Ev [QtGui]
_ZN25QStyleOptionProgressBarV2C1ERK23QStyleOptionProgressBar [QtGui]	_ZN25QStyleOptionProgressBarV2C1ERKS_ [QtGui]
_ZN25QStyleOptionProgressBarV2C1Ei [QtGui]	_ZN25QStyleOptionProgressBarV2C1Ev [QtGui]
_ZN25QStyleOptionProgressBarV2C2ERK23QStyleOptionProgressBar [QtGui]	_ZN25QStyleOptionProgressBarV2C2ERKS_ [QtGui]
_ZN25QStyleOptionProgressBarV2C2Ei [QtGui]	_ZN25QStyleOptionProgressBarV2C2Ev [QtGui]
_ZN25QStyleOptionProgressBarV2aSERK23QStyleOptionProgressBar [QtGui]	_ZN26QStyleOptionQ3ListViewItemC1Ei [QtGui]
_ZN26QStyleOptionQ3ListViewItemC1Ev [QtGui]	_ZN26QStyleOptionQ3ListViewItemC2Ei [QtGui]
_ZN26QStyleOptionQ3ListViewItemC2Ev [QtGui]	_ZN26QStyleOptionTabWidgetFrameC1Ei [QtGui]
_ZN26QStyleOptionTabWidgetFrameC1Ev [QtGui]	_ZN26QStyleOptionTabWidgetFrameC2Ei [QtGui]
_ZN26QStyleOptionTabWidgetFrameC2Ev [QtGui]	_ZN6QStyle10visualRectEN2Qt15LayoutDirectionERK5QRectS4_ [QtGui]
_ZN6QStyle11alignedRectEN2Qt15LayoutDirectionE6QFlagsINS0_13Align	_ZN6QStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]

nmentFlagEERK5QSizeRK5QRect [QtGui]	
_ZN6QStyle11qt_metacastEPKc [QtGui]	_ZN6QStyle15visualAlignmentEN2Qt15LayoutDirectionE6QFlagsINS0_13AlignmentFlagEE [QtGui]
_ZN6QStyle23sliderPositionFromValueEiiiiib [QtGui]	_ZN6QStyle23sliderValueFromPositionEiiiiib [QtGui]
_ZN6QStyle6polishEP12QApplication [QtGui]	_ZN6QStyle6polishEP7QWidget [QtGui]
_ZN6QStyle6polishER8QPalette [QtGui]	_ZN6QStyle8unpolishEP12QApplication [QtGui]
_ZN6QStyle8unpolishEP7QWidget [QtGui]	_ZN6QStyle9visualPosEN2Qt15LayoutDirectionERK5QRectRK6QPoint [QtGui]
_ZN6QStyleC1Ev [QtGui]	_ZN6QStyleC2Ev [QtGui]
_ZN6QStyleD0Ev [QtGui]	_ZN6QStyleD1Ev [QtGui]
_ZN6QStyleD2Ev [QtGui]	_ZN9QCDEStyle11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QCDEStyle11qt_metacastEPKc [QtGui]	_ZN9QCDEStyleC1Eb [QtGui]
_ZN9QCDEStyleC2Eb [QtGui]	_ZN9QCDEStyleD0Ev [QtGui]
_ZN9QCDEStyleD1Ev [QtGui]	_ZN9QCDEStyleD2Ev [QtGui]
_ZNK11QMotifStyle10metaObjectEv [QtGui]	_ZNK11QMotifStyle11drawControlEN6QStyle14ControlElementEPK12QStyleOptionP8QPainterPK7QWidget [QtGui]
_ZNK11QMotifStyle11pixelMetricEN6QStyle11PixelMetricEPK12QStyleOptionPK7QWidget [QtGui]	_ZNK11QMotifStyle13drawPrimitiveEN6QStyle16PrimitiveElementEPK12QStyleOptionP8QPainterPK7QWidget [QtGui]
_ZNK11QMotifStyle14standardPixmapEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtGui]	_ZNK11QMotifStyle14subControlRectEN6QStyle14ComplexControlEPK19QStyleOptionComplexNS0_10SubControlEPK7QWidget [QtGui]
_ZNK11QMotifStyle14subElementRectEN6QStyle10SubElementEPK12QStyleOptionPK7QWidget [QtGui]	_ZNK11QMotifStyle15standardPaletteEv [QtGui]
_ZNK11QMotifStyle16sizeFromContentsEN6QStyle12ContentsTypeEPK12QStyleOptionRK5QSizePK7QWidget [QtGui]	_ZNK11QMotifStyle18drawComplexControlEN6QStyle14ComplexControlEPK19QStyleOptionComplexP8QPainterPK7QWidget [QtGui]
_ZNK11QMotifStyle18useHighlightColorsEv [QtGui]	_ZNK11QMotifStyle26standardIconImplementationEN6QStyle14Standard

	dPixmapEPK12QStyleOptionPK7QW idget [QtXml]
_ZNK11QMotifStyle9styleHintEN6Q Style9StyleHintEPK12QStyleOptionP K7QWidgetP16QStyleHintReturn [QtGui]	_ZNK12QCommonStyle10metaObjec tEv [QtGui]
_ZNK12QCommonStyle11drawCont rolEN6QStyle14ControlElementEPK1 2QStyleOptionP8QPainterPK7QWid get [QtGui]	_ZNK12QCommonStyle11pixelMetri cEN6QStyle11PixelMetricEPK12QSty leOptionPK7QWidget [QtGui]
_ZNK12QCommonStyle13drawPrimi tiveEN6QStyle16PrimitiveElementEP K12QStyleOptionP8QPainterPK7QW idget [QtGui]	_ZNK12QCommonStyle14standardP ixmapEN6QStyle14StandardPixmap EPK12QStyleOptionPK7QWidget [QtGui]
_ZNK12QCommonStyle14subContro lRectEN6QStyle14ComplexControlE PK19QStyleOptionComplexNS0_10S ubControlEPK7QWidget [QtGui]	_ZNK12QCommonStyle14subEleme ntRectEN6QStyle10SubElementEPK1 2QStyleOptionPK7QWidget [QtGui]
_ZNK12QCommonStyle16sizeFromC ontentsEN6QStyle12ContentsTypeEP K12QStyleOptionRK5QSizePK7QWi dget [QtGui]	_ZNK12QCommonStyle18drawCom plexControlEN6QStyle14ComplexCo ntrolEPK19QStyleOptionComplexP8 QPainterPK7QWidget [QtGui]
_ZNK12QCommonStyle19generatedI conPixmapEN5QIcon4ModeERK7QP ixmapPK12QStyleOption [QtGui]	_ZNK12QCommonStyle21hitTestCo mplexControlEN6QStyle14Complex ControlEPK19QStyleOptionComplex RK6QPointPK7QWidget [QtGui]
_ZNK12QCommonStyle26standardIc onImplementationEN6QStyle14Stan dardPixmapEPK12QStyleOptionPK7 QWidget [QtXml]	_ZNK12QCommonStyle9styleHintE N6QStyle9StyleHintEPK12QStyleOpt ionPK7QWidgetP16QStyleHintRetur n [QtGui]
_ZNK13QWindowsStyle10metaObjec tEv [QtGui]	_ZNK13QWindowsStyle11drawCont rolEN6QStyle14ControlElementEPK1 2QStyleOptionP8QPainterPK7QWid get [QtGui]
_ZNK13QWindowsStyle11pixelMetri cEN6QStyle11PixelMetricEPK12QSty leOptionPK7QWidget [QtGui]	_ZNK13QWindowsStyle13drawPrim itiveEN6QStyle16PrimitiveElementE PK12QStyleOptionP8QPainterPK7Q Widget [QtGui]
_ZNK13QWindowsStyle14standardP ixmapEN6QStyle14StandardPixmap EPK12QStyleOptionPK7QWidget [QtGui]	_ZNK13QWindowsStyle14subEleme ntRectEN6QStyle10SubElementEPK1 2QStyleOptionPK7QWidget [QtGui]
_ZNK13QWindowsStyle16sizeFrom ContentsEN6QStyle12ContentsType EPK12QStyleOptionRK5QSizePK7Q Widget [QtGui]	_ZNK13QWindowsStyle18drawCom plexControlEN6QStyle14ComplexCo ntrolEPK19QStyleOptionComplexP8 QPainterPK7QWidget [QtGui]

_ZNK13QWindowsStyle26standardImplementationEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtXml]	_ZNK13QWindowsStyle9styleHintEN6QStyle9StyleHintEPK12QStyleOptionPK7QWidgetP16QStyleHintReturn [QtGui]
_ZNK15QPlastiqueStyle10metaObjectEv [QtGui]	_ZNK15QPlastiqueStyle11drawControlEN6QStyle14ControlElementEPK12QStyleOptionP8QPainterPK7QWidget [QtGui]
_ZNK15QPlastiqueStyle11pixelMetricEN6QStyle11PixelMetricEPK12QStyleOptionPK7QWidget [QtGui]	_ZNK15QPlastiqueStyle13drawPrimitiveEN6QStyle16PrimitiveElementEPK12QStyleOptionP8QPainterPK7QWidget [QtGui]
_ZNK15QPlastiqueStyle14standardPixmapEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtXml]	_ZNK15QPlastiqueStyle14subControlRectEN6QStyle14ComplexControlEPK19QStyleOptionComplexNS0_10SubControlEPK7QWidget [QtGui]
_ZNK15QPlastiqueStyle14subElementRectEN6QStyle10SubElementEPK12QStyleOptionPK7QWidget [QtGui]	_ZNK15QPlastiqueStyle15standardPaletteEv [QtGui]
_ZNK15QPlastiqueStyle16sizeFromContentsEN6QStyle12ContentsTypeEPK12QStyleOptionRK5QSizePK7QWidget [QtGui]	_ZNK15QPlastiqueStyle18drawComplexControlEN6QStyle14ComplexControlEPK19QStyleOptionComplexP8QPainterPK7QWidget [QtGui]
_ZNK15QPlastiqueStyle21hitTestComplexControlEN6QStyle14ComplexControlEPK19QStyleOptionComplexRK6QPointPK7QWidget [QtGui]	_ZNK15QPlastiqueStyle26standardIconImplementationEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtXml]
_ZNK15QPlastiqueStyle9styleHintEN6QStyle9StyleHintEPK12QStyleOptionPK7QWidgetP16QStyleHintReturn [QtGui]	_ZNK16QCleanlooksStyle10metaObjectEv [QtXml]
_ZNK16QCleanlooksStyle11drawControlEN6QStyle14ControlElementEPK12QStyleOptionP8QPainterPK7QWidget [QtXml]	_ZNK16QCleanlooksStyle11pixelMetricEN6QStyle11PixelMetricEPK12QStyleOptionPK7QWidget [QtXml]
_ZNK16QCleanlooksStyle12drawItemTextEP8QPainterRK5QRectiRK8QPalettebRK7QStringNS5_9ColorRole [QtXml]	_ZNK16QCleanlooksStyle13drawPrimitiveEN6QStyle16PrimitiveElementEPK12QStyleOptionP8QPainterPK7QWidget [QtXml]
_ZNK16QCleanlooksStyle14drawItemPixmapEP8QPainterRK5QRectiRK7QPixmap [QtXml]	_ZNK16QCleanlooksStyle14itemPixmapRectERK5QRectiRK7QPixmap [QtXml]
_ZNK16QCleanlooksStyle14standardPixmapEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtXml]	_ZNK16QCleanlooksStyle14subControlRectEN6QStyle14ComplexControlEPK19QStyleOptionComplexNS0_10SubControlEPK7QWidget [QtXml]

_Znk16QCleanlooksStyle14subElementRectEN6QStyle10SubElementEPK12QStyleOptionPK7QWidget [QtXml]	_Znk16QCleanlooksStyle15standardPaletteEv [QtXml]
_Znk16QCleanlooksStyle16sizeFromContentsEN6QStyle12ContentsTypeEPK12QStyleOptionRK5QSizePK7QWidget [QtXml]	_Znk16QCleanlooksStyle18drawComplexControlEN6QStyle14ComplexControlEPK19QStyleOptionComplexP8QPainterPK7QWidget [QtXml]
_Znk16QCleanlooksStyle19generateIconPixmapEN5QIcon4ModeERK7QPixmapPK12QStyleOption [QtXml]	_Znk16QCleanlooksStyle21hitTestComplexControlEN6QStyle14ComplexControlEPK19QStyleOptionComplexRK6QPointPK7QWidget [QtXml]
_Znk16QCleanlooksStyle26standardIconImplementationEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtXml]	_Znk16QCleanlooksStyle9styleHintEN6QStyle9StyleHintEPK12QStyleOptionPK7QWidgetP16QStyleHintReturn [QtXml]
_Znk6QStyle10metaObjectEv [QtGui]	_Znk6QStyle12drawItemTextEP8QPainterRK5QRectRK8QPalettebRK7QStringNS5_9ColorRoleE [QtGui]
_Znk6QStyle12itemTextRectERK12QFontMetricsRK5QRectbRK7QString [QtGui]	_Znk6QStyle12standardIconENS_14StandardPixmapEPK12QStyleOptionPK7QWidget [QtGui]
_Znk6QStyle14drawItemPixmapEP8QPainterRK5QRectRK7QPixmap [QtGui]	_Znk6QStyle14itemPixmapRectERK5QRectRK7QPixmap [QtGui]
_Znk6QStyle15standardPaletteEv [QtGui]	_Znk6QStyle26standardIconImplementationENS_14StandardPixmapEPK12QStyleOptionPK7QWidget [QtGui]
_Znk9QCDEStyle10metaObjectEv [QtGui]	_Znk9QCDEStyle11drawControlEN6QStyle14ControlElementEPK12QStyleOptionP8QPainterPK7QWidget [QtGui]
_Znk9QCDEStyle11pixelMetricEN6QStyle11PixelMetricEPK12QStyleOptionPK7QWidget [QtGui]	_Znk9QCDEStyle13drawPrimitiveEN6QStyle16PrimitiveElementEPK12QStyleOptionP8QPainterPK7QWidget [QtGui]
_Znk9QCDEStyle15standardPaletteEv [QtGui]	_Znk9QCDEStyle26standardIconImplementationEN6QStyle14StandardPixmapEPK12QStyleOptionPK7QWidget [QtXml]

18.5.17 Qt4 Text

18.5.17.1 Class data for QTextObject

The virtual table for the QTextObject class is described by Table 18-363

Table 18-363 Primary vtable for QTextObject

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextObject
vfunc[0]:	QTextObject::metaObject() const
vfunc[1]:	QTextObject::qt_metacast(char const*)
vfunc[2]:	QTextObject::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTextObject::~~QTextObject()
vfunc[4]:	QTextObject::~QTextObject()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QTextObject class is described by Table 18-364

Table 18-364 typeinfo for QTextObject

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextObject
basetype:	typeinfo for QObject

18.5.17.2 Class data for QTextBlockGroup

The virtual table for the QTextBlockGroup class is described by Table 18-365

Table 18-365 Primary vtable for QTextBlockGroup

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextBlockGroup
vfunc[0]:	QTextBlockGroup::metaObject() const

vfunc[1]:	QTextBlockGroup::qt_metacast(char const*)
vfunc[2]:	QTextBlockGroup::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTextBlockGroup::~~QTextBlockGroup()
vfunc[4]:	QTextBlockGroup::~~QTextBlockGroup()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QTextBlockGroup::blockInserted(QTextBlock const&)
vfunc[13]:	QTextBlockGroup::blockRemoved(QTextBlock const&)
vfunc[14]:	QTextBlockGroup::blockFormatChanged(QTextBlock const&)

The Run Time Type Information for the QTextBlockGroup class is described by Table 18-366

Table 18-366 typeinfo for QTextBlockGroup

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextBlockGroup
basetype:	typeinfo for QTextObject

18.5.17.3 Class data for QTextFrameLayoutData

The virtual table for the QTextFrameLayoutData class is described by Table 18-367

Table 18-367 Primary vtable for QTextFrameLayoutData

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextFrameLayoutData

vfunc[0]:	QTextFrameLayoutData::~~QTextFrameLayoutData()
vfunc[1]:	QTextFrameLayoutData::~~QTextFrameLayoutData()

The Run Time Type Information for the QTextFrameLayoutData class is described by Table 18-368

Table 18-368 typeinfo for QTextFrameLayoutData

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTextFrameLayoutData

18.5.17.4 Class data for QTextFrame

The virtual table for the QTextFrame class is described by Table 18-369

Table 18-369 Primary vtable for QTextFrame

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextFrame
vfunc[0]:	QTextFrame::metaObject() const
vfunc[1]:	QTextFrame::qt_metacast(char const*)
vfunc[2]:	QTextFrame::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTextFrame::~~QTextFrame()
vfunc[4]:	QTextFrame::~~QTextFrame()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QTextFrame class is described by Table 18-370

Table 18-370 typeinfo for QTextFrame

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextFrame
basetype:	typeinfo for QTextObject

18.5.17.5 Class data for QTextBlockUserData

The virtual table for the QTextBlockUserData class is described by Table 18-371

Table 18-371 Primary vtable for QTextBlockUserData

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextBlockUserData
vfunc[0]:	QTextBlockUserData::~~QTextBlockUserData()
vfunc[1]:	QTextBlockUserData::~~QTextBlockUserData()

The Run Time Type Information for the QTextBlockUserData class is described by Table 18-372

Table 18-372 typeinfo for QTextBlockUserData

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTextBlockUserData

18.5.17.6 Class data for QTextDocument

The virtual table for the QTextDocument class is described by Table 18-373

Table 18-373 Primary vtable for QTextDocument

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextDocument
vfunc[0]:	QTextDocument::metaObject() const
vfunc[1]:	QTextDocument::qt_metacast(char const*)
vfunc[2]:	QTextDocument::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTextDocument::~~QTextDocument()
vfunc[4]:	QTextDocument::~~QTextDocument()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QTextDocument::clear()
vfunc[13]:	QTextDocument::createObject(QTextFormat const&)
vfunc[14]:	QTextDocument::loadResource(int, QUrl const&)

The Run Time Type Information for the QTextDocument class is described by Table 18-374

Table 18-374 typeinfo for QTextDocument

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextDocument
basetype:	typeinfo for QObject

18.5.17.7 Class data for QTextTable

The virtual table for the QTextTable class is described by Table 18-375

Table 18-375 Primary vtable for QTextTable

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextTable
vfunc[0]:	QTextTable::metaObject() const
vfunc[1]:	QTextTable::qt_metacast(char const*)
vfunc[2]:	QTextTable::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTextTable::~~QTextTable()
vfunc[4]:	QTextTable::~~QTextTable()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QTextTable class is described by Table 18-376

Table 18-376 typeinfo for QTextTable

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextTable
basetype:	typeinfo for QTextFrame

18.5.17.8 Class data for QTextList

The virtual table for the QTextList class is described by Table 18-377

Table 18-377 Primary vtable for QTextList

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextList
vfunc[0]:	QTextList::metaObject() const
vfunc[1]:	QTextList::qt_metacast(char const*)
vfunc[2]:	QTextList::qt_metacall(QMetaObject: :Call, int, void**)
vfunc[3]:	QTextList::~~QTextList()
vfunc[4]:	QTextList::~QTextList()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QTextBlockGroup::blockInserted(QTextBlock const&)

vfunc[13]:	QTextBlockGroup::blockRemoved(QTextBlock const&)
vfunc[14]:	QTextBlockGroup::blockFormatChanged(QTextBlock const&)

The Run Time Type Information for the QTextList class is described by Table 18-378

Table 18-378 typeinfo for QTextList

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextList
basetype:	typeinfo for QTextBlockGroup

18.5.17.9 Class data for QTextObjectInterface

The virtual table for the QTextObjectInterface class is described by Table 18-379

Table 18-379 Primary vtable for QTextObjectInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextObjectInterface
vfunc[0]:	NULL or QTextObjectInterface::~~QTextObjectInterface()
vfunc[1]:	NULL or QTextObjectInterface::~~QTextObjectInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QTextObjectInterface class is described by Table 18-380

Table 18-380 typeinfo for QTextObjectInterface

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QTextObjectInterface

18.5.17.10 Class data for QSyntaxHighlighter

The virtual table for the QSyntaxHighlighter class is described by Table 18-381

Table 18-381 Primary vtable for QSyntaxHighlighter

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QSyntaxHighlighter
vfunc[0]:	QSyntaxHighlighter::metaObject() const
vfunc[1]:	QSyntaxHighlighter::qt_metacast(char const*)
vfunc[2]:	QSyntaxHighlighter::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSyntaxHighlighter::~QSyntaxHighlighter()
vfunc[4]:	QSyntaxHighlighter::~QSyntaxHighlighter()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual

The Run Time Type Information for the QSyntaxHighlighter class is described by Table 18-382

Table 18-382 typeinfo for QSyntaxHighlighter

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSyntaxHighlighter
basetype:	typeinfo for QObject

18.5.17.11 Interfaces for Qt4 Text

An LSB conforming implementation shall provide the generic functions for Qt4 Text specified in Table 18-383, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-383 libQtGui - Qt4 Text Function Interfaces

_ZN10QTextBlock11setUserDataEP18QTextBlockUserData [QtGui]	_ZN10QTextBlock12setUserStateEi [QtGui]

_ZN10QTextFrame11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN10QTextFrame11qt_metacastEPKc [QtGui]
_ZN10QTextFrame13setLayoutDataEP20QTextFrameLayoutData [QtGui]	_ZN10QTextFrame8iteratorC1ERKS0_ [QtGui]
_ZN10QTextFrame8iteratorC1Ev [QtGui]	_ZN10QTextFrame8iteratorC2ERKS0_ [QtGui]
_ZN10QTextFrame8iteratorC2Ev [QtGui]	_ZN10QTextFrameC1EP13QTextDocument [QtGui]
_ZN10QTextFrameC2EP13QTextDocument [QtGui]	_ZN10QTextFrameD0Ev [QtGui]
_ZN10QTextFrameD1Ev [QtGui]	_ZN10QTextFrameD2Ev [QtGui]
_ZN10QTextTable10insertRowsEii [QtGui]	_ZN10QTextTable10mergeCellsERK11QTextCursor [QtGui]
_ZN10QTextTable10mergeCellsEiiii [QtGui]	_ZN10QTextTable10removeRowsEii [QtGui]
_ZN10QTextTable11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN10QTextTable11qt_metacastEPKc [QtGui]
_ZN10QTextTable13insertColumnsEii [QtGui]	_ZN10QTextTable13removeColumnsEii [QtGui]
_ZN10QTextTable6resizeEii [QtGui]	_ZN10QTextTable9splitCellEiiii [QtGui]
_ZN10QTextTableC1EP13QTextDocument [QtGui]	_ZN10QTextTableC2EP13QTextDocument [QtGui]
_ZN10QTextTableD0Ev [QtGui]	_ZN10QTextTableD1Ev [QtGui]
_ZN10QTextTableD2Ev [QtGui]	_ZN11QTextCursor10createListEN15QTextListFormat5StyleE [QtGui]
_ZN11QTextCursor10createListERK15QTextListFormat [QtGui]	_ZN11QTextCursor10deleteCharEv [QtGui]
_ZN11QTextCursor10insertHtmlERK7QString [QtGui]	_ZN11QTextCursor10insertListEN15QTextListFormat5StyleE [QtGui]
_ZN11QTextCursor10insertListERK15QTextListFormat [QtGui]	_ZN11QTextCursor10insertTextERK7QString [QtGui]
_ZN11QTextCursor10insertTextERK7QStringRK15QTextCharFormat [QtGui]	_ZN11QTextCursor11insertBlockERK16QTextBlockFormat [QtGui]
_ZN11QTextCursor11insertBlockERK16QTextBlockFormatRK15QTextCharFormat [QtGui]	_ZN11QTextCursor11insertBlockEv [QtGui]
_ZN11QTextCursor11insertFrameERK16QTextFrameFormat [QtGui]	_ZN11QTextCursor11insertImageERK16QTextImageFormat [QtGui]

_ZN11QTextCursor11insertImageERK16QTextImageFormatN16QTextFrameFormat8PositionE [QtXml]	_ZN11QTextCursor11insertImageERK7QString [QtGui]
_ZN11QTextCursor11insertTableEii [QtGui]	_ZN11QTextCursor11insertTableEiiRK16QTextTableFormat [QtGui]
_ZN11QTextCursor11setPositionEiNS_8MoveModeE [QtGui]	_ZN11QTextCursor12endEditBlockEv [QtGui]
_ZN11QTextCursor12movePositionENS_13MoveOperationENS_8MoveModeEi [QtGui]	_ZN11QTextCursor13setCharFormatERK15QTextCharFormat [QtGui]
_ZN11QTextCursor14beginEditBlockEv [QtGui]	_ZN11QTextCursor14clearSelectionEv [QtGui]
_ZN11QTextCursor14insertFragmentERK21QTextDocumentFragment [QtGui]	_ZN11QTextCursor14setBlockFormatERK16QTextBlockFormat [QtGui]
_ZN11QTextCursor15mergeCharFormatERK15QTextCharFormat [QtGui]	_ZN11QTextCursor16mergeBlockFormatERK16QTextBlockFormat [QtGui]
_ZN11QTextCursor18deletePreviousCharEv [QtGui]	_ZN11QTextCursor18removeSelectedTextEv [QtGui]
_ZN11QTextCursor18setBlockCharFormatERK15QTextCharFormat [QtGui]	_ZN11QTextCursor20mergeBlockCharFormatERK15QTextCharFormat [QtGui]
_ZN11QTextCursor21joinPreviousEditBlockEv [QtGui]	_ZN11QTextCursor6selectENS_13SelectionTypeE [QtGui]
_ZN11QTextCursorC1EP10QTextFrame [QtGui]	_ZN11QTextCursorC1EP13QTextDocument [QtGui]
_ZN11QTextCursorC1ERK10QTextBlock [QtGui]	_ZN11QTextCursorC1ERKS_ [QtGui]
_ZN11QTextCursorC1Ev [QtGui]	_ZN11QTextCursorC2EP10QTextFrame [QtGui]
_ZN11QTextCursorC2EP13QTextDocument [QtGui]	_ZN11QTextCursorC2ERK10QTextBlock [QtGui]
ZN11QTextCursorC2ERKS [QtGui]	_ZN11QTextCursorC2Ev [QtGui]
_ZN11QTextCursorD1Ev [QtGui]	_ZN11QTextCursorD2Ev [QtGui]
ZN11QTextCursoraSERKS [QtGui]	_ZN11QTextFormat11setPropertyEiRK7QVectorI11QTextLengthE [QtGui]
_ZN11QTextFormat11setPropertyEiRK8QVariant [QtGui]	_ZN11QTextFormat13clearPropertyEi [QtGui]
_ZN11QTextFormat14setObjectIndexEi [QtGui]	_ZN11QTextFormat5mergeERKS_ [QtGui]

ZN11QTextFormatC1ERKS [QtGui]	_ZN11QTextFormatC1Ei [QtGui]
_ZN11QTextFormatC1Ev [QtGui]	_ZN11QTextFormatC2ERKS_ [QtGui]
_ZN11QTextFormatC2Ei [QtGui]	_ZN11QTextFormatC2Ev [QtGui]
_ZN11QTextFormatD1Ev [QtGui]	_ZN11QTextFormatD2Ev [QtGui]
ZN11QTextFormataSERKS [QtGui]	_ZN11QTextLayout10createLineEv [QtGui]
_ZN11QTextLayout11beginLayoutEv [QtGui]	_ZN11QTextLayout11setPositionER K7QPointF [QtGui]
_ZN11QTextLayout13setTextOption ERK11QTextOption [QtGui]	_ZN11QTextLayout14setPreeditArea EiRK7QString [QtGui]
_ZN11QTextLayout15setCacheEnabl edEb [QtGui]	_ZN11QTextLayout20setAdditionalF ormatsERK5QListINS_11FormatRan geEE [QtGui]
_ZN11QTextLayout22clearAdditional FormatsEv [QtGui]	_ZN11QTextLayout7setFontERK5QF ont [QtGui]
_ZN11QTextLayout7setTextERK7QS tring [QtGui]	_ZN11QTextLayout9endLayoutEv [QtGui]
_ZN11QTextLayoutC1ERK10QTextB lock [QtGui]	_ZN11QTextLayoutC1ERK7QString [QtGui]
_ZN11QTextLayoutC1ERK7QString RK5QFontP12QPaintDevice [QtGui]	_ZN11QTextLayoutC1Ev [QtGui]
_ZN11QTextLayoutC2ERK10QTextB lock [QtGui]	_ZN11QTextLayoutC2ERK7QString [QtGui]
_ZN11QTextLayoutC2ERK7QString RK5QFontP12QPaintDevice [QtGui]	_ZN11QTextLayoutC2Ev [QtGui]
_ZN11QTextLayoutD1Ev [QtGui]	_ZN11QTextLayoutD2Ev [QtGui]
_ZN11QTextObject11qt_metacallEN1 1QMetaObject4CallEiPPv [QtGui]	_ZN11QTextObject11qt_metacastEP Kc [QtGui]
_ZN11QTextObject9setFormatERK11 QTextFormat [QtGui]	_ZN11QTextObjectC1EP13QTextDoc ument [QtGui]
_ZN11QTextObjectC2EP13QTextDoc ument [QtGui]	_ZN11QTextObjectD0Ev [QtGui]
_ZN11QTextObjectD1Ev [QtGui]	_ZN11QTextObjectD2Ev [QtGui]
_ZN11QTextOption11setTabArrayE5 QListIdE [QtGui]	_ZN11QTextOptionC1E6QFlagsIN2 Qt13AlignmentFlagEE [QtGui]
ZN11QTextOptionC1ERKS [QtGui]	_ZN11QTextOptionC1Ev [QtGui]
_ZN11QTextOptionC2E6QFlagsIN2 Qt13AlignmentFlagEE [QtGui]	_ZN11QTextOptionC2ERKS_ [QtGui]

_ZN11QTextOptionC2Ev [QtGui]	_ZN11QTextOptionD1Ev [QtGui]
_ZN11QTextOptionD2Ev [QtGui]	_ZN11QTextOptionaSERKS_ [QtGui]
_ZN13QTextDocument10adjustSizeEv [QtXml]	_ZN13QTextDocument11addResourceEiRK4QUrlRK8QVariant [QtGui]
_ZN13QTextDocument11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN13QTextDocument11qt_metacastEPKc [QtGui]
_ZN13QTextDocument11setModifiedEb [QtGui]	_ZN13QTextDocument11setPageSizeERK6QSizeF [QtGui]
_ZN13QTextDocument12createObjectERK11QTextFormat [QtGui]	_ZN13QTextDocument12drawContentsEP8QPainterRK6QRectF [QtXml]
_ZN13QTextDocument12loadResourceEiRK4QUrl [QtGui]	_ZN13QTextDocument12setPlainTextERK7QString [QtGui]
_ZN13QTextDocument12setTextWidthEd [QtXml]	_ZN13QTextDocument13redoAvailableEb [QtGui]
_ZN13QTextDocument13undoAvailableEb [QtGui]	_ZN13QTextDocument14appendUndoItemEP17QAbstractUndoItem [LSB]
_ZN13QTextDocument14contentsChangeEiii [QtGui]	_ZN13QTextDocument14setDefaultFontERK5QFont [QtGui]
_ZN13QTextDocument15contentsChangedEv [QtGui]	_ZN13QTextDocument17markContentsDirtyEii [QtGui]
_ZN13QTextDocument17setLayoutEP27QAbstractTextDocumentLayout [QtGui]	_ZN13QTextDocument18setMetaInformationENS_15MetaInformationERK7QString [QtGui]
_ZN13QTextDocument18setUndoRedoEnabledEb [QtGui]	_ZN13QTextDocument19modificationChangedEb [QtGui]
_ZN13QTextDocument19setUseDesignMetricsEb [QtGui]	_ZN13QTextDocument20setDefaultStyleSheetERK7QString [QtXml]
_ZN13QTextDocument20setMaximumBlockCountEi [QtXml]	_ZN13QTextDocument21cursorPositionChangedERK11QTextCursor [QtGui]
_ZN13QTextDocument4redoEP11QTextCursor [QtXml]	_ZN13QTextDocument4redoEv [QtGui]
_ZN13QTextDocument4undoEP11QTextCursor [QtXml]	_ZN13QTextDocument4undoEv [QtGui]
_ZN13QTextDocument5clearEv [QtGui]	_ZN13QTextDocument7setHtmlERK7QString [QtGui]
_ZN13QTextDocumentC1EP7QObject [QtGui]	_ZN13QTextDocumentC1ERK7QStringP7QObject [QtGui]
_ZN13QTextDocumentC2EP7QObject [QtGui]	_ZN13QTextDocumentC2ERK7QStringP7QObject [QtGui]

_ZN13QTextDocumentD0Ev [QtGui]	_ZN13QTextDocumentD1Ev [QtGui]
_ZN13QTextDocumentD2Ev [QtGui]	_ZN14QTextTableCell9setFormatERK15QTextCharFormat [QtXml]
_ZN15QTextBlockGroup11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN15QTextBlockGroup11qt_metacastEPKc [QtGui]
_ZN15QTextBlockGroup12blockRemovedERK10QTextBlock [QtGui]	_ZN15QTextBlockGroup13blockInsertedERK10QTextBlock [QtGui]
_ZN15QTextBlockGroup18blockFormatChangedERK10QTextBlock [QtGui]	_ZN15QTextBlockGroupC1EP13QTextDocument [QtGui]
_ZN15QTextBlockGroupC2EP13QTextDocument [QtGui]	_ZN15QTextBlockGroupD0Ev [QtGui]
_ZN15QTextBlockGroupD1Ev [QtGui]	_ZN15QTextBlockGroupD2Ev [QtGui]
_ZN15QTextCharFormat17setUnderlineStyleENS_14UnderlineStyleE [QtXml]	_ZN15QTextCharFormat7setFontERK5QFont [QtGui]
_ZN15QTextCharFormatC1Ev [QtGui]	_ZN15QTextCharFormatC2Ev [QtGui]
_ZN15QTextListFormatC1Ev [QtGui]	_ZN15QTextListFormatC2Ev [QtGui]
_ZN16QTextBlockFormatC1Ev [QtGui]	_ZN16QTextBlockFormatC2Ev [QtGui]
_ZN16QTextFrameFormatC1Ev [QtGui]	_ZN16QTextFrameFormatC2Ev [QtGui]
_ZN16QTextImageFormatC1Ev [QtGui]	_ZN16QTextImageFormatC2Ev [QtGui]
_ZN16QTextTableFormatC1Ev [QtGui]	_ZN16QTextTableFormatC2Ev [QtGui]
_ZN17QTextInlineObject10setDescender [QtGui]	_ZN17QTextInlineObject8setWidth [QtGui]
_ZN17QTextInlineObject9setAscent [QtGui]	_ZN18QSyntaxHighlighter11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN18QSyntaxHighlighter11qt_metacastEPKc [QtGui]	_ZN18QSyntaxHighlighter11rehighlightEv [QtXml]
_ZN18QSyntaxHighlighter11setDocumentEP13QTextDocument [QtGui]	_ZN18QSyntaxHighlighter20setCurrentBlockStateEi [QtGui]
_ZN18QSyntaxHighlighter23setCurrentBlockUserDataEP18QTextBlockUserData [QtGui]	_ZN18QSyntaxHighlighter9setFormatEiiRK15QTextCharFormat [QtGui]
_ZN18QSyntaxHighlighter9setFormatEiiRK5QFont [QtGui]	_ZN18QSyntaxHighlighter9setFormatEiiRK6QColor [QtGui]

_ZN18QSyntaxHighlighterC1EP13Q TextDocument [QtGui]	_ZN18QSyntaxHighlighterC1EP7QO bject [QtGui]
_ZN18QSyntaxHighlighterC1EP9QT extEdit [QtGui]	_ZN18QSyntaxHighlighterC2EP13Q TextDocument [QtGui]
_ZN18QSyntaxHighlighterC2EP7QO bject [QtGui]	_ZN18QSyntaxHighlighterC2EP9QT extEdit [QtGui]
_ZN18QSyntaxHighlighterD0Ev [QtGui]	_ZN18QSyntaxHighlighterD1Ev [QtGui]
_ZN18QSyntaxHighlighterD2Ev [QtGui]	_ZN18QTextBlockUserDataD0Ev [QtGui]
_ZN18QTextBlockUserDataD1Ev [QtGui]	_ZN18QTextBlockUserDataD2Ev [QtGui]
_ZN20QTextFrameLayoutDataD0Ev [QtGui]	_ZN20QTextFrameLayoutDataD1Ev [QtGui]
_ZN20QTextFrameLayoutDataD2Ev [QtGui]	_ZN21QTextDocumentFragment13fr omPlainTextERK7QString [QtGui]
_ZN21QTextDocumentFragment8fro mHtmlERK7QString [QtGui]	_ZN21QTextDocumentFragment8fro mHtmlERK7QStringPK13QTextDocu ment [QtXml]
_ZN21QTextDocumentFragmentC1E PK13QTextDocument [QtGui]	_ZN21QTextDocumentFragmentC1E RK11QTextCursor [QtGui]
ZN21QTextDocumentFragmentC1E RKS [QtGui]	_ZN21QTextDocumentFragmentC1E v [QtGui]
_ZN21QTextDocumentFragmentC2E PK13QTextDocument [QtGui]	_ZN21QTextDocumentFragmentC2E RK11QTextCursor [QtGui]
ZN21QTextDocumentFragmentC2E RKS [QtGui]	_ZN21QTextDocumentFragmentC2E v [QtGui]
_ZN21QTextDocumentFragmentD1E v [QtGui]	_ZN21QTextDocumentFragmentD2E v [QtGui]
ZN21QTextDocumentFragmentaSE RKS [QtGui]	_ZN9QTextLine11setPositionERK7Q PointF [QtGui]
_ZN9QTextLine12setLineWidthEd [QtGui]	_ZN9QTextLine13setNumColumnsE i [QtGui]
_ZN9QTextList10removeItemEi [QtGui]	_ZN9QTextList11qt_metacallEN11Q MetaObject4CallEiPPv [QtGui]
_ZN9QTextList11qt_metacastEPKc [QtGui]	_ZN9QTextList3addERK10QTextBlo ck [QtGui]
_ZN9QTextList6removeERK10QText Block [QtGui]	_ZN9QTextListC1EP13QTextDocum ent [QtGui]
_ZN9QTextListC2EP13QTextDocum ent [QtGui]	_ZN9QTextListD0Ev [QtGui]
_ZN9QTextListD1Ev [QtGui]	_ZN9QTextListD2Ev [QtGui]

_ZNK10QTextBlock10charFormatEv [QtGui]	_ZNK10QTextBlock11blockFormatEv [QtGui]
_ZNK10QTextBlock15charFormatIndexEv [QtGui]	_ZNK10QTextBlock16blockFormatIndexEv [QtGui]
_ZNK10QTextBlock3endEv [QtGui]	_ZNK10QTextBlock4nextEv [QtGui]
_ZNK10QTextBlock4textEv [QtGui]	_ZNK10QTextBlock5beginEv [QtGui]
_ZNK10QTextBlock6layoutEv [QtGui]	_ZNK10QTextBlock6lengthEv [QtGui]
_ZNK10QTextBlock8containsEi [QtGui]	_ZNK10QTextBlock8documentEv [QtGui]
_ZNK10QTextBlock8positionEv [QtGui]	_ZNK10QTextBlock8previousEv [QtGui]
_ZNK10QTextBlock8textListEv [QtGui]	_ZNK10QTextBlock8userDataEv [QtGui]
_ZNK10QTextBlock9userStateEv [QtGui]	_ZNK10QTextFrame10layoutDataEv [QtGui]
_ZNK10QTextFrame10metaObjectEv [QtGui]	_ZNK10QTextFrame11childFramesEv [QtGui]
_ZNK10QTextFrame11parentFrameEv [QtGui]	_ZNK10QTextFrame12lastPositionEv [QtGui]
_ZNK10QTextFrame13firstPositionEv [QtGui]	_ZNK10QTextFrame18lastCursorPositionEv [QtGui]
_ZNK10QTextFrame19firstCursorPositionEv [QtGui]	_ZNK10QTextFrame3endEv [QtGui]
_ZNK10QTextFrame5beginEv [QtGui]	_ZNK10QTextTable10metaObjectEv [QtGui]
_ZNK10QTextTable4rowsEv [QtGui]	_ZNK10QTextTable6cellAtERK11QTextCursor [QtGui]
_ZNK10QTextTable6cellAtEi [QtGui]	_ZNK10QTextTable6cellAtEii [QtGui]
_ZNK10QTextTable6rowEndERK11QTextCursor [QtGui]	_ZNK10QTextTable7columnsEv [QtGui]
_ZNK10QTextTable8rowStartERK11QTextCursor [QtGui]	_ZNK11QTextCursor10atBlockEndEv [QtGui]
_ZNK11QTextCursor10charFormatEv [QtGui]	_ZNK11QTextCursor11blockFormatEv [QtGui]
_ZNK11QTextCursor11blockNumberEv [QtXml]	_ZNK11QTextCursor11currentListEv [QtGui]
_ZNK11QTextCursor12atBlockStartEv [QtGui]	_ZNK11QTextCursor12columnNumberEv [QtXml]

_Znk11QTextCursor12currentFrameEv [QtGui]	_Znk11QTextCursor12currentTableEv [QtGui]
_Znk11QTextCursor12hasSelectionEv [QtGui]	_Znk11QTextCursor12selectedTextEv [QtGui]
_Znk11QTextCursor12selectionEndEv [QtGui]	_Znk11QTextCursor14selectionStartEv [QtGui]
_Znk11QTextCursor15blockCharFormatEv [QtGui]	_Znk11QTextCursor18selectedTableCellsEPiS0_S0_S0_ [QtGui]
_Znk11QTextCursor19hasComplexSelectionEv [QtGui]	_Znk11QTextCursor5atEndEv [QtGui]
_Znk11QTextCursor5blockEv [QtGui]	_Znk11QTextCursor6anchorEv [QtGui]
_Znk11QTextCursor6isNullEv [QtGui]	_Znk11QTextCursor7atStartEv [QtGui]
Znk11QTextCursor8isCopyOfERKS [QtGui]	_Znk11QTextCursor8positionEv [QtGui]
_Znk11QTextCursor9selectionEv [QtGui]	_Znk11QTextCursoreqERKS_ [QtGui]
Znk11QTextCursorgeERKS [QtGui]	_Znk11QTextCursorgtERKS_ [QtGui]
Znk11QTextCursorleERKS [QtGui]	_Znk11QTextCursorltERKS_ [QtGui]
Znk11QTextCursorneERKS [QtGui]	_Znk11QTextFormat10propertiesEv [QtGui]
_Znk11QTextFormat11hasPropertyEi [QtGui]	_Znk11QTextFormat11intPropertyEi [QtGui]
_Znk11QTextFormat11objectIndexEv [QtGui]	_Znk11QTextFormat11penPropertyEi [QtGui]
_Znk11QTextFormat12boolPropertyEi [QtGui]	_Znk11QTextFormat12toCharFormatEv [QtGui]
_Znk11QTextFormat12toListFormatEv [QtGui]	_Znk11QTextFormat13brushPropertyEi [QtGui]
_Znk11QTextFormat13colorPropertyEi [QtGui]	_Znk11QTextFormat13toBlockFormatEv [QtGui]
_Znk11QTextFormat13toFrameFormatEv [QtGui]	_Znk11QTextFormat13toImageFormatEv [QtGui]
_Znk11QTextFormat13toTableFormatEv [QtGui]	_Znk11QTextFormat14doublePropertyEi [QtGui]
_Znk11QTextFormat14lengthPropertyEi [QtGui]	_Znk11QTextFormat14stringPropertyEi [QtGui]
_Znk11QTextFormat20lengthVectorPropertyEi [QtGui]	_Znk11QTextFormat4typeEv [QtGui]

_ZNK11QTextFormat8propertyEi [QtGui]	_ZNK11QTextFormatcv8QVariantEv [QtGui]
ZNK11QTextFormateqERKS [QtGui]	_ZNK11QTextLayout10drawCursorEP8QPainterRK7QPointF [QtGui]
_ZNK11QTextLayout10drawCursorEP8QPainterRK7QPointF [QtGui]	_ZNK11QTextLayout10textOptionEv [QtGui]
_ZNK11QTextLayout12boundingRectEv [QtGui]	_ZNK11QTextLayout12cacheEnableEv [QtGui]
_ZNK11QTextLayout12maximumWidthEv [QtGui]	_ZNK11QTextLayout12minimumWidthEv [QtGui]
_ZNK11QTextLayout15preeditAreaTextEv [QtGui]	_ZNK11QTextLayout17additionalFormatsEv [QtGui]
_ZNK11QTextLayout18nextCursorPositionEiNS_10CursorModeE [QtGui]	_ZNK11QTextLayout19lineForTextPositionEi [QtGui]
_ZNK11QTextLayout19preeditAreaCursorPositionEv [QtGui]	_ZNK11QTextLayout21isValidCursorPositionEi [QtGui]
_ZNK11QTextLayout22previousCursorPositionEiNS_10CursorModeE [QtGui]	_ZNK11QTextLayout4drawEP8QPainterRK7QPointF RK7QVectorINS_11FormatRangeEERK6QRectF [QtGui]
_ZNK11QTextLayout4fontEv [QtGui]	_ZNK11QTextLayout4textEv [QtGui]
_ZNK11QTextLayout6lineAtEi [QtGui]	_ZNK11QTextLayout8positionEv [QtGui]
_ZNK11QTextLayout9lineCountEv [QtGui]	_ZNK11QTextLengthcv8QVariantEv [QtGui]
_ZNK11QTextObject10metaObjectEv [QtGui]	_ZNK11QTextObject11formatIndexEv [QtGui]
_ZNK11QTextObject11objectIndexEv [QtGui]	_ZNK11QTextObject6formatEv [QtGui]
_ZNK11QTextObject8documentEv [QtGui]	_ZNK11QTextObject9docHandleEv [QtGui]
_ZNK11QTextOption8tabArrayEv [QtGui]	_ZNK13QTextDocument10allFormatsEv [QtGui]
_ZNK13QTextDocument10blockCountEv [QtGui]	_ZNK13QTextDocument10idealWidthEv [QtGui]
_ZNK13QTextDocument10isModifiedEv [QtGui]	_ZNK13QTextDocument10metaObjectEv [QtGui]
_ZNK13QTextDocument11defaultFontEv [QtGui]	_ZNK13QTextDocument11toPlainTextEv [QtGui]
_ZNK13QTextDocument14documentLayoutEv [QtGui]	_ZNK13QTextDocument15isRedoAvailableEv [QtGui]

_ZNK13QTextDocument15isUndoAvailableEv [QtGui]	_ZNK13QTextDocument15metaInformationENS_15MetaInformationE [QtGui]
_ZNK13QTextDocument15objectForFormatERK11QTextFormat [QtGui]	_ZNK13QTextDocument16useDesignMetricsEv [QtGui]
_ZNK13QTextDocument17defaultStyleSheetEv [QtXml]	_ZNK13QTextDocument17isUndoRedoEnabledEv [QtGui]
_ZNK13QTextDocument17maximumBlockCountEv [QtXml]	_ZNK13QTextDocument3endEv [QtGui]
_ZNK13QTextDocument4findERK7QRegExpRK11QTextCursor6QFlagsINS_8FindFlagEE [QtXml]	_ZNK13QTextDocument4findERK7QRegExpRK11QTextCursor6QFlagsINS_8FindFlagEE [QtXml]
_ZNK13QTextDocument4findERK7QStringRK11QTextCursor6QFlagsINS_8FindFlagEE [QtGui]	_ZNK13QTextDocument4findERK7QStringRK11QTextCursor6QFlagsINS_8FindFlagEE [QtGui]
_ZNK13QTextDocument4sizeEv [QtXml]	_ZNK13QTextDocument5beginEv [QtGui]
_ZNK13QTextDocument5cloneEP7QObject [QtGui]	_ZNK13QTextDocument5printEP8QPrinter [QtGui]
_ZNK13QTextDocument6objectEi [QtGui]	_ZNK13QTextDocument6toHtmlERK10QByteArray [QtGui]
_ZNK13QTextDocument7frameAtEi [LSB]	_ZNK13QTextDocument7isEmptyEv [QtGui]
_ZNK13QTextDocument8pageSizeEv [QtGui]	_ZNK13QTextDocument8resourceEiRK4QUrl [QtGui]
_ZNK13QTextDocument9docHandleEv [QtGui]	_ZNK13QTextDocument9findBlockEi [QtGui]
_ZNK13QTextDocument9pageCountEv [QtGui]	_ZNK13QTextDocument9rootFrameEv [QtGui]
_ZNK13QTextDocument9textWidthEv [QtXml]	_ZNK13QTextFragment10charFormatEv [QtGui]
_ZNK13QTextFragment15charFormatIndexEv [QtGui]	_ZNK13QTextFragment4textEv [QtGui]
_ZNK13QTextFragment6lengthEv [QtGui]	_ZNK13QTextFragment8containsEi [QtGui]
_ZNK13QTextFragment8positionEv [QtGui]	_ZNK14QTextTableCell10columnSpanEv [QtGui]
_ZNK14QTextTableCell12lastPositionEv [LSB]	_ZNK14QTextTableCell13firstPositionEv [LSB]
_ZNK14QTextTableCell18lastCursorPositionEv [QtGui]	_ZNK14QTextTableCell19firstCursorPositionEv [QtGui]
_ZNK14QTextTableCell3endEv [QtGui]	_ZNK14QTextTableCell3rowEv [QtGui]

_ZNK14QTextTableCell5beginEv [QtGui]	_ZNK14QTextTableCell6columnEv [QtGui]
_ZNK14QTextTableCell6formatEv [QtGui]	_ZNK14QTextTableCell7rowSpanEv [QtGui]
_ZNK15QTextBlockGroup10metaObjectEv [QtGui]	_ZNK15QTextBlockGroup9blockListEv [QtGui]
_ZNK15QTextCharFormat13fontUnderlineEv [QtXml]	_ZNK15QTextCharFormat4fontEv [QtGui]
_ZNK17QTextInlineObject11formatIndexEv [QtGui]	_ZNK17QTextInlineObject12textPositionEv [QtGui]
_ZNK17QTextInlineObject13textDirectionEv [QtGui]	_ZNK17QTextInlineObject4rectEv [QtGui]
_ZNK17QTextInlineObject5widthEv [QtGui]	_ZNK17QTextInlineObject6ascentEv [QtGui]
_ZNK17QTextInlineObject6formatEv [QtGui]	_ZNK17QTextInlineObject6heightEv [QtGui]
_ZNK17QTextInlineObject7descentEv [QtGui]	_ZNK18QSyntaxHighlighter10metaObjectEv [QtGui]
_ZNK18QSyntaxHighlighter17currentBlockStateEv [QtGui]	_ZNK18QSyntaxHighlighter18previousBlockStateEv [QtGui]
_ZNK18QSyntaxHighlighter20currentBlockUserDataEv [QtGui]	_ZNK18QSyntaxHighlighter6formatEi [QtGui]
_ZNK18QSyntaxHighlighter8documentEv [QtGui]	_ZNK21QTextDocumentFragment11toPlainTextEv [QtGui]
_ZNK21QTextDocumentFragment6toHtmlERK10QByteArray [QtXml]	_ZNK21QTextDocumentFragment6toHtmlEv [QtGui]
_ZNK21QTextDocumentFragment7isEmptyEv [QtGui]	_ZNK9QTextItem11renderFlagsEv [QtGui]
_ZNK9QTextItem4fontEv [QtGui]	_ZNK9QTextItem4textEv [QtGui]
_ZNK9QTextItem5widthEv [QtGui]	_ZNK9QTextItem6ascentEv [QtGui]
_ZNK9QTextItem7descentEv [QtGui]	_ZNK9QTextLine10textLengthEv [QtGui]
_ZNK9QTextLine15naturalTextRectEv [QtGui]	_ZNK9QTextLine16naturalTextWidthEv [QtGui]
_ZNK9QTextLine1xEv [QtGui]	_ZNK9QTextLine1yEv [QtGui]
_ZNK9QTextLine4drawEP8QPainterRK7QPointFPKN11QTextLayout11FormatRangeE [QtGui]	_ZNK9QTextLine4rectEv [QtGui]
_ZNK9QTextLine5widthEv [QtGui]	_ZNK9QTextLine6ascentEv [QtGui]
_ZNK9QTextLine6heightEv [QtGui]	_ZNK9QTextLine7descentEv [QtGui]

_ZNK9QTextLine8positionEv [QtXml]	_ZNK9QTextLine9cursorToXEPiNS_ 4EdgeE [QtGui]
_ZNK9QTextLine9textStartEv [QtGui]	_ZNK9QTextLine9xToCursorEdNS_ 14CursorPositionE [QtGui]
_ZNK9QTextList10itemNumberERK 10QTextBlock [QtGui]	_ZNK9QTextList10metaObjectEv [QtGui]
_ZNK9QTextList4itemEi [QtGui]	_ZNK9QTextList5countEv [QtGui]
_ZNK9QTextList8itemTextERK10QT extBlock [QtGui]	_ZlsR11QDataStreamRK11QTextFor mat [QtGui]
_ZlsR11QDataStreamRK11QTextLen gth [QtGui]	_ZrsR11QDataStreamR11QTextForm at [QtGui]
_ZrsR11QDataStreamR11QTextLengt h [QtGui]	

18.5.18 Qt4 Main Window and Application

18.5.18.1 Class data for QClipboard

The virtual table for the QClipboard class is described by Table 18-384

Table 18-384 Primary vtable for QClipboard

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QClipboard
vfunc[0]:	QClipboard::metaObject() const
vfunc[1]:	QClipboard::qt_metacast(char const*)
vfunc[2]:	QClipboard::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QClipboard::~QClipboard()
vfunc[4]:	QClipboard::~QClipboard()
vfunc[5]:	QClipboard::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QClipboard::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QClipboard class is described by Table 18-385

Table 18-385 typeinfo for QClipboard

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QClipboard
basetype:	typeinfo for QObject

18.5.18.2 Class data for QSessionManager

The virtual table for the QSessionManager class is described by Table 18-386

Table 18-386 Primary vtable for QSessionManager

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSessionManager
vfunc[0]:	QSessionManager::metaObject() const
vfunc[1]:	QSessionManager::qt_metacast(char const*)
vfunc[2]:	QSessionManager::qt_metacall(QMet aObject::Call, int, void**)
vfunc[3]:	QSessionManager::~~QSessionManag er()
vfunc[4]:	QSessionManager::~~QSessionManag er()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QSessionManager class is described by Table 18-387

Table 18-387 typeinfo for QSessionManager

Base Vtable	vtable for __cxxabiv1::__si_class_type_info

Name	typeinfo name for QSessionManager
basetype:	typeinfo for QObject

18.5.18.3 Class data for QApplication

The virtual table for the QApplication class is described by Table 18-388

Table 18-388 Primary vtable for QApplication

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QApplication
vfunc[0]:	QApplication::metaObject() const
vfunc[1]:	QApplication::qt_metacast(char const*)
vfunc[2]:	QApplication::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QApplication::~~QApplication()
vfunc[4]:	QApplication::~~QApplication()
vfunc[5]:	QApplication::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QApplication::notify(QObject*, QEvent*)
vfunc[13]:	QApplication::compressEvent(QEvent*, QObject*, QPostEventList*)
vfunc[14]:	QApplication::x11EventFilter(_XEvent*)
vfunc[15]:	QApplication::x11ClientMessage(QWidget*, _XEvent*, bool)
vfunc[16]:	QApplication::commitData(QSessionManager&)
vfunc[17]:	QApplication::saveState(QSessionManager&)

The Run Time Type Information for the QApplication class is described by Table 18-389

Table 18-389 typeinfo for QApplication

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QApplication
basetype:	typeinfo for QApplication

18.5.18.4 Class data for QAction

The virtual table for the QAction class is described by Table 18-390

Table 18-390 Primary vtable for QAction

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAction
vfunc[0]:	QAction::metaObject() const
vfunc[1]:	QAction::qt_metacast(char const*)
vfunc[2]:	QAction::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAction::~~QAction()
vfunc[4]:	QAction::~QAction()
vfunc[5]:	QAction::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QAction class is described by Table 18-391

Table 18-391 typeinfo for QAction

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAction
basetype:	typeinfo for QObject

18.5.18.5 Class data for QActionGroup

The virtual table for the QActionGroup class is described by Table 18-392

Table 18-392 Primary vtable for QActionGroup

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QActionGroup
vfunc[0]:	QActionGroup::metaObject() const
vfunc[1]:	QActionGroup::qt_metacast(char const*)
vfunc[2]:	QActionGroup::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QActionGroup::~QActionGroup()
vfunc[4]:	QActionGroup::~QActionGroup()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QActionGroup class is described by Table 18-393

Table 18-393 typeinfo for QActionGroup

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QActionGroup
basetype:	typeinfo for QObject

18.5.18.6 Class data for QWorkspace

The virtual table for the QWorkspace class is described by Table 18-394

Table 18-394 Primary vtable for QWorkspace

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWorkspace
vfunc[0]:	QWorkspace::metaObject() const
vfunc[1]:	QWorkspace::qt_metacast(char const*)

vfunc[2]:	QWorkspace::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QWorkspace::~QWorkspace()
vfunc[4]:	QWorkspace::~~QWorkspace()
vfunc[5]:	QWorkspace::event(QEvent*)
vfunc[6]:	QWorkspace::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QWorkspace::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWorkspace::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWorkspace::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)

vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWorkspace::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWorkspace::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWorkspace::showEvent(QShowEvent*)
vfunc[41]:	QWorkspace::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWorkspace::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)

vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QWorkspace class is described by Table 18-395

Table 18-395 typeinfo for QWorkspace

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWorkspace
basetype:	typeinfo for QWidget

18.5.18.7 Class data for QSizeGrip

The virtual table for the QSizeGrip class is described by Table 18-396

Table 18-396 Primary vtable for QSizeGrip

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSizeGrip
vfunc[0]:	QSizeGrip::metaObject() const
vfunc[1]:	QSizeGrip::qt_metacast(char const*)
vfunc[2]:	QSizeGrip::qt_metacall(QMetaObject ::Call, int, void**)
vfunc[3]:	QSizeGrip::~~QSizeGrip()
vfunc[4]:	QSizeGrip::~~QSizeGrip()
vfunc[5]:	QSizeGrip::event(QEvent*)
vfunc[6]:	QSizeGrip::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QSizeGrip::setVisible(bool)
vfunc[14]:	QSizeGrip::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const

vfunc[18]:	QSizeGrip::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QSizeGrip::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QSizeGrip::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)

vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QSizeGrip class is described by Table 18-397

Table 18-397 typeinfo for QSizeGrip

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSizeGrip
basetype:	typeinfo for QWidget

18.5.18.8 Class data for QStatusBar

The virtual table for the QStatusBar class is described by Table 18-398

Table 18-398 Primary vtable for QStatusBar

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStatusBar
vfunc[0]:	QStatusBar::metaObject() const
vfunc[1]:	QStatusBar::qt_metacast(char const*)
vfunc[2]:	QStatusBar::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QStatusBar::~~QStatusBar()
vfunc[4]:	QStatusBar::~~QStatusBar()

vfunc[5]:	QStatusBar::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QStatusBar::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)

vfunc[31]:	QStatusBar::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QStatusBar class is described by Table 18-399

Table 18-399 typeinfo for QStatusBar

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStatusBar
basetype:	typeinfo for QWidget

18.5.18.9 Class data for QToolBar

The virtual table for the QToolBar class is described by Table 18-400

Table 18-400 Primary vtable for QToolBar

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QToolBar
vfunc[0]:	QToolBar::metaObject() const
vfunc[1]:	QToolBar::qt_metacast(char const*)
vfunc[2]:	QToolBar::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QToolBar::~QToolBar()
vfunc[4]:	QToolBar::~QToolBar()
vfunc[5]:	QToolBar::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QToolBar::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)

vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QToolBar::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QToolBar::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QToolBar::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QToolBar::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const

vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QToolBar class is described by Table 18-401

Table 18-401 typeinfo for QToolBar

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QToolBar
basetype:	typeinfo for QWidget

18.5.18.10 Class data for QWidgetAction

The virtual table for the QWidgetAction class is described by Table 18-402

Table 18-402 Primary vtable for QWidgetAction

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWidgetAction
vfunc[0]:	QWidgetAction::metaObject() const
vfunc[1]:	QWidgetAction::qt_metacast(char const*)
vfunc[2]:	QWidgetAction::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QWidgetAction::~~QWidgetAction()
vfunc[4]:	QWidgetAction::~~QWidgetAction()
vfunc[5]:	QWidgetAction::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)

vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidgetAction::createWidget(QWidget*)
vfunc[13]:	QWidgetAction::deleteWidget(QWidget*)

The Run Time Type Information for the QWidgetAction class is described by Table 18-403

Table 18-403 typeinfo for QWidgetAction

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QWidgetAction
basetype:	typeinfo for QAction

18.5.18.11 Interfaces for Qt4 Main Window and Application

An LSB conforming implementation shall provide the generic functions for Qt4 Main Window and Application specified in Table 18-404, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-404 libQtGui Qt4 Main Window and Application Function Interfaces

_ZN10QClipboard11dataChangedEv [QtGui]	_ZN10QClipboard11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN10QClipboard11qt_metacastEPKc [QtGui]	_ZN10QClipboard11setMimeDataEP9QMimeDataNS_4ModeE [QtGui]
_ZN10QClipboard13connectNotifyEPKc [QtGui]	_ZN10QClipboard16selectionChangedEv [QtGui]
_ZN10QClipboard17findBufferChangedEv [QtXml]	_ZN10QClipboard5clearENS_4ModeE [QtGui]
_ZN10QClipboard5eventEP6QEvent [QtGui]	_ZN10QClipboard7changedENS_4ModeE [QtXml]
_ZN10QClipboard7setDataEP11QMimeSourceNS_4ModeE [QtGui]	_ZN10QClipboard7setTextERK7QStringNS_4ModeE [QtGui]
_ZN10QClipboard8setImageERK6QImageNS_4ModeE [QtGui]	_ZN10QClipboard9setPixmapERK7QPixmapNS_4ModeE [QtGui]
_ZN10QClipboardD0Ev [QtGui]	_ZN10QClipboardD1Ev [QtGui]

_ZN10QClipboardD2Ev [QtGui]	_ZN10QStatusBar10hideOrShowEv [QtGui]
_ZN10QStatusBar10paintEventEP11QPaintEvent [QtGui]	_ZN10QStatusBar11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN10QStatusBar11qt_metacastEPKc [QtGui]	_ZN10QStatusBar11resizeEventEP12QResizeEvent [QtGui]
_ZN10QStatusBar11showMessageERK7QStringi [QtGui]	_ZN10QStatusBar12clearMessageEv [QtGui]
_ZN10QStatusBar12insertWidgetEiP7QWidgeti [QtGui]	_ZN10QStatusBar12removeWidgetEiP7QWidgeti [QtGui]
_ZN10QStatusBar14messageChangedERK7QString [QtGui]	_ZN10QStatusBar18addPermanentWidgetEP7QWidgeti [QtGui]
_ZN10QStatusBar18setSizeGripEnabledEb [QtGui]	_ZN10QStatusBar21insertPermanentWidgetEiP7QWidgeti [QtGui]
_ZN10QStatusBar5eventEP6QEvent [QtGui]	_ZN10QStatusBar8reformatEv [QtGui]
_ZN10QStatusBar9addWidgetEP7QWidgeti [QtGui]	_ZN10QStatusBarC1EP7QWidget [QtGui]
_ZN10QStatusBarC1EP7QWidgetPKc [QtGui]	_ZN10QStatusBarC2EP7QWidget [QtGui]
_ZN10QStatusBarC2EP7QWidgetPKc [QtGui]	_ZN10QStatusBarD0Ev [QtGui]
_ZN10QStatusBarD1Ev [QtGui]	_ZN10QStatusBarD2Ev [QtGui]
_ZN10QWorkspace10childEventEP11QChildEvent [QtGui]	_ZN10QWorkspace10paintEventEP11QPaintEvent [QtGui]
_ZN10QWorkspace10wheelEventEP11QWheelEvent [QtGui]	_ZN10QWorkspace11changeEventEP6QEvent [QtGui]
_ZN10QWorkspace11eventFilterEP7QObjectP6QEvent [QtGui]	_ZN10QWorkspace11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN10QWorkspace11qt_metacastEPKc [QtGui]	_ZN10QWorkspace11resizeEventEP12QResizeEvent [QtGui]
_ZN10QWorkspace12arrangeIconsEv [QtGui]	_ZN10QWorkspace13setBackgroundERK6QBrush [QtGui]
_ZN10QWorkspace15closeAllWindowsEv [QtGui]	_ZN10QWorkspace15setActiveWindowEP7QWidget [QtGui]
_ZN10QWorkspace15windowActivatedEP7QWidget [QtGui]	_ZN10QWorkspace17closeActiveWindowEv [QtGui]
_ZN10QWorkspace18activateNextWindowEv [QtGui]	_ZN10QWorkspace20setScrollBarsEnabledEb [QtGui]
_ZN10QWorkspace22activatePreviousWindowEv [QtGui]	_ZN10QWorkspace25setPaletteBackgroundColorERK6QColor [QtGui]

_ZN10QWorkspace26setPaletteBackg roundPixmapERK7QPixmap [QtGui]	_ZN10QWorkspace4tileEv [QtGui]
_ZN10QWorkspace5eventEP6QEven t [QtGui]	_ZN10QWorkspace7cascadeEv [QtGui]
_ZN10QWorkspace9addWindowEP7 QWidget6QFlagsIN2Qt10WindowTy peEE [QtGui]	_ZN10QWorkspace9hideEventEP10 QHideEvent [QtGui]
_ZN10QWorkspace9showEventEP10 QShowEvent [QtGui]	_ZN10QWorkspaceC1EP7QWidget [QtGui]
_ZN10QWorkspaceC1EP7QWidgetP Kc [QtGui]	_ZN10QWorkspaceC2EP7QWidget [QtGui]
_ZN10QWorkspaceC2EP7QWidgetP Kc [QtGui]	_ZN10QWorkspaceD0Ev [QtGui]
_ZN10QWorkspaceD1Ev [QtGui]	_ZN10QWorkspaceD2Ev [QtGui]
_ZN12QActionGroup10setEnabledE b [QtGui]	_ZN12QActionGroup10setVisibleEb [QtGui]
_ZN12QActionGroup11qt_metacalle N11QMetaObject4CallEiPPv [QtGui]	_ZN12QActionGroup11qt_metacastE PKc [QtGui]
_ZN12QActionGroup12removeActio nEP7QAction [QtGui]	_ZN12QActionGroup12setExclusive Eb [QtGui]
_ZN12QActionGroup7hoveredEP7Q Action [QtGui]	_ZN12QActionGroup8selectedEP7Q Action [QtGui]
_ZN12QActionGroup9addActionEP7 QAction [QtGui]	_ZN12QActionGroup9addActionER K5QIconRK7QString [QtGui]
_ZN12QActionGroup9addActionER K7QString [QtGui]	_ZN12QActionGroup9triggeredEP7 QAction [QtGui]
_ZN12QActionGroupC1EP7QObject [QtGui]	_ZN12QActionGroupC2EP7QObject [QtGui]
_ZN12QActionGroupD0Ev [QtGui]	_ZN12QActionGroupD1Ev [QtGui]
_ZN12QActionGroupD2Ev [QtGui]	_ZN12QApplication10allWidgetsEv [QtGui]
_ZN12QApplication10commitDataE R15QSessionManager [QtGui]	_ZN12QApplication10mainWidgetE v [QtGui]
_ZN12QApplication10setPaletteERK 8QPalettePKc [QtGui]	_ZN12QApplication10topLevelAtER K6QPoint [QtGui]
_ZN12QApplication10windowIconE v [QtGui]	_ZN12QApplication11focusWidgetE v [QtGui]
_ZN12QApplication11fontMetricsEv [QtGui]	_ZN12QApplication11globalStrutEv [QtGui]
_ZN12QApplication11qt_metacalle N11QMetaObject4CallEiPPv [QtGui]	_ZN12QApplication11qt_metacastE PKc [QtGui]

_ZN12QApplication12activeWindowEv [QtGui]	_ZN12QApplication12focusChangedEP7QWidgetS1_ [QtGui]
_ZN12QApplication12mouseButtonsEv [QtGui]	_ZN12QApplication12setColorSpecEi [QtGui]
_ZN12QApplication13compressEventEP6QEventP7QObjectP14QPostEventList [QtGui]	_ZN12QApplication13setMainWidgetEP7QWidget [QtGui]
_ZN12QApplication13setStyleSheetERK7QString [QtXml]	_ZN12QApplication13setWindowIconERK5QIcon [QtGui]
_ZN12QApplication13startDragTimeEv [QtGui]	_ZN12QApplication14overrideCursorEv [QtGui]
_ZN12QApplication14setGlobalStrutERK5QSize [QtGui]	_ZN12QApplication14x11EventFilterEP7_XEvent [QtGui]
_ZN12QApplication15closeAllWindowsEv [QtGui]	_ZN12QApplication15cursorFlashTimeEv [QtGui]
_ZN12QApplication15isEffectEnabledEN2Qt8UIEffectE [QtGui]	_ZN12QApplication15layoutDirectionEv [QtGui]
_ZN12QApplication15setActiveWindowEP7QWidget [QtGui]	_ZN12QApplication15setInputContextEP13QInputContext [QtGui]
_ZN12QApplication15topLevelWidgetsEv [QtGui]	_ZN12QApplication15x11ProcessEventEP7_XEvent [QtGui]
_ZN12QApplication16lastWindowClosedEv [QtGui]	_ZN12QApplication16saveStateRequestER15QSessionManager [QtXml]
_ZN12QApplication16setEffectEnabledEN2Qt8UIEffectEb [QtGui]	_ZN12QApplication16setStartDragTimeEi [QtGui]
_ZN12QApplication16wheelScrollLinesEv [QtGui]	_ZN12QApplication16x11ClientMessageEP7QWidgetP7_XEventb [QtGui]
_ZN12QApplication17activeModalWidgetEv [QtGui]	_ZN12QApplication17activePopupWidgetEv [QtGui]
_ZN12QApplication17commitDataRequestER15QSessionManager [QtXml]	_ZN12QApplication17keyboardModifiersEv [QtGui]
_ZN12QApplication17setOverrideCursorERK7QCursor [QtGui]	_ZN12QApplication17startDragDistanceEv [QtGui]
_ZN12QApplication18setCursorFlashTimeEi [QtGui]	_ZN12QApplication18setLayoutDirectionEN2Qt15LayoutDirectionE [QtGui]
_ZN12QApplication19doubleClickIntervalEv [QtGui]	_ZN12QApplication19horizontalAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN12QApplication19keyboardInputLocaleEv [QtXml]	_ZN12QApplication19setWheelScrollLinesEi [QtGui]
_ZN12QApplication20changeOverrideCursorERK7QCursor [QtGui]	_ZN12QApplication20desktopSettingsAwareEv [QtGui]

_ZN12QApplication20setStartDragDistanceEi [QtGui]	_ZN12QApplication21keyboardInputIntervalEv [QtGui]
_ZN12QApplication21restoreOverrideCursorEv [QtGui]	_ZN12QApplication22keyboardInputDirectionEv [QtXml]
_ZN12QApplication22quitOnLastWindowClosedEv [QtGui]	_ZN12QApplication22setDoubleClickIntervalEi [QtGui]
_ZN12QApplication23setDesktopSettingsAwareEb [QtGui]	_ZN12QApplication24setKeyboardInputIntervalEi [QtGui]
_ZN12QApplication25setQuitOnLastWindowClosedEb [QtGui]	_ZN12QApplication4beepEv [QtGui]
_ZN12QApplication4execEv [QtGui]	_ZN12QApplication4fontEPK7QWidget [QtGui]
_ZN12QApplication4fontEPKc [QtXml]	_ZN12QApplication4fontEv [QtXml]
_ZN12QApplication4typeEv [QtGui]	_ZN12QApplication5eventEP6QEvent [QtGui]
_ZN12QApplication5styleEv [QtGui]	_ZN12QApplication5syncXEv [QtGui]
_ZN12QApplication6notifyEP7QObjectP6QEvent [QtGui]	_ZN12QApplication7aboutQtEv [QtGui]
_ZN12QApplication7desktopEv [QtGui]	_ZN12QApplication7paletteEPK7QWidget [QtGui]
_ZN12QApplication7paletteEPKc [QtGui]	_ZN12QApplication7paletteEv [QtGui]
_ZN12QApplication7setFontEPK5QFontPKc [QtGui]	_ZN12QApplication8setStyleEP6QStyle [QtGui]
_ZN12QApplication8setStyleERK7QString [QtGui]	_ZN12QApplication8widgetAtERK6QPoint [QtGui]
_ZN12QApplication9clipboardEv [QtGui]	_ZN12QApplication9colorSpecEv [QtGui]
_ZN12QApplication9saveStateER15QSessionManager [QtGui]	_ZN12QApplicationC1EP9_XDisplayRiPPcmm [QtGui]
_ZN12QApplicationC1EP9_XDisplayRiPPcmmi [QtXml]	_ZN12QApplicationC1EP9_XDisplaymm [QtGui]
_ZN12QApplicationC1EP9_XDisplaymmi [QtXml]	_ZN12QApplicationC1ERiPPc [QtGui]
_ZN12QApplicationC1ERiPPcNS_4TypeE [QtGui]	_ZN12QApplicationC1ERiPPcNS_4TypeEi [QtXml]
_ZN12QApplicationC1ERiPPcb [QtGui]	_ZN12QApplicationC1ERiPPcbi [QtXml]
_ZN12QApplicationC1ERiPPci [QtXml]	_ZN12QApplicationC2EP9_XDisplayRiPPcmm [QtGui]

_ZN12QApplicationC2EP9_XDisplay RiPPcmmi [QtXml]	_ZN12QApplicationC2EP9_XDisplay mm [QtGui]
_ZN12QApplicationC2EP9_XDisplay mmi [QtXml]	_ZN12QApplicationC2ERiPPc [QtGui]
_ZN12QApplicationC2ERiPPcNS_4T ypeE [QtGui]	_ZN12QApplicationC2ERiPPcNS_4T ypeEi [QtXml]
_ZN12QApplicationC2ERiPPcb [QtGui]	_ZN12QApplicationC2ERiPPcbi [QtXml]
_ZN12QApplicationC2ERiPPci [QtXml]	_ZN12QApplicationD0Ev [QtGui]
_ZN12QApplicationD1Ev [QtGui]	_ZN12QApplicationD2Ev [QtGui]
_ZN13QWidgetAction11qt_metacall EN11QMetaObject4CallEiPPv [QtXml]	_ZN13QWidgetAction11qt_metacast EPKc [QtXml]
_ZN13QWidgetAction12createWidge tEP7QWidget [QtXml]	_ZN13QWidgetAction12deleteWidge tEP7QWidget [QtXml]
_ZN13QWidgetAction13releaseWidg etEP7QWidget [QtXml]	_ZN13QWidgetAction13requestWid getEP7QWidget [QtXml]
_ZN13QWidgetAction16setDefaultW idgetEP7QWidget [QtXml]	_ZN13QWidgetAction5eventEP6QEv ent [QtXml]
_ZN13QWidgetActionC1EP7QObject [QtXml]	_ZN13QWidgetActionC2EP7QObject [QtXml]
_ZN13QWidgetActionD0Ev [QtXml]	_ZN13QWidgetActionD1Ev [QtXml]
_ZN13QWidgetActionD2Ev [QtXml]	_ZN15QSessionManager11qt_metaca llEN11QMetaObject4CallEiPPv [QtGui]
_ZN15QSessionManager11qt_metaca stEPKc [QtGui]	_ZN15QSessionManager13requestPh ase2Ev [QtGui]
_ZN15QSessionManager14setRestart HintENS_11RestartHintE [QtGui]	_ZN15QSessionManager17allowsInte ractionEv [QtGui]
_ZN15QSessionManager17setDiscar dCommandERK11QStringList [QtGui]	_ZN15QSessionManager17setRestart CommandERK11QStringList [QtGui]
_ZN15QSessionManager18setManag erPropertyERK7QStringRK11QString List [QtGui]	_ZN15QSessionManager18setManag erPropertyERK7QStringS2_ [QtGui]
_ZN15QSessionManager22allowsErr orInteractionEv [QtGui]	_ZN15QSessionManager6cancelEv [QtGui]
_ZN15QSessionManager7releaseEv [QtGui]	_ZN15QSessionManagerD0Ev [QtGui]
_ZN15QSessionManagerD1Ev [QtGui]	_ZN15QSessionManagerD2Ev [QtGui]

_ZN7QAction10setCheckedEb [QtGui]	_ZN7QAction10setEnabledEb [QtGui]
_ZN7QAction10setToolTipERK7QString [QtGui]	_ZN7QAction10setVisibleEb [QtGui]
_ZN7QAction11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN7QAction11qt_metacastEPKc [QtGui]
_ZN7QAction11setIconTextERK7QString [QtGui]	_ZN7QAction11setMenuRoleENS_8MenuRoleE [QtXml]
_ZN7QAction11setShortcutERK12QKeySequence [QtGui]	_ZN7QAction12setCheckableEb [QtGui]
_ZN7QAction12setSeparatorEb [QtGui]	_ZN7QAction12setShortcutsEN12QKeySequence11StandardKeyE [QtXml]
_ZN7QAction12setShortcutsERK5QListH12QKeySequenceE [QtXml]	_ZN7QAction12setStatusTipERK7QString [QtGui]
_ZN7QAction12setWhatsThisERK7QString [QtGui]	_ZN7QAction13setAutoRepeatEb [QtXml]
_ZN7QAction14setActionGroupEP12QActionGroup [QtGui]	_ZN7QAction14showStatusTextEP7QWidget [QtGui]
_ZN7QAction18setShortcutContextEN2Qt15ShortcutContextE [QtGui]	_ZN7QAction5eventEP6QEvent [QtGui]
_ZN7QAction6toggleEv [QtGui]	_ZN7QAction7changedEv [QtGui]
_ZN7QAction7hoveredEv [QtGui]	_ZN7QAction7setDataERK8QVariant [QtGui]
_ZN7QAction7setFontERK5QFont [QtGui]	_ZN7QAction7setIconERK5QIcon [QtGui]
_ZN7QAction7setMenuEP5QMenu [QtGui]	_ZN7QAction7setTextERK7QString [QtGui]
_ZN7QAction7toggledEb [QtGui]	_ZN7QAction8activateENS_11ActionEventE [QtGui]
_ZN7QAction9activatedEi [QtGui]	_ZN7QAction9triggeredEb [QtGui]
_ZN7QActionC1EP7QObject [QtGui]	_ZN7QActionC1EP7QObjectPKc [QtGui]
_ZN7QActionC1ERK5QIconRK7QStringP7QObject [QtGui]	_ZN7QActionC1ERK5QIconRK7QStringRK12QKeySequenceP7QObjectPKc [QtGui]
_ZN7QActionC1ERK7QStringP7QObject [QtGui]	_ZN7QActionC1ERK7QStringRK12QKeySequenceP7QObjectPKc [QtGui]
_ZN7QActionC2EP7QObject [QtGui]	_ZN7QActionC2EP7QObjectPKc [QtGui]
_ZN7QActionC2ERK5QIconRK7QStringP7QObject [QtGui]	_ZN7QActionC2ERK5QIconRK7QStringRK12QKeySequenceP7QObjectPKc [QtGui]

_ZN7QActionC2ERK7QStringP7QObject [QtGui]	_ZN7QActionC2ERK7QStringRK12QKeySequenceP7QObjectPKc [QtGui]
_ZN7QActionD0Ev [QtGui]	_ZN7QActionD1Ev [QtGui]
_ZN7QActionD2Ev [QtGui]	_ZN8QToolBar10childEventEP11QChildEvent [QtGui]
_ZN8QToolBar10paintEventEP11QPaintEvent [QtGui]	_ZN8QToolBar10setMovableEb [QtGui]
_ZN8QToolBar11actionEventEP12QActionEvent [QtGui]	_ZN8QToolBar11changeEventEP6QEvent [QtGui]
_ZN8QToolBar11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN8QToolBar11qt_metacastEPKc [QtGui]
_ZN8QToolBar11resizeEventEP12QRaiseEvent [QtGui]	_ZN8QToolBar11setIconSizeERK5QSize [QtGui]
_ZN8QToolBar12addSeparatorEv [QtGui]	_ZN8QToolBar12insertWidgetEP7QActionP7QWidget [QtGui]
_ZN8QToolBar14movableChangedEb [QtGui]	_ZN8QToolBar14setOrientationEN2Qt11OrientationE [QtGui]
_ZN8QToolBar15actionTriggeredEP7QAction [QtGui]	_ZN8QToolBar15iconSizeChangedERK5QSize [QtGui]
_ZN8QToolBar15insertSeparatorEP7QAction [QtGui]	_ZN8QToolBar15setAllowedAreasE6QFlagsIN2Qt11ToolBarAreaEE [QtGui]
_ZN8QToolBar18orientationChangedEN2Qt11OrientationE [QtGui]	_ZN8QToolBar18setToolButtonStyleEN2Qt15ToolButtonStyleE [QtGui]
_ZN8QToolBar19allowedAreasChangedE6QFlagsIN2Qt11ToolBarAreaEE [QtGui]	_ZN8QToolBar22toolButtonStyleChangedEN2Qt15ToolButtonStyleE [QtGui]
_ZN8QToolBar5clearEv [QtGui]	_ZN8QToolBar5eventEP6QEvent [QtGui]
_ZN8QToolBar9addActionERK5QIconRK7QString [QtGui]	_ZN8QToolBar9addActionERK5QIconRK7QStringPK7QObjectPKc [QtGui]
_ZN8QToolBar9addActionERK7QString [QtGui]	_ZN8QToolBar9addActionERK7QStringPK7QObjectPKc [QtGui]
_ZN8QToolBar9addWidgetEP7QWidget [QtGui]	_ZN8QToolBarC1EP7QWidget [QtGui]
_ZN8QToolBarC1EP7QWidgetPKc [QtGui]	_ZN8QToolBarC1ERK7QStringP7QWidget [QtGui]
_ZN8QToolBarC2EP7QWidget [QtGui]	_ZN8QToolBarC2EP7QWidgetPKc [QtGui]
_ZN8QToolBarC2ERK7QStringP7QWidget [QtGui]	_ZN8QToolBarD0Ev [QtGui]

_ZN8QToolBarD1Ev [QtGui]	_ZN8QToolBarD2Ev [QtGui]
_ZN9QSizeGrip10paintEventEP11QPaintEvent [QtGui]	_ZN9QSizeGrip10setVisibleEb [QtGui]
_ZN9QSizeGrip11eventFilterEP7QObjectP6QEvent [QtGui]	_ZN9QSizeGrip11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QSizeGrip11qt_metacastEPKc [QtGui]	_ZN9QSizeGrip14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN9QSizeGrip15mousePressEventEP11QMouseEvent [QtGui]	_ZN9QSizeGrip5eventEP6QEvent [QtGui]
_ZN9QSizeGripC1EP7QWidget [QtGui]	_ZN9QSizeGripC1EP7QWidgetPKc [QtGui]
_ZN9QSizeGripC2EP7QWidget [QtGui]	_ZN9QSizeGripC2EP7QWidgetPKc [QtGui]
_ZN9QSizeGripD0Ev [QtGui]	_ZN9QSizeGripD1Ev [QtGui]
_ZN9QSizeGripD2Ev [QtGui]	_ZNK10QClipboard10metaObjectEv [QtGui]
_ZNK10QClipboard13ownsClipboardEv [QtGui]	_ZNK10QClipboard13ownsSelectionEv [QtGui]
_ZNK10QClipboard14ownsFindBufferEv [QtGui]	_ZNK10QClipboard17supportsSelectionEv [QtGui]
_ZNK10QClipboard18supportsFindBufferEv [QtGui]	_ZNK10QClipboard4dataENS_4ModeE [QtGui]
_ZNK10QClipboard4textENS_4ModeE [QtGui]	_ZNK10QClipboard4textER7QStringNS_4ModeE [QtGui]
_ZNK10QClipboard5imageENS_4ModeE [QtGui]	_ZNK10QClipboard6pixmapENS_4ModeE [QtGui]
_ZNK10QClipboard8mimeDataENS_4ModeE [QtGui]	_ZNK10QStatusBar10metaObjectEv [QtGui]
_ZNK10QStatusBar14currentMessageEv [QtGui]	_ZNK10QStatusBar17isSizeGripEnabledEv [QtGui]
_ZNK10QWorkspace10backgroundEv [QtGui]	_ZNK10QWorkspace10metaObjectEv [QtGui]
_ZNK10QWorkspace10windowListENS_11WindowOrderE [QtGui]	_ZNK10QWorkspace12activeWindowEv [QtGui]
_ZNK10QWorkspace17scrollBarsEnabledEv [QtGui]	_ZNK10QWorkspace8sizeHintEv [QtGui]
_ZNK12QActionGroup10metaObjectEv [QtGui]	_ZNK12QActionGroup11isExclusiveEv [QtGui]
_ZNK12QActionGroup13checkedActionEv [QtGui]	_ZNK12QActionGroup7actionsEv [QtGui]
_ZNK12QActionGroup9isEnabledEv [QtGui]	_ZNK12QActionGroup9isVisibleEv [QtGui]

_ZNK12QApplication10metaObjectEv [QtGui]	_ZNK12QApplication10sessionKeyEv [QtGui]
_ZNK12QApplication10styleSheetEv [QtXml]	_ZNK12QApplication12inputContextEv [QtGui]
_ZNK12QApplication17isSessionRestoredEv [QtGui]	_ZNK12QApplication9sessionIdEv [QtGui]
_ZNK13QWidgetAction10metaObjectEv [QtXml]	_ZNK13QWidgetAction13defaultWidgetEv [QtXml]
_ZNK13QWidgetAction14createdWidgetsEv [QtXml]	_ZNK15QSessionManager10metaObjectEv [QtGui]
_ZNK15QSessionManager10sessionKeyEv [QtGui]	_ZNK15QSessionManager11restartHintEv [QtGui]
_ZNK15QSessionManager14discardCommandEv [QtGui]	_ZNK15QSessionManager14restartCommandEv [QtGui]
_ZNK15QSessionManager6handleEv [QtGui]	_ZNK15QSessionManager8isPhase2Ev [QtGui]
_ZNK15QSessionManager9sessionIdEv [QtGui]	_ZNK7QAction10autoRepeatEv [QtXml]
_ZNK7QAction10metaObjectEv [QtGui]	_ZNK7QAction11actionGroupEv [QtGui]
_ZNK7QAction11isCheckableEv [QtGui]	_ZNK7QAction11isSeparatorEv [QtGui]
_ZNK7QAction12parentWidgetEv [QtGui]	_ZNK7QAction15shortcutContextEv [QtGui]
_ZNK7QAction17associatedWidgetsEv [QtXml]	_ZNK7QAction4dataEv [QtGui]
_ZNK7QAction4fontEv [QtGui]	_ZNK7QAction4iconEv [QtGui]
_ZNK7QAction4menuEv [QtGui]	_ZNK7QAction4textEv [QtGui]
_ZNK7QAction7toolTipEv [QtGui]	_ZNK7QAction8iconTextEv [QtGui]
_ZNK7QAction8menuRoleEv [QtXml]	_ZNK7QAction8shortcutEv [QtGui]
_ZNK7QAction9isCheckedEv [QtGui]	_ZNK7QAction9isEnabledEv [QtGui]
_ZNK7QAction9isVisibleEv [QtGui]	_ZNK7QAction9shortcutsEv [QtXml]
_ZNK7QAction9statusTipEv [QtGui]	_ZNK7QAction9whatsThisEv [QtGui]
_ZNK8QToolBar10metaObjectEv [QtGui]	_ZNK8QToolBar11orientationEv [QtGui]
_ZNK8QToolBar12allowedAreasEv [QtGui]	_ZNK8QToolBar14actionGeometryEP7QAction [LSB]

_ZNK8QToolBar15toolButtonStyleEv [QtGui]	_ZNK8QToolBar15widgetForAction EP7QAction [QtXml]
_ZNK8QToolBar16toggleViewAction Ev [QtGui]	_ZNK8QToolBar8actionAtERK6QPoint [QtGui]
_ZNK8QToolBar8iconSizeEv [QtGui]	_ZNK8QToolBar9isMovableEv [QtGui]
_ZNK9QSizeGrip10metaObjectEv [QtGui]	_ZNK9QSizeGrip8sizeHintEv [QtGui]

18.5.19 Qt4 Menus

18.5.19.1 Class data for QMenu

The virtual table for the QMenu class is described by Table 18-405

Table 18-405 Primary vtable for QMenu

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMenu
vfunc[0]:	QMenu::metaObject() const
vfunc[1]:	QMenu::qt_metacast(char const*)
vfunc[2]:	QMenu::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMenu::~~QMenu()
vfunc[4]:	QMenu::~~QMenu()
vfunc[5]:	QMenu::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QMenu::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QMenu::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const

vfunc[18]:	QMenu::mousePressEvent(QMouseEvent*)
vfunc[19]:	QMenu::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QMenu::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QMenu::wheelEvent(QWheelEvent*)
vfunc[23]:	QMenu::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QMenu::enterEvent(QEvent*)
vfunc[28]:	QMenu::leaveEvent(QEvent*)
vfunc[29]:	QMenu::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QMenu::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QMenu::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QMenu::changeEvent(QEvent*)

vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QMenu class is described by Table 18-406

Table 18-406 typeinfo for QMenu

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMenu
basetype:	typeinfo for QWidget

18.5.19.2 Class data for QMenuItem

The virtual table for the QMenuItem class is described by Table 18-407

Table 18-407 Primary vtable for QMenuItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMenuItem
vfunc[0]:	QAction::metaObject() const
vfunc[1]:	QAction::qt_metacast(char const*)
vfunc[2]:	QAction::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QMenuItem::~~QMenuItem()
vfunc[4]:	NULL or QMenuItem::~~QMenuItem()
vfunc[5]:	QAction::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QMenuItem class is described by Table 18-408

Table 18-408 typeinfo for QMenuItem

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMenuItem
basetype:	typeinfo for QAction

18.5.19.3 Class data for QMenuBar

The virtual table for the QMenuBar class is described by Table 18-409

Table 18-409 Primary vtable for QMenuBar

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMenuBar
vfunc[0]:	QMenuBar::metaObject() const
vfunc[1]:	QMenuBar::qt_metacast(char const*)
vfunc[2]:	QMenuBar::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMenuBar::~~QMenuBar()
vfunc[4]:	QMenuBar::~~QMenuBar()
vfunc[5]:	QMenuBar::event(QEvent*)
vfunc[6]:	QMenuBar::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QMenuBar::sizeHint() const
vfunc[15]:	QMenuBar::minimumSizeHint() const
vfunc[16]:	QMenuBar::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QMenuBar::mousePressEvent(QMouse Event*)
vfunc[19]:	QMenuBar::mouseReleaseEvent(QM ouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QMenuBar::mouseMoveEvent(QMou seEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent *)
vfunc[23]:	QMenuBar::keyPressEvent(QKeyEve nt*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEve nt*)
vfunc[25]:	QMenuBar::focusInEvent(QFocusEve nt*)
vfunc[26]:	QMenuBar::focusOutEvent(QFocusE vent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QMenuBar::leaveEvent(QEvent*)
vfunc[29]:	QMenuBar::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QMenuBar::resizeEvent(QResizeEve nt*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QCont extMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QMenuBar::actionEvent(QActionEve nt*)

vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QMenuBar::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QMenuBar class is described by Table 18-410.

Table 18-410 typeinfo for QMenuBar

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMenuBar
basetype:	typeinfo for QWidget

18.5.19.4 Interfaces for Qt4 Menus

An LSB conforming implementation shall provide the generic functions for Qt4 Menus specified in Table 18-411, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-411 libQtGui - Qt4 Menus Function Interfaces

_ZN5QMenu10enterEventEP6QEvent [QtGui]	_ZN5QMenu10insertItemEP9QMenuItem [QtGui]
ZN5QMenu10insertMenuEP7QActionPS [QtGui]	_ZN5QMenu10leaveEventEP6QEvent [QtGui]
_ZN5QMenu10paintEventEP11QPaintEvent [QtGui]	_ZN5QMenu10timerEventEP11QTimerEvent [QtGui]
_ZN5QMenu10wheelEventEP11QWheelEvent [QtGui]	_ZN5QMenu11aboutToHideEv [QtGui]
_ZN5QMenu11aboutToShowEv [QtGui]	_ZN5QMenu11actionEventEP12QActionEvent [QtGui]
_ZN5QMenu11changeEventEP6QEvent [QtGui]	_ZN5QMenu11highlightedEi [QtGui]
_ZN5QMenu11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN5QMenu11qt_metacastEPKc [QtGui]
_ZN5QMenu12addSeparatorEv [QtGui]	_ZN5QMenu13keyPressEventEP9QKeyEvent [QtGui]
_ZN5QMenu14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN5QMenu14setNoReplayForEP7QWidget [QtGui]
_ZN5QMenu15hideTearOffMenuEv [QtGui]	_ZN5QMenu15insertSeparatorEP7QAction [QtGui]
_ZN5QMenu15insertSeparatorEi [QtGui]	_ZN5QMenu15mousePressEventEP11QMouseEvent [QtGui]
_ZN5QMenu15setActiveActionEP7QAction [QtGui]	_ZN5QMenu16setDefaultActionEP7QAction [QtGui]
_ZN5QMenu16setItemParameterEii [QtGui]	_ZN5QMenu17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN5QMenu17setTearOffEnabledEb [QtGui]	_ZN5QMenu18focusNextPrevChildEb [QtGui]
_ZN5QMenu24setSeparatorsCollapsibleEb [QtGui]	_ZN5QMenu4execE5QListIP7QActionERK6QPointS2_ [QtGui]
_ZN5QMenu4execERK6QPointP7QAction [QtGui]	_ZN5QMenu4execEv [QtGui]
_ZN5QMenu5clearEv [QtGui]	_ZN5QMenu5eventEP6QEvent [QtGui]
_ZN5QMenu5popupERK6QPointP7QAction [QtGui]	_ZN5QMenu5setIdEii [QtGui]
ZN5QMenu7addMenuEPS [QtGui]	_ZN5QMenu7addMenuERK5QIconRK7QString [QtGui]
_ZN5QMenu7addMenuERK7QString [QtGui]	_ZN5QMenu7hoveredEP7QAction [QtGui]

_ZN5QMenu7setIconERK5QIcon [QtGui]	_ZN5QMenu8setTitleERK7QString [QtGui]
_ZN5QMenu9activatedEi [QtGui]	_ZN5QMenu9addActionERK5QIconRK7QString [QtGui]
_ZN5QMenu9addActionERK5QIconRK7QStringPK7QObjectPKcRK12QKeySequence [QtGui]	_ZN5QMenu9addActionERK7QString [QtGui]
_ZN5QMenu9addActionERK7QStringPK7QObjectPKcRK12QKeySequence [QtGui]	_ZN5QMenu9findPopupEPS_Pi [QtGui]
_ZN5QMenu9hideEventEP10QHideEvent [QtGui]	_ZN5QMenu9insertAnyEPK5QIconPK7QStringPK7QObjectPKcPK12QKeySequencePKS_ii [QtGui]
_ZN5QMenu9triggeredEP7QAction [QtGui]	_ZN5QMenuC1EP7QWidget [QtGui]
_ZN5QMenuC1ERK7QStringP7QWidget [QtGui]	_ZN5QMenuC2EP7QWidget [QtGui]
_ZN5QMenuC2ERK7QStringP7QWidget [QtGui]	_ZN5QMenuD0Ev [QtGui]
_ZN5QMenuD1Ev [QtGui]	_ZN5QMenuD2Ev [QtGui]
_ZN8QMenuBar10insertMenuEP7QActionP5QMenu [QtGui]	_ZN8QMenuBar10leaveEventEP6QEvent [QtGui]
_ZN8QMenuBar10paintEventEP11QPaintEvent [QtGui]	_ZN8QMenuBar11actionEventEP12QActionEvent [QtGui]
_ZN8QMenuBar11changeEventEP6QEvent [QtGui]	_ZN8QMenuBar11eventFilterEP7QObjectP6QEvent [QtGui]
_ZN8QMenuBar11highlightedEi [QtGui]	_ZN8QMenuBar11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN8QMenuBar11qt_metacastEPKc [QtGui]	_ZN8QMenuBar11resizeEventEP12QResizeEvent [QtGui]
_ZN8QMenuBar12addSeparatorEv [QtGui]	_ZN8QMenuBar12focusInEventEP11QFocusEvent [QtGui]
_ZN8QMenuBar12setDefaultUpEb [QtGui]	_ZN8QMenuBar13focusOutEventEP11QFocusEvent [QtGui]
_ZN8QMenuBar13keyPressEventEP9QKeyEvent [QtGui]	_ZN8QMenuBar14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN8QMenuBar15insertSeparatorEP7QAction [QtGui]	_ZN8QMenuBar15insertSeparatorEi [QtGui]
_ZN8QMenuBar15mousePressEventEP11QMouseEvent [QtGui]	_ZN8QMenuBar15setActiveActionEP7QAction [QtGui]
_ZN8QMenuBar15setAutoGeometryEb [QtGui]	_ZN8QMenuBar15setCornerWidgetEP7QWidgetN2Qt6CornerE [LSB]

_ZN8QMenuBar16setItemParameterEii [QtGui]	_ZN8QMenuBar17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN8QMenuBar5clearEv [QtGui]	_ZN8QMenuBar5eventEP6QEvent [QtGui]
_ZN8QMenuBar7addMenuEP5QMenu [QtGui]	_ZN8QMenuBar7addMenuERK5QIconRK7QString [QtGui]
_ZN8QMenuBar7addMenuERK7QString [QtGui]	_ZN8QMenuBar7hoveredEP7QAction [QtGui]
_ZN8QMenuBar9activatedEi [QtGui]	_ZN8QMenuBar9addActionERK7QString [QtGui]
_ZN8QMenuBar9addActionERK7QStringPK7QObjectPKc [QtGui]	_ZN8QMenuBar9triggeredEP7QAction [QtGui]
_ZN8QMenuBarC1EP7QWidget [QtGui]	_ZN8QMenuBarC1EP7QWidgetPKc [QtGui]
_ZN8QMenuBarC2EP7QWidget [QtGui]	_ZN8QMenuBarC2EP7QWidgetPKc [QtGui]
_ZN8QMenuBarD0Ev [QtGui]	_ZN8QMenuBarD1Ev [QtGui]
_ZN8QMenuBarD2Ev [QtGui]	_ZN9QMenuItem11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN9QMenuItem11qt_metacastEPKc [QtXml]	_ZN9QMenuItemC1Ev [QtGui]
_ZN9QMenuItemC2Ev [QtGui]	_ZNK5QMenu10frameWidthEv [QtGui]
_ZNK5QMenu10menuActionEv [QtGui]	_ZNK5QMenu10metaObjectEv [QtGui]
_ZNK5QMenu11columnCountEv [QtGui]	_ZNK5QMenu12activeActionEv [QtGui]
_ZNK5QMenu13defaultActionEv [QtGui]	_ZNK5QMenu13itemParameterEi [QtGui]
_ZNK5QMenu14actionGeometryEP7QAction [QtGui]	_ZNK5QMenu16isTearOffEnabledEv [QtGui]
_ZNK5QMenu20isTearOffMenuVisibleEv [QtGui]	_ZNK5QMenu21separatorsCollapsibleEv [QtXml]
_ZNK5QMenu4iconEv [QtGui]	_ZNK5QMenu5titleEv [QtGui]
_ZNK5QMenu7isEmptyEv [QtXml]	_ZNK5QMenu8actionAtERK6QPoint [QtGui]
_ZNK5QMenu8sizeHintEv [QtGui]	_ZNK8QMenuBar10frameWidthEv [QtGui]
_ZNK8QMenuBar10metaObjectEv [QtGui]	_ZNK8QMenuBar11isDefaultUpEv [QtGui]
_ZNK8QMenuBar12activeActionEv [QtGui]	_ZNK8QMenuBar12autoGeometryEv [QtGui]

_ZNK8QMenuBar12cornerWidgetE N2Qt6CornerE [LSB]	_ZNK8QMenuBar13itemParameterEi [QtGui]
_ZNK8QMenuBar14actionGeometry EP7QAction [LSB]	_ZNK8QMenuBar14heightForWidth Ei [QtGui]
_ZNK8QMenuBar15minimumSizeHi ntEv [QtGui]	_ZNK8QMenuBar8actionAtERK6QP oint [LSB]
_ZNK8QMenuBar8sizeHintEv [QtGui]	_ZNK9QMenuItem10metaObjectEv [QtXml]
_ZNK9QMenuItem11signalValueEv [QtGui]	_ZNK9QMenuItem2idEv [QtGui]

18.5.20 Qt4 Widgets

18.5.20.1 Class data for QWidget

The virtual table for the QWidget class is described by Table 18-412.

Table 18-412 Primary vtable for QWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QWidget
vfunc[0]:	QWidget::metaObject() const
vfunc[1]:	QWidget::qt_metacast(char const*)
vfunc[2]:	QWidget::qt_metacall(QMetaObject:: Call, int, void**)
vfunc[3]:	QWidget::~~QWidget()
vfunc[4]:	QWidget::~~QWidget()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const

vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)

vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QWidget class is described by Table 18-413

Table 18-413 typeinfo for QWidget

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QWidget	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QPaintDevice	2050

18.5.20.2 Class data for QFrame

The virtual table for the QFrame class is described by Table 18-414

Table 18-414 Primary vtable for QFrame

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFrame

vfunc[0]:	QFrame::metaObject() const
vfunc[1]:	QFrame::qt_metacast(char const*)
vfunc[2]:	QFrame::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFrame::~~QFrame()
vfunc[4]:	QFrame::~~QFrame()
vfunc[5]:	QFrame::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QFrame::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)

vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QFrame::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QFrame class is described by Table 18-415

Table 18-415 typeinfo for QFrame

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFrame
basetype:	typeinfo for QWidget

18.5.20.3 Class data for QLineEdit

The virtual table for the QLineEdit class is described by Table 18-416

Table 18-416 Primary vtable for QLineEdit

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QLineEdit
vfunc[0]:	QLineEdit::metaObject() const
vfunc[1]:	QLineEdit::qt_metacast(char const*)
vfunc[2]:	QLineEdit::qt_metacall(QMetaObject: :Call, int, void**)
vfunc[3]:	QLineEdit::~QLineEdit()
vfunc[4]:	QLineEdit::~~QLineEdit()
vfunc[5]:	QLineEdit::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QLineEdit::sizeHint() const
vfunc[15]:	QLineEdit::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QLineEdit::mousePressEvent(QMous eEvent*)

vfunc[19]:	QLineEdit::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QLineEdit::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QLineEdit::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QLineEdit::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QLineEdit::focusInEvent(QFocusEvent*)
vfunc[26]:	QLineEdit::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QLineEdit::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QLineEdit::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QLineEdit::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QLineEdit::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QLineEdit::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QLineEdit::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QLineEdit::changeEvent(QEvent*)

vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QLineEdit::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QLineEdit::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QLineEdit class is described by Table 18-417

Table 18-417 typeinfo for QLineEdit

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QLineEdit
basetype:	typeinfo for QWidget

18.5.20.4 Class data for QSlider

The virtual table for the QSlider class is described by Table 18-418

Table 18-418 Primary vtable for QSlider

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSlider
vfunc[0]:	QSlider::metaObject() const
vfunc[1]:	QSlider::qt_metacast(char const*)
vfunc[2]:	QSlider::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSlider::~~QSlider()
vfunc[4]:	QSlider::~~QSlider()
vfunc[5]:	QSlider::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSlider::timerEvent(QTimer Event*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QSlider::sizeHint() const
vfunc[15]:	QSlider::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QSlider::mousePressEvent(QMouseEvent*)
vfunc[19]:	QSlider::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QSlider::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractSlider::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSlider::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QSlider::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)

vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSlider::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractSlider::sliderChange(QAbstractSlider::SliderChange)

The Run Time Type Information for the QSlider class is described by Table 18-419

Table 18-419 typeinfo for QSlider

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSlider
basetype:	typeinfo for QAbstractSlider

18.5.20.5 Class data for QTabBar

The virtual table for the QTabBar class is described by Table 18-420

Table 18-420 Primary vtable for QTabBar

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTabBar
vfunc[0]:	QTabBar::metaObject() const
vfunc[1]:	QTabBar::qt_metacast(char const*)
vfunc[2]:	QTabBar::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTabBar::~~QTabBar()
vfunc[4]:	QTabBar::~QTabBar()
vfunc[5]:	QTabBar::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QTabBar::sizeHint() const
vfunc[15]:	QTabBar::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QTabBar::mousePressEvent(QMouseEvent*)
vfunc[19]:	QTabBar::mouseReleaseEvent(QMouseEvent*)

vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QTabBar::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QTabBar::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTabBar::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QTabBar::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QTabBar::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QTabBar::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const

vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QTabBar::tabSizeHint(int) const
vfunc[55]:	QTabBar::tabInserted(int)
vfunc[56]:	QTabBar::tabRemoved(int)
vfunc[57]:	QTabBar::tabLayoutChange()

The Run Time Type Information for the QTabBar class is described by Table 18-421

Table 18-421 typeinfo for QTabBar

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTabBar
basetype:	typeinfo for QWidget

18.5.20.6 Class data for QTabWidget

The virtual table for the QTabWidget class is described by Table 18-422

Table 18-422 Primary vtable for QTabWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTabWidget
vfunc[0]:	QTabWidget::metaObject() const
vfunc[1]:	QTabWidget::qt_metacast(char const*)
vfunc[2]:	QTabWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTabWidget::~QTabWidget()

vfunc[4]:	QTabWidget::~~QTabWidget()
vfunc[5]:	QTabWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QTabWidget::sizeHint() const
vfunc[15]:	QTabWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QTabWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTabWidget::paintEvent(QPaintEvent*)

vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QTabWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QTabWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QTabWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QTabWidget::tabInserted(int)
vfunc[55]:	QTabWidget::tabRemoved(int)

The Run Time Type Information for the QTabWidget class is described by Table 18-423

Table 18-423 typeinfo for QTabWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTabWidget
basetype:	typeinfo for QWidget

18.5.20.7 Class data for QListView

The virtual table for the QListView class is described by Table 18-424

Table 18-424 Primary vtable for QListView

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QListView
vfunc[0]:	QListView::metaObject() const
vfunc[1]:	QListView::qt_metacast(char const*)
vfunc[2]:	QListView::qt_metacall(QMetaObject ::Call, int, void**)
vfunc[3]:	QListView::~~QListView()
vfunc[4]:	QListView::~~QListView()
vfunc[5]:	QListView::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QListView::timerEvent(QTimerEvent *)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSize Hint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const

vfunc[18]:	QAbstractItemView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QListView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractItemView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QListView::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QListView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QListView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QListView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QListView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QListView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)

vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QListView::scrollContentsBy(int, int)
vfunc[56]:	QAbstractItemView::setModel(QAbstractItemModel*)
vfunc[57]:	QAbstractItemView::setSelectionMode(QItemSelectionModel*)
vfunc[58]:	QAbstractItemView::keyboardSearch(QString const&)
vfunc[59]:	QListView::visualRect(QModelIndex const&) const
vfunc[60]:	QListView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QListView::indexAt(QPoint const&) const
vfunc[62]:	QAbstractItemView::sizeHintForRow(int) const
vfunc[63]:	QAbstractItemView::sizeHintForColumn(int) const
vfunc[64]:	QListView::reset()

vfunc[65]:	QListView::setRootIndex(QModelIndex const&)
vfunc[66]:	QListView::doItemsLayout()
vfunc[67]:	QAbstractItemView::selectAll()
vfunc[68]:	QListView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QListView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QListView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)
vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QListView::updateGeometries()
vfunc[76]:	QAbstractItemView::verticalScrollbarAction(int)
vfunc[77]:	QAbstractItemView::horizontalScrollbarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollbarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollbarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*, QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(QWidget*)
vfunc[82]:	QAbstractItemView::editorDestroyed(QObject*)
vfunc[83]:	QListView::moveCursor(QAbstractItemView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QListView::horizontalOffset() const
vfunc[85]:	QListView::verticalOffset() const

vfunc[86]:	QListView::isIndexHidden(QModelIndex const&) const
vfunc[87]:	QListView::setSelection(QRect const&, QFlags<QItemSelectionModel::SelectionFlag>)
vfunc[88]:	QListView::visualRegionForSelection(QItemSelection const&) const
vfunc[89]:	QListView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelIndex const&, QAbstractItemView::EditTrigger, QEvent*)
vfunc[91]:	QAbstractItemView::selectionCommand(QModelIndex const&, QEvent const*) const
vfunc[92]:	QListView::startDrag(QFlags<Qt::DropAction>)
vfunc[93]:	QListView::viewOptions() const

The Run Time Type Information for the QListView class is described by Table 18-425

Table 18-425 typeinfo for QListView

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QListView
basetype:	typeinfo for QAbstractItemView

18.5.20.8 Class data for QListWidgetItem

The virtual table for the QListWidgetItem class is described by Table 18-426

Table 18-426 Primary vtable for QListWidgetItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QListWidgetItem
vfunc[0]:	QListWidgetItem::~~QListWidgetItem()
vfunc[1]:	QListWidgetItem::~~QListWidgetItem()
vfunc[2]:	QListWidgetItem::clone() const

vfunc[3]:	NULL or QListWidgetItem::setBackgroundCol or(QColor const&)
vfunc[4]:	QListWidgetItem::data(int) const
vfunc[5]:	QListWidgetItem::setData(int, QVariant const&)
vfunc[6]:	QListWidgetItem::operator<(QListW idgetItem const&) const
vfunc[7]:	QListWidgetItem::read(QDataStream &)
vfunc[8]:	QListWidgetItem::write(QDataStrea m&) const

The Run Time Type Information for the QListWidgetItem class is described by Table 18-427

Table 18-427 typeinfo for QListWidgetItem

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QListWidgetItem

18.5.20.9 Class data for QListWidget

The virtual table for the QListWidget class is described by Table 18-428

Table 18-428 Primary vtable for QListWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QListWidget
vfunc[0]:	QListWidget::metaObject() const
vfunc[1]:	QListWidget::qt_metacast(char const*)
vfunc[2]:	QListWidget::qt_metacall(QMetaObj ect::Call, int, void**)
vfunc[3]:	QListWidget::~~QListWidget()
vfunc[4]:	QListWidget::~~QListWidget()
vfunc[5]:	QListWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QListView::timerEvent(QTimerEvent *)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)

vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractItemView::mousePressEvent(QMouseEvent*)
vfunc[19]:	QListView::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractItemView::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QListView::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractItemView::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractItemView::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractItemView::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QListView::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QListView::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)

vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractItemView::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QListView::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QListView::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QListView::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractItemView::viewportEvent(QEvent*)
vfunc[55]:	QListView::scrollContentsBy(int, int)
vfunc[56]:	QListWidget::setModel(QAbstractItemModel*)
vfunc[57]:	QAbstractItemView::setSelectionMode(QItemSelectionModel*)
vfunc[58]:	QAbstractItemView::keyboardSearch(QString const&)
vfunc[59]:	QListView::visualRect(QModelIndex const&) const

vfunc[60]:	QListView::scrollTo(QModelIndex const&, QAbstractItemView::ScrollHint)
vfunc[61]:	QListView::indexAt(QPoint const&) const
vfunc[62]:	QAbstractItemView::sizeHintForRow(int) const
vfunc[63]:	QAbstractItemView::sizeHintForColumn(int) const
vfunc[64]:	QListView::reset()
vfunc[65]:	QListView::setRootIndex(QModelIndex const&)
vfunc[66]:	QListView::doItemsLayout()
vfunc[67]:	QAbstractItemView::selectAll()
vfunc[68]:	QListView::dataChanged(QModelIndex const&, QModelIndex const&)
vfunc[69]:	QListView::rowsInserted(QModelIndex const&, int, int)
vfunc[70]:	QListView::rowsAboutToBeRemoved(QModelIndex const&, int, int)
vfunc[71]:	QAbstractItemView::selectionChanged(QItemSelection const&, QItemSelection const&)
vfunc[72]:	QAbstractItemView::currentChanged(QModelIndex const&, QModelIndex const&)
vfunc[73]:	QAbstractItemView::updateEditorData()
vfunc[74]:	QAbstractItemView::updateEditorGeometries()
vfunc[75]:	QListView::updateGeometries()
vfunc[76]:	QAbstractItemView::verticalScrollBarAction(int)
vfunc[77]:	QAbstractItemView::horizontalScrollBarAction(int)
vfunc[78]:	QAbstractItemView::verticalScrollBarValueChanged(int)
vfunc[79]:	QAbstractItemView::horizontalScrollBarValueChanged(int)
vfunc[80]:	QAbstractItemView::closeEditor(QWidget*,

	QAbstractItemDelegate::EndEditHint)
vfunc[81]:	QAbstractItemView::commitData(Q Widget*)
vfunc[82]:	QAbstractItemView::editorDestroyed (QObject*)
vfunc[83]:	QListView::moveCursor(QAbstractIt emView::CursorAction, QFlags<Qt::KeyboardModifier>)
vfunc[84]:	QListView::horizontalOffset() const
vfunc[85]:	QListView::verticalOffset() const
vfunc[86]:	QListView::isIndexHidden(QModelI ndex const&) const
vfunc[87]:	QListView::setSelection(QRect const&, QFlags<QItemSelectionModel::Select ionFlag>)
vfunc[88]:	QListView::visualRegionForSelection (QItemSelection const&) const
vfunc[89]:	QListView::selectedIndexes() const
vfunc[90]:	QAbstractItemView::edit(QModelInd ex const&, QAbstractItemView::EditTrigger, QEvent*)
vfunc[91]:	QAbstractItemView::selectionComm and(QModelIndex const&, QEvent const*) const
vfunc[92]:	QListView::startDrag(QFlags<Qt::Dr opAction>)
vfunc[93]:	QListView::viewOptions() const
vfunc[94]:	QListWidget::mimeTypes() const
vfunc[95]:	QListWidget::mimeData(QList<QList WidgetItem*>) const
vfunc[96]:	QListWidget::dropMimeData(int, QMimeData const*, Qt::DropAction)
vfunc[97]:	QListWidget::supportedDropActions () const

The Run Time Type Information for the QListWidget class is described by Table 18-429

Table 18-429 typeinfo for QListWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info

Name	typeinfo name for QListWidget
basetype:	typeinfo for QListView

18.5.20.10 Class data for QDesktopWidget

The virtual table for the QDesktopWidget class is described by Table 18-430

Table 18-430 Primary vtable for QDesktopWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDesktopWidget
vfunc[0]:	QDesktopWidget::metaObject() const
vfunc[1]:	QDesktopWidget::qt_metacast(char const*)
vfunc[2]:	QDesktopWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDesktopWidget::~~QDesktopWidget()
vfunc[4]:	QDesktopWidget::~~QDesktopWidget()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)

vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDesktopWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const

vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QDesktopWidget class is described by Table 18-431

Table 18-431 typeinfo for QDesktopWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDesktopWidget
basetype:	typeinfo for QWidget

18.5.20.11 Class data for QInputContextFactoryInterface

The virtual table for the QInputContextFactoryInterface class is described by Table 18-432

Table 18-432 Primary vtable for QInputContextFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QInputContextFactoryInterface
vfunc[0]:	NULL or QInputContextFactoryInterface::~~QInputContextFactoryInterface()
vfunc[1]:	NULL or QInputContextFactoryInterface::~~QInputContextFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual

vfunc[6]:	__cxa_pure_virtual

The Run Time Type Information for the QInputContextFactoryInterface class is described by Table 18-433

Table 18-433 typeinfo for QInputContextFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QInputContextFactoryInterface
basetype:	typeinfo for QFactoryInterface

18.5.20.12 Class data for QInputContext

The virtual table for the QInputContext class is described by Table 18-434

Table 18-434 Primary vtable for QInputContext

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QInputContext
vfunc[0]:	QInputContext::metaObject() const
vfunc[1]:	QInputContext::qt_metacast(char const*)
vfunc[2]:	QInputContext::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QInputContext::~QInputContext()
vfunc[4]:	QInputContext::~QInputContext()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	QInputContext::update()

vfunc[16]:	QInputContext::mouseHandler(int, QMouseEvent*)
vfunc[17]:	QInputContext::font() const
vfunc[18]:	__cxa_pure_virtual
vfunc[19]:	QInputContext::setFocusWidget(QWidget*)
vfunc[20]:	QInputContext::widgetDestroyed(QWidget*)
vfunc[21]:	QInputContext::actions()
vfunc[22]:	QInputContext::x11FilterEvent(QWidget*, _XEvent*)
vfunc[23]:	QInputContext::filterEvent(QEvent const*)

The Run Time Type Information for the QInputContext class is described by Table 18-435

Table 18-435 typeinfo for QInputContext

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QInputContext
basetype:	typeinfo for QObject

18.5.20.13 Class data for QCheckBox

The virtual table for the QCheckBox class is described by Table 18-436

Table 18-436 Primary vtable for QCheckBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QCheckBox
vfunc[0]:	QCheckBox::metaObject() const
vfunc[1]:	QCheckBox::qt_metacast(char const*)
vfunc[2]:	QCheckBox::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QCheckBox::~~QCheckBox()
vfunc[4]:	NULL or QCheckBox::~~QCheckBox()
vfunc[5]:	QCheckBox::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractButton::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QCheckBox::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractButton::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractButton::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QCheckBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractButton::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractButton::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractButton::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractButton::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QCheckBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)

vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractButton::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QCheckBox::hitButton(QPoint const&) const
vfunc[55]:	QCheckBox::checkStateSet()
vfunc[56]:	QCheckBox::nextCheckState()

The Run Time Type Information for the QCheckBox class is described by Table 18-437

Table 18-437 typeinfo for QCheckBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QCheckBox
basetype:	typeinfo for QAbstractButton

18.5.20.14 Class data for QSpinBox

The virtual table for the QSpinBox class is described by Table 18-438

Table 18-438 Primary vtable for QSpinBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSpinBox
vfunc[0]:	QSpinBox::metaObject() const
vfunc[1]:	QSpinBox::qt_metacast(char const*)
vfunc[2]:	QSpinBox::qt_metacall(QMetaObject: :Call, int, void**)
vfunc[3]:	NULL or QSpinBox::~QSpinBox()
vfunc[4]:	NULL or QSpinBox::~QSpinBox()
vfunc[5]:	QSpinBox::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSpinBox::timerEvent(QTim erEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractSpinBox::sizeHint() const
vfunc[15]:	QAbstractSpinBox::minimumSizeHin t() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractSpinBox::mousePressEvent (QMouseEvent*)

vfunc[19]:	QAbstractSpinBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractSpinBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractSpinBox::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSpinBox::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractSpinBox::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractSpinBox::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractSpinBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractSpinBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractSpinBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QAbstractSpinBox::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractSpinBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QAbstractSpinBox::showEvent(QShowEvent*)

vfunc[41]:	QAbstractSpinBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSpinBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QSpinBox::validate(QString&, int&) const
vfunc[55]:	QSpinBox::fixup(QString&) const
vfunc[56]:	QAbstractSpinBox::stepBy(int)
vfunc[57]:	QAbstractSpinBox::clear()
vfunc[58]:	QAbstractSpinBox::stepEnabled() const
vfunc[59]:	QSpinBox::valueFromText(QString const&) const
vfunc[60]:	QSpinBox::textFromValue(int) const

The Run Time Type Information for the QSpinBox class is described by Table 18-439

Table 18-439 typeinfo for QSpinBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSpinBox
basetype:	typeinfo for QAbstractSpinBox

18.5.20.15 Class data for QDoubleSpinBox

The virtual table for the QDoubleSpinBox class is described by Table 18-440

Table 18-440 Primary vtable for QDoubleSpinBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDoubleSpinBox
vfunc[0]:	QDoubleSpinBox::metaObject() const
vfunc[1]:	QDoubleSpinBox::qt_metacast(char const*)
vfunc[2]:	QDoubleSpinBox::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QDoubleSpinBox::~~QDoubleSpinBox()
vfunc[4]:	NULL or QDoubleSpinBox::~~QDoubleSpinBox()
vfunc[5]:	QAbstractSpinBox::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSpinBox::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractSpinBox::sizeHint() const
vfunc[15]:	QAbstractSpinBox::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractSpinBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractSpinBox::mouseReleaseEvent(QMouseEvent*)

vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractSpinBox::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractSpinBox::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSpinBox::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractSpinBox::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractSpinBox::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractSpinBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractSpinBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QAbstractSpinBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QAbstractSpinBox::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractSpinBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QAbstractSpinBox::showEvent(QShowEvent*)
vfunc[41]:	QAbstractSpinBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)

vfunc[43]:	QAbstractSpinBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDoubleSpinBox::validate(QString&, int&) const
vfunc[55]:	QDoubleSpinBox::fixup(QString&) const
vfunc[56]:	QAbstractSpinBox::stepBy(int)
vfunc[57]:	QAbstractSpinBox::clear()
vfunc[58]:	QAbstractSpinBox::stepEnabled() const
vfunc[59]:	QDoubleSpinBox::valueFromText(QString const&) const
vfunc[60]:	QDoubleSpinBox::textFromValue(double) const

The Run Time Type Information for the QDoubleSpinBox class is described by Table 18-441

Table 18-441 typeinfo for QDoubleSpinBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDoubleSpinBox
basetype:	typeinfo for QAbstractSpinBox

18.5.20.16 Class data for QLCDNumber

The virtual table for the QLCDNumber class is described by Table 18-442

Table 18-442 Primary vtable for QLCDNumber

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QLCDNumber
vfunc[0]:	QLCDNumber::metaObject() const
vfunc[1]:	QLCDNumber::qt_metacast(char const*)
vfunc[2]:	QLCDNumber::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QLCDNumber::~~QLCDNumber()
vfunc[4]:	QLCDNumber::~~QLCDNumber()
vfunc[5]:	QLCDNumber::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QLCDNumber::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)

vfunc[24]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QLCDNumber::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)

vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QLCDNumber class is described by Table 18-443

Table 18-443 typeinfo for QLCDNumber

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QLCDNumber
basetype:	typeinfo for QFrame

18.5.20.17 Class data for QStackedWidget

The virtual table for the QStackedWidget class is described by Table 18-444

Table 18-444 Primary vtable for QStackedWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QStackedWidget
vfunc[0]:	QStackedWidget::metaObject() const
vfunc[1]:	QStackedWidget::qt_metacast(char const*)
vfunc[2]:	QStackedWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QStackedWidget::~~QStackedWidget()
vfunc[4]:	QStackedWidget::~~QStackedWidget()
vfunc[5]:	QStackedWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QFrame::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QFrame::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)

vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QStackedWidget class is described by Table 18-445

Table 18-445 typeinfo for QStackedWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QStackedWidget
basetype:	typeinfo for QFrame

18.5.20.18 Class data for QPushButton

The virtual table for the QPushButton class is described by Table 18-446

Table 18-446 Primary vtable for QPushButton

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QPushButton
vfunc[0]:	QPushButton::metaObject() const
vfunc[1]:	QPushButton::qt_metacast(char const*)
vfunc[2]:	QPushButton::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QPushButton::~~QPushButton()
vfunc[4]:	QPushButton::~~QPushButton()
vfunc[5]:	QPushButton::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractButton::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QPushButton::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractButton::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractButton::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractButton::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QPushButton::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractButton::keyReleaseEvent(QKeyEvent*)

vfunc[25]:	QPushButton::focusInEvent(QFocusEvent*)
vfunc[26]:	QPushButton::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QPushButton::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractButton::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)

vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractButton::hitButton(QPoint const&) const
vfunc[55]:	QAbstractButton::checkStateSet()
vfunc[56]:	QAbstractButton::nextCheckState()

The Run Time Type Information for the QPushButton class is described by Table 18-447

Table 18-447 typeinfo for QPushButton

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QPushButton
basetype:	typeinfo for QAbstractButton

18.5.20.19 Class data for QLabel

The virtual table for the QLabel class is described by Table 18-448

Table 18-448 Primary vtable for QLabel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QLabel
vfunc[0]:	QLabel::metaObject() const
vfunc[1]:	QLabel::qt_metacast(char const*)
vfunc[2]:	QLabel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QLabel::~~QLabel()
vfunc[4]:	QLabel::~~QLabel()
vfunc[5]:	QLabel::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QLabel::sizeHint() const
vfunc[15]:	QLabel::minimumSizeHint() const
vfunc[16]:	QLabel::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QLabel::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)

vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QLabel::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QLabel class is described by Table 18-449

Table 18-449 typeinfo for QLabel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QLabel
basetype:	typeinfo for QFrame

18.5.20.20 Class data for QDockWidget

The virtual table for the QDockWidget class is described by Table 18-450

Table 18-450 Primary vtable for QDockWidget

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QDockWidget
vfunc[0]:	QDockWidget::metaObject() const
vfunc[1]:	QDockWidget::qt_metacast(char const*)
vfunc[2]:	QDockWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDockWidget::~QDockWidget()
vfunc[4]:	QDockWidget::~QDockWidget()
vfunc[5]:	QDockWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)

vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QDockWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QDockWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QDockWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)

vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QDockWidget class is described by Table 18-451

Table 18-451 typeinfo for QDockWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDockWidget
basetype:	typeinfo for QWidget

18.5.20.21 Class data for QMainWindow

The virtual table for the QMainWindow class is described by Table 18-452

Table 18-452 Primary vtable for QMainWindow

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMainWindow
vfunc[0]:	QMainWindow::metaObject() const
vfunc[1]:	QMainWindow::qt_metacast(char const*)
vfunc[2]:	QMainWindow::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QMainWindow::~~QMainWindow()
vfunc[4]:	QMainWindow::~~QMainWindow()
vfunc[5]:	QMainWindow::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const

vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QMainWindow::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)

vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QMainWindow::createPopupMenu()

The Run Time Type Information for the QMainWindow class is described by Table 18-453

Table 18-453 typeinfo for QMainWindow

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QMainWindow
basetype:	typeinfo for QWidget

18.5.20.22 Class data for QTextEdit

The virtual table for the QTextEdit class is described by Table 18-454

Table 18-454 Primary vtable for QTextEdit

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextEdit

vfunc[0]:	QTextEdit::metaObject() const
vfunc[1]:	QTextEdit::qt_metacast(char const*)
vfunc[2]:	QTextEdit::qt_metacall(QMetaObject: :Call, int, void**)
vfunc[3]:	QTextEdit::~~QTextEdit()
vfunc[4]:	QTextEdit::~~QTextEdit()
vfunc[5]:	QTextEdit::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTextEdit::timerEvent(QTimerEvent *)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSize Hint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QTextEdit::mousePressEvent(QMous eEvent*)
vfunc[19]:	QTextEdit::mouseReleaseEvent(QMo useEvent*)
vfunc[20]:	QTextEdit::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QTextEdit::mouseMoveEvent(QMou seEvent*)
vfunc[22]:	QTextEdit::wheelEvent(QWheelEven t*)
vfunc[23]:	QTextEdit::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QTextEdit::focusInEvent(QFocusEve nt*)

vfunc[26]:	QTextEdit::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QTextEdit::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QTextEdit::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QTextEdit::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QTextEdit::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QTextEdit::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QTextEdit::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QTextEdit::dropEvent(QDropEvent*)
vfunc[40]:	QTextEdit::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QTextEdit::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QTextEdit::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QTextEdit::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QTextEdit::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)

vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractScrollArea::viewportEvent(QEvent*)
vfunc[55]:	QTextEdit::scrollContentsBy(int, int)
vfunc[56]:	QTextEdit::loadResource(int, QUrl const&)
vfunc[57]:	QTextEdit::createMimeDataFromSelection() const
vfunc[58]:	QTextEdit::canInsertFromMimeData(QMimeData const*) const
vfunc[59]:	QTextEdit::insertFromMimeData(QMimeData const*)

The Run Time Type Information for the QTextEdit class is described by Table 18-455

Table 18-455 typeinfo for QTextEdit

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextEdit
basetype:	typeinfo for QAbstractScrollArea

18.5.20.23 Class data for QProgressBar

The virtual table for the QProgressBar class is described by Table 18-456

Table 18-456 Primary vtable for QProgressBar

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QProgressBar
vfunc[0]:	QProgressBar::metaObject() const
vfunc[1]:	QProgressBar::qt_metacast(char const*)
vfunc[2]:	QProgressBar::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QProgressBar::~~QProgressBar()
vfunc[4]:	NULL or QProgressBar::~~QProgressBar()
vfunc[5]:	QProgressBar::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QProgressBar::sizeHint() const
vfunc[15]:	QProgressBar::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QProgressBar::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)

vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QProgressBar::text() const

The Run Time Type Information for the QProgressBar class is described by Table 18-457

Table 18-457 typeinfo for QProgressBar

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QProgressBar
basetype:	typeinfo for QWidget

18.5.20.24 Class data for QScrollBar

The virtual table for the QScrollBar class is described by Table 18-458

Table 18-458 Primary vtable for QScrollBar

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QScrollBar
vfunc[0]:	QScrollBar::metaObject() const
vfunc[1]:	QScrollBar::qt_metacast(char const*)
vfunc[2]:	QScrollBar::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QScrollBar::~QScrollBar()
vfunc[4]:	QScrollBar::~QScrollBar()
vfunc[5]:	QScrollBar::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSlider::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QScrollBar::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QScrollBar::mousePressEvent(QMouseEvent*)

vfunc[19]:	QScrollBar::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QScrollBar::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractSlider::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSlider::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QScrollBar::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QScrollBar::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QScrollBar::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)

vfunc[43]:	QAbstractSlider::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QScrollBar::sliderChange(QAbstractSlider::SliderChange)

The Run Time Type Information for the QScrollBar class is described by Table 18-459

Table 18-459 typeinfo for QScrollBar

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QScrollBar
basetype:	typeinfo for QAbstractSlider

18.5.20.25 Class data for QTextBrowser

The virtual table for the QTextBrowser class is described by Table 18-460

Table 18-460 Primary vtable for QTextBrowser

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTextBrowser
vfunc[0]:	QTextBrowser::metaObject() const
vfunc[1]:	QTextBrowser::qt_metacast(char const*)
vfunc[2]:	QTextBrowser::qt_metacall(QMetaObject::Call, int, void**)

vfunc[3]:	QTextBrowser::~~QTextBrowser()
vfunc[4]:	QTextBrowser::~~QTextBrowser()
vfunc[5]:	QTextBrowser::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QTextEdit::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QAbstractScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QTextBrowser::mousePressEvent(QMouseEvent*)
vfunc[19]:	QTextBrowser::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QTextEdit::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QTextBrowser::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QTextEdit::wheelEvent(QWheelEvent*)
vfunc[23]:	QTextBrowser::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QTextEdit::focusInEvent(QFocusEvent*)
vfunc[26]:	QTextBrowser::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)

vfunc[29]:	QTextBrowser::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QTextEdit::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QTextEdit::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QTextEdit::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QTextEdit::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QTextEdit::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QTextEdit::dropEvent(QDropEvent*)
vfunc[40]:	QTextEdit::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QTextEdit::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QTextEdit::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QTextEdit::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QTextBrowser::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

vfunc[54]:	QAbstractScrollArea::viewportEvent(QEvent*)
vfunc[55]:	QTextEdit::scrollContentsBy(int, int)
vfunc[56]:	QTextBrowser::loadResource(int, QUrl const&)
vfunc[57]:	QTextEdit::createMimeDataFromSelection() const
vfunc[58]:	QTextEdit::canInsertFromMimeData(QMimeData const*) const
vfunc[59]:	QTextEdit::insertFromMimeData(QMimeData const*)
vfunc[60]:	QTextBrowser::setSource(QUrl const&)
vfunc[61]:	QTextBrowser::backward()
vfunc[62]:	QTextBrowser::forward()
vfunc[63]:	QTextBrowser::home()
vfunc[64]:	QTextBrowser::reload()

The Run Time Type Information for the QTextBrowser class is described by Table 18-461

Table 18-461 typeinfo for QTextBrowser

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTextBrowser
basetype:	typeinfo for QTextEdit

18.5.20.26 Class data for QScrollArea

The virtual table for the QScrollArea class is described by Table 18-462

Table 18-462 Primary vtable for QScrollArea

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QScrollArea
vfunc[0]:	QScrollArea::metaObject() const
vfunc[1]:	QScrollArea::qt_metacast(char const*)
vfunc[2]:	QScrollArea::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QScrollArea::~QScrollArea()
vfunc[4]:	QScrollArea::~QScrollArea()

vfunc[5]:	QScrollArea::event(QEvent*)
vfunc[6]:	QScrollArea::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QScrollArea::sizeHint() const
vfunc[15]:	QAbstractScrollArea::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractScrollArea::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractScrollArea::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QAbstractScrollArea::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractScrollArea::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractScrollArea::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractScrollArea::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QAbstractScrollArea::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)

vfunc[31]:	QScrollArea::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QAbstractScrollArea::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QAbstractScrollArea::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QAbstractScrollArea::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QAbstractScrollArea::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QAbstractScrollArea::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QFrame::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QScrollArea::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractScrollArea::viewportEvent(QEvent*)

vfunc[55]:	QScrollArea::scrollContentsBy(int, int)

The Run Time Type Information for the QScrollArea class is described by Table 18-463

Table 18-463 typeinfo for QScrollArea

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QScrollArea
basetype:	typeinfo for QAbstractScrollArea

18.5.20.27 Class data for QToolButton

The virtual table for the QToolButton class is described by Table 18-464

Table 18-464 Primary vtable for QToolButton

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QToolButton
vfunc[0]:	QToolButton::metaObject() const
vfunc[1]:	QToolButton::qt_metacast(char const*)
vfunc[2]:	QToolButton::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QToolButton::~~QToolButton()
vfunc[4]:	QToolButton::~QToolButton()
vfunc[5]:	QToolButton::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QToolButton::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QToolButton::sizeHint() const

vfunc[15]:	QToolButton::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QToolButton::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractButton::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QAbstractButton::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractButton::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractButton::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractButton::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractButton::focusOutEvent(QFocusEvent*)
vfunc[27]:	QToolButton::enterEvent(QEvent*)
vfunc[28]:	QToolButton::leaveEvent(QEvent*)
vfunc[29]:	QToolButton::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QToolButton::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)

vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QToolButton::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QAbstractButton::hitButton(QPoint const&) const
vfunc[55]:	QAbstractButton::checkStateSet()
vfunc[56]:	QToolButton::nextCheckState()

The Run Time Type Information for the QToolButton class is described by Table 18-465.

Table 18-465 typeinfo for QToolButton

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QToolButton
basetype:	typeinfo for QAbstractButton

18.5.20.28 Class data for QComboBox

The virtual table for the QComboBox class is described by Table 18-466

Table 18-466 Primary vtable for QComboBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QComboBox
vfunc[0]:	QComboBox::metaObject() const
vfunc[1]:	QComboBox::qt_metacast(char const*)
vfunc[2]:	QComboBox::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QComboBox::~~QComboBox()
vfunc[4]:	QComboBox::~~QComboBox()
vfunc[5]:	QComboBox::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QComboBox::sizeHint() const
vfunc[15]:	QComboBox::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QComboBox::mousePressEvent(QMouseEvent*)
vfunc[19]:	QComboBox::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QComboBox::wheelEvent(QWheelEvent*)
vfunc[23]:	QComboBox::keyPressEvent(QKeyEvent*)

vfunc[24]:	QComboBox::keyPressEvent(QKeyEvent*)
vfunc[25]:	QComboBox::focusInEvent(QFocusEvent*)
vfunc[26]:	QComboBox::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QComboBox::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QComboBox::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QComboBox::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QComboBox::showEvent(QShowEvent*)
vfunc[41]:	QComboBox::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QComboBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QComboBox::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QComboBox::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)

vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QComboBox::showPopup()
vfunc[55]:	QComboBox::hidePopup()

The Run Time Type Information for the QComboBox class is described by Table 18-467

Table 18-467 typeinfo for QComboBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QComboBox
basetype:	typeinfo for QWidget

18.5.20.29 Class data for QRadioButton

The virtual table for the QRadioButton class is described by Table 18-468

Table 18-468 Primary vtable for QRadioButton

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QRadioButton
vfunc[0]:	QRadioButton::metaObject() const
vfunc[1]:	QRadioButton::qt_metacast(char const*)
vfunc[2]:	QRadioButton::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	NULL or QRadioButton::~~QRadioButton()
vfunc[4]:	NULL or QRadioButton::~~QRadioButton()
vfunc[5]:	QRadioButton::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractButton::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QRadioButton::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QAbstractButton::mousePressEvent(QMouseEvent*)
vfunc[19]:	QAbstractButton::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QRadioButton::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractButton::keyPressEvent(QKeyEvent*)
vfunc[24]:	QAbstractButton::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QAbstractButton::focusInEvent(QFocusEvent*)
vfunc[26]:	QAbstractButton::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QRadioButton::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)

vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractButton::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QRadioButton::hitButton(QPoint const&) const
vfunc[55]:	QAbstractButton::checkStateSet()
vfunc[56]:	QAbstractButton::nextCheckState()

The Run Time Type Information for the QRadioButton class is described by Table 18-469

Table 18-469 typeinfo for QRadioButton

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QRadioButton
basetype:	typeinfo for QAbstractButton

18.5.20.30 Class data for QFocusFrame

The virtual table for the QFocusFrame class is described by Table 18-470

Table 18-470 Primary vtable for QFocusFrame

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFocusFrame
vfunc[0]:	QFocusFrame::metaObject() const
vfunc[1]:	QFocusFrame::qt_metacast(char const*)
vfunc[2]:	QFocusFrame::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFocusFrame::~~QFocusFrame()
vfunc[4]:	QFocusFrame::~~QFocusFrame()
vfunc[5]:	QFocusFrame::event(QEvent*)
vfunc[6]:	QFocusFrame::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)

vfunc[19]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QFocusFrame::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)

vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QFocusFrame class is described by Table 18-471

Table 18-471 typeinfo for QFocusFrame

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFocusFrame
basetype:	typeinfo for QWidget

18.5.20.31 Class data for QToolBox

The virtual table for the QToolBox class is described by Table 18-472

Table 18-472 Primary vtable for QToolBox

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QToolBox
vfunc[0]:	QToolBox::metaObject() const
vfunc[1]:	QToolBox::qt_metacast(char const*)
vfunc[2]:	QToolBox::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QToolBox::~~QToolBox()
vfunc[4]:	QToolBox::~~QToolBox()
vfunc[5]:	QToolBox::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QFrame::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QFrame::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)

vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QToolBox::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QToolBox::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QToolBox::itemInserted(int)
vfunc[55]:	QToolBox::itemRemoved(int)

The Run Time Type Information for the QToolBox class is described by Table 18-473

Table 18-473 typeinfo for QToolBox

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QToolBox
basetype:	typeinfo for QFrame

18.5.20.32 Class data for QDial

The virtual table for the QDial class is described by Table 18-474

Table 18-474 Primary vtable for QDial

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDial
vfunc[0]:	QDial::metaObject() const
vfunc[1]:	QDial::qt_metacast(char const*)
vfunc[2]:	QDial::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDial::~~QDial()
vfunc[4]:	QDial::~~QDial()
vfunc[5]:	QDial::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QAbstractSlider::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QDial::sizeHint() const
vfunc[15]:	QDial::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QDial::mousePressEvent(QMouseEvent*)

vfunc[19]:	QDial::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QDial::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QAbstractSlider::wheelEvent(QWheelEvent*)
vfunc[23]:	QAbstractSlider::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QDial::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QDial::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QAbstractSlider::changeEvent(QEvent*)

vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QDial::sliderChange(QAbstractSlider::SliderChange)

The Run Time Type Information for the QDial class is described by Table 18-475

Table 18-475 typeinfo for QDial

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDial
basetype:	typeinfo for QAbstractSlider

18.5.20.33 Class data for QSystemTrayIcon

The virtual table for the QSystemTrayIcon class is described by Table 18-476

Table 18-476 Primary vtable for QSystemTrayIcon

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSystemTrayIcon
vfunc[0]:	QSystemTrayIcon::metaObject() const
vfunc[1]:	QSystemTrayIcon::qt_metacast(char const*)
vfunc[2]:	QSystemTrayIcon::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSystemTrayIcon::~QSystemTrayIcon()

vfunc[4]:	QSystemTrayIcon::~QSystemTrayIcon()
vfunc[5]:	QSystemTrayIcon::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

18.5.20.34 Interfaces for Qt4 Widgets

An LSB conforming implementation shall provide the generic functions for Qt4 Widgets specified in Table 18-477, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-477 libQtGui - Qt4 Widgets Function Interfaces

_ZN10QLCDNumber10paintEventEP11QPaintEvent [QtGui]	_ZN10QLCDNumber10setBinModeEv [QtGui]
_ZN10QLCDNumber10setDecModeEv [QtGui]	_ZN10QLCDNumber10setHexModeEv [QtGui]
_ZN10QLCDNumber10setOctModeEv [QtGui]	_ZN10QLCDNumber11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN10QLCDNumber11qt_metacastEPKc [QtGui]	_ZN10QLCDNumber12setNumDigitsEi [QtGui]
_ZN10QLCDNumber15setSegmentStyleENS_12SegmentStyleE [QtGui]	_ZN10QLCDNumber20setSmallDecimalPointEb [QtGui]
_ZN10QLCDNumber5eventEP6QEvent [QtGui]	_ZN10QLCDNumber7displayERK7QString [QtGui]
_ZN10QLCDNumber7displayEd [QtGui]	_ZN10QLCDNumber7displayEi [QtGui]
_ZN10QLCDNumber7setModeENS_4ModeE [QtGui]	_ZN10QLCDNumber8overflowEv [QtGui]
_ZN10QLCDNumberC1EP7QWidget [QtGui]	_ZN10QLCDNumberC1EP7QWidgetPKc [QtGui]
_ZN10QLCDNumberC1EjP7Widget [QtGui]	_ZN10QLCDNumberC1EjP7WidgettPKc [QtGui]
_ZN10QLCDNumberC2EP7QWidget [QtGui]	_ZN10QLCDNumberC2EP7QWidgetPKc [QtGui]
_ZN10QLCDNumberC2EjP7Widget [QtGui]	_ZN10QLCDNumberC2EjP7WidgettPKc [QtGui]

_ZN10QLCDNumberD0Ev [QtGui]	_ZN10QLCDNumberD1Ev [QtGui]
_ZN10QLCDNumberD2Ev [QtGui]	_ZN10QScrollBar10paintEventEP11QPaintEvent [QtGui]
_ZN10QScrollBar11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN10QScrollBar11qt_metacastEPKc [QtGui]
_ZN10QScrollBar12sliderChangeEventEN15QAbstractSlider12SliderChangeEvent [QtGui]	_ZN10QScrollBar14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN10QScrollBar15mousePressEventEP11QMouseEvent [QtGui]	_ZN10QScrollBar16contextMenuEventEP17QContextMenuEvent [QtGui]
_ZN10QScrollBar17mouseReleaseEventEP11QMouseEvent [QtGui]	_ZN10QScrollBar5eventEP6QEvent [QtGui]
_ZN10QScrollBar9hideEventEP10QHideEvent [QtGui]	_ZN10QScrollBarC1EN2Qt11OrientationEP7QWidget [QtGui]
_ZN10QScrollBarC1EN2Qt11OrientationEP7QWidgetPKc [QtGui]	_ZN10QScrollBarC1EP7QWidget [QtGui]
_ZN10QScrollBarC1EP7QWidgetPKc [QtGui]	_ZN10QScrollBarC1EiiiiN2Qt11OrientationEP7QWidgetPKc [QtGui]
_ZN10QScrollBarC2EN2Qt11OrientationEP7QWidget [QtGui]	_ZN10QScrollBarC2EN2Qt11OrientationEP7QWidgetPKc [QtGui]
_ZN10QScrollBarC2EP7QWidget [QtGui]	_ZN10QScrollBarC2EP7QWidgetPKc [QtGui]
_ZN10QScrollBarC2EiiiiN2Qt11OrientationEP7QWidgetPKc [QtGui]	_ZN10QScrollBarD0Ev [QtGui]
_ZN10QScrollBarD1Ev [QtGui]	_ZN10QScrollBarD2Ev [QtGui]
_ZN10QTabWidget10paintEventEP11QPaintEvent [QtGui]	_ZN10QTabWidget10setTabIconEiRK5QIcon [QtGui]
_ZN10QTabWidget10setTabTextEiRK7QString [QtGui]	_ZN10QTabWidget10tabRemovedEi [QtGui]
_ZN10QTabWidget11changeEventEP6QEvent [QtGui]	_ZN10QTabWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN10QTabWidget11qt_metacastEPKc [QtGui]	_ZN10QTabWidget11resizeEventEP12QResizeEvent [QtGui]
_ZN10QTabWidget11setIconSizeERK5QSize [QtXml]	_ZN10QTabWidget11setTabShapeENS_8TabShapeE [QtGui]
_ZN10QTabWidget11tabInsertedEi [QtGui]	_ZN10QTabWidget12setElideModeEN2Qt13TextElideModeE [QtXml]
_ZN10QTabWidget13keyPressEventEP9QKeyEvent [QtGui]	_ZN10QTabWidget13setTabEnabledEib [QtGui]
_ZN10QTabWidget13setTabToolTipEiRK7QString [QtGui]	_ZN10QTabWidget14currentChangedEP7QWidget [QtGui]

_ZN10QTabWidget14currentChange dEi [QtGui]	_ZN10QTabWidget14setTabPosition ENS_11TabPositionE [QtGui]
_ZN10QTabWidget15setCornerWidg etEP7QWidgetN2Qt6CornerE [QtGui]	_ZN10QTabWidget15setCurrentInde xEi [QtGui]
_ZN10QTabWidget15setTabWhatsTh isEiRK7QString [QtGui]	_ZN10QTabWidget16setCurrentWid getEP7QWidget [QtGui]
_ZN10QTabWidget20setUsesScrollB uttonsEb [QtXml]	_ZN10QTabWidget5eventEP6QEven t [QtGui]
_ZN10QTabWidget6addTabEP7QWi dgetRK5QIconRK7QString [QtGui]	_ZN10QTabWidget6addTabEP7QWi dgetRK7QString [QtGui]
_ZN10QTabWidget8selectedERK7QS tring [QtXml]	_ZN10QTabWidget9insertTabEiP7Q WidgetRK5QIconRK7QString [QtGui]
_ZN10QTabWidget9insertTabEiP7Q WidgetRK7QString [QtGui]	_ZN10QTabWidget9removeTabEi [QtGui]
_ZN10QTabWidget9setTabBarEP7Q TabBar [QtGui]	_ZN10QTabWidget9showEventEP10 QShowEvent [QtGui]
_ZN10QTabWidgetC1EP7QWidget [QtGui]	_ZN10QTabWidgetC1EP7QWidgetP Kc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN10QTabWidgetC2EP7QWidget [QtGui]	_ZN10QTabWidgetC2EP7QWidgetP Kc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN10QTabWidgetD0Ev [QtGui]	_ZN10QTabWidgetD1Ev [QtGui]
_ZN10QTabWidgetD2Ev [QtGui]	_ZN10QWhatsThis12createActionEP 7QObject [QtGui]
_ZN10QWhatsThis15inWhatsThisMo deEv [QtGui]	_ZN10QWhatsThis15whatsThisButto nEP7QWidget [QtGui]
_ZN10QWhatsThis18enterWhatsThis ModeEv [QtGui]	_ZN10QWhatsThis18leaveWhatsThis ModeEv [QtGui]
_ZN10QWhatsThis3addEP7QWidget RK7QString [QtGui]	_ZN10QWhatsThis6removeEP7QWi dget [QtGui]
_ZN10QWhatsThis8hideTextEv [QtGui]	_ZN10QWhatsThis8showTextERK6Q PointRK7QStringP7QWidget [QtGui]
_ZN11QDockWidget10closeEventEP 11QCloseEvent [QtGui]	_ZN11QDockWidget10paintEventEP 11QPaintEvent [QtGui]
_ZN11QDockWidget11changeEvent EP6QEvent [QtGui]	_ZN11QDockWidget11qt_metacalle N11QMetaObject4CallEiPPv [QtGui]
_ZN11QDockWidget11qt_metacastE PKc [QtGui]	_ZN11QDockWidget11setFeaturesE6 QFlagsINS_17DockWidgetFeatureEE [QtGui]

_ZN11QDockWidget11setFloatingEb [QtGui]	_ZN11QDockWidget15featuresChangedE6QFlagsINS_17DockWidgetFeatureEE [QtGui]
_ZN11QDockWidget15setAllowedAreasE6QFlagsINS2Qt14DockWidgetAreaEE [QtGui]	_ZN11QDockWidget15topLevelChangedEb [QtGui]
_ZN11QDockWidget19allowedAreasChangedE6QFlagsINS2Qt14DockWidgetAreaEE [QtGui]	_ZN11QDockWidget5eventEP6QEvent [QtGui]
_ZN11QDockWidget9setWidgetEP7QWidget [QtGui]	_ZN11QDockWidgetC1EP7QWidget6QFlagsINS2Qt10WindowTypeEE [QtGui]
_ZN11QDockWidgetC1ERK7QStringP7QWidget6QFlagsINS2Qt10WindowTypeEE [QtGui]	_ZN11QDockWidgetC2EP7QWidget6QFlagsINS2Qt10WindowTypeEE [QtGui]
_ZN11QDockWidgetC2ERK7QStringP7QWidget6QFlagsINS2Qt10WindowTypeEE [QtGui]	_ZN11QDockWidgetD0Ev [QtGui]
_ZN11QDockWidgetD1Ev [QtGui]	_ZN11QDockWidgetD2Ev [QtGui]
_ZN11QFocusFrame10paintEventEP11QPaintEvent [QtGui]	_ZN11QFocusFrame11eventFilterEP7QObjectP6QEvent [QtGui]
_ZN11QFocusFrame11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QFocusFrame11qt_metacastEPKc [QtGui]
_ZN11QFocusFrame5eventEP6QEvent [QtGui]	_ZN11QFocusFrame9setWidgetEP7QWidget [QtGui]
_ZN11QFocusFrameC1EP7QWidget [QtGui]	_ZN11QFocusFrameC2EP7QWidget [QtGui]
_ZN11QFocusFrameD0Ev [QtGui]	_ZN11QFocusFrameD1Ev [QtGui]
_ZN11QFocusFrameD2Ev [QtGui]	_ZN11QListWidget10insertItemEiP15QListWidgetItem [QtGui]
_ZN11QListWidget10insertItemEiRK7QString [QtGui]	_ZN11QListWidget11insertItemsEiRK11QStringList [QtGui]
_ZN11QListWidget11itemChangedEP15QListWidgetItem [QtGui]	_ZN11QListWidget11itemClickedEP15QListWidgetItem [QtGui]
_ZN11QListWidget11itemEnteredEP15QListWidgetItem [QtGui]	_ZN11QListWidget11itemPressedEP15QListWidgetItem [QtGui]
_ZN11QListWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QListWidget11qt_metacastEPKc [QtGui]
_ZN11QListWidget12dropMimeDataEiPK9QMimeDataN2Qt10DropActionE [QtGui]	_ZN11QListWidget12scrollToItemEPK15QListWidgetItemN17QAbstractItemView10ScrollHintE [QtGui]
_ZN11QListWidget13itemActivatedEP15QListWidgetItem [QtGui]	_ZN11QListWidget13setCurrentRowEi [QtGui]

_ZN11QListWidget13setItemHiddenEPK15QListWidgetItem [QtGui]	_ZN11QListWidget13setItemWidgetEP15QListWidgetItemP7QWidget [QtGui]
_ZN11QListWidget14setCurrentItemEP15QListWidgetItem [QtGui]	_ZN11QListWidget15setItemSelectedEPK15QListWidgetItem [QtGui]
_ZN11QListWidget17currentRowChangedEi [QtGui]	_ZN11QListWidget17itemDoubleClickedEP15QListWidgetItem [QtGui]
_ZN11QListWidget17setSortingEnabledEb [QtXml]	_ZN11QListWidget18currentItemChangedEP15QListWidgetItemS1_ [QtGui]
_ZN11QListWidget18currentTextChangedERK7QString [QtGui]	_ZN11QListWidget20itemSelectionChangedEv [QtGui]
_ZN11QListWidget20openPersistentEditorEP15QListWidgetItem [QtGui]	_ZN11QListWidget21closePersistentEditorEP15QListWidgetItem [QtGui]
_ZN11QListWidget5clearEv [QtGui]	_ZN11QListWidget5eventEP6QEvent [QtGui]
_ZN11QListWidget8editItemEP15QListWidgetItem [QtGui]	_ZN11QListWidget8setModelEP18QAbstractItemModel [QtGui]
_ZN11QListWidget8takeItemEi [QtGui]	_ZN11QListWidget9dropEventEP10QDropEvent [QtXml]
_ZN11QListWidget9sortItemsEN2Qt9SortOrderE [QtGui]	_ZN11QListWidgetC1EP7QWidget [QtGui]
_ZN11QListWidgetC2EP7QWidget [QtGui]	_ZN11QListWidgetD0Ev [QtGui]
_ZN11QListWidgetD1Ev [QtGui]	_ZN11QListWidgetD2Ev [QtGui]
_ZN11QMainWindow10addToolBarEN2Qt11ToolBarAreaEP8QToolBar [QtGui]	_ZN11QMainWindow10addToolBarEP8QToolBar [QtGui]
_ZN11QMainWindow10addToolBarERK7QString [QtGui]	_ZN11QMainWindow10setMenuBarEP8QMenuBar [QtGui]
_ZN11QMainWindow11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QMainWindow11qt_metacastEPKc [QtGui]
_ZN11QMainWindow11setAnimatedEb [QtXml]	_ZN11QMainWindow11setIconSizeERK5QSize [QtGui]
_ZN11QMainWindow12restoreStateERK10QByteArrayi [QtGui]	_ZN11QMainWindow12setStatusBarEP10QStatusBar [QtGui]
_ZN11QMainWindow13addDockWidgetEN2Qt14DockWidgetAreaEP11QDockWidget [QtGui]	_ZN11QMainWindow13addDockWidgetEN2Qt14DockWidgetAreaEP11QDockWidgetNS0_11OrientationE [QtGui]
ZN11QMainWindow13insertToolBarEP8QToolBarS1 [QtGui]	_ZN11QMainWindow13removeToolBarEP8QToolBar [QtGui]

_ZN11QMainWindow13setMenuWidgetEP7QWidget [QtXml]	_ZN11QMainWindow15addToolBarBreakEN2Qt11ToolBarAreaE [QtGui]
_ZN11QMainWindow15createPopupMenuEv [QtGui]	_ZN11QMainWindow15iconSizeChangedERK5QSize [QtGui]
_ZN11QMainWindow15splitDockWidgetEP11QDockWidgetS1_N2Qt11OrientationE [QtGui]	_ZN11QMainWindow16contextMenuEventEP17QContextMenuEvent [QtGui]
_ZN11QMainWindow16removeDockWidgetEP11QDockWidget [QtGui]	_ZN11QMainWindow16setCentralWidgetEP7QWidget [QtGui]
ZN11QMainWindow16tabifyDockWidgetEP11QDockWidgetS1 [QtXml]	_ZN11QMainWindow18insertToolBarBreakEP8QToolBar [QtGui]
_ZN11QMainWindow18setToolButtonStyleEN2Qt15ToolButtonStyleE [QtGui]	_ZN11QMainWindow21setDockNestingEnabledEb [QtXml]
_ZN11QMainWindow22toolButtonStyleChangedEN2Qt15ToolButtonStyleE [QtGui]	_ZN11QMainWindow5eventEP6QEvent [QtGui]
_ZN11QMainWindow9setCornerEN2Qt6CornerENS0_14DockWidgetAreaE [QtGui]	_ZN11QMainWindowC1EP7QWidgett6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN11QMainWindowC1EP7QWidgett6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN11QMainWindowC2EP7QWidgett6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN11QMainWindowC2EP7QWidgett6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN11QMainWindowD0Ev [QtGui]
_ZN11QMainWindowD1Ev [QtGui]	_ZN11QMainWindowD2Ev [QtGui]
_ZN11QPushButton10paintEventEP11QPaintEvent [QtGui]	_ZN11QPushButton10setDefaultEb [QtGui]
_ZN11QPushButton11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QPushButton11qt_metacastEPKc [QtGui]
_ZN11QPushButton12focusInEventEP11QFocusEvent [QtGui]	_ZN11QPushButton13focusOutEventEP11QFocusEvent [QtGui]
_ZN11QPushButton13keyPressEventEP9QKeyEvent [QtGui]	_ZN11QPushButton14setAutoDefaultEb [QtGui]
_ZN11QPushButton5eventEP6QEvent [QtGui]	_ZN11QPushButton7setFlatEb [QtGui]
_ZN11QPushButton7setMenuEP5QMenu [QtGui]	_ZN11QPushButton8showMenuEv [QtGui]
_ZN11QPushButtonC1EP7QWidget [QtGui]	_ZN11QPushButtonC1EP7QWidgetPKc [QtGui]

_ZN11QPushButtonC1ERK5QIconRK7QStringP7QWidget [QtGui]	_ZN11QPushButtonC1ERK5QIconRK7QStringP7QWidgetPKc [QtGui]
_ZN11QPushButtonC1ERK7QStringP7QWidget [QtGui]	_ZN11QPushButtonC1ERK7QStringP7QWidgetPKc [QtGui]
_ZN11QPushButtonC2EP7QWidget [QtGui]	_ZN11QPushButtonC2EP7QWidgetPKc [QtGui]
_ZN11QPushButtonC2ERK5QIconRK7QStringP7QWidget [QtGui]	_ZN11QPushButtonC2ERK5QIconRK7QStringP7QWidgetPKc [QtGui]
_ZN11QPushButtonC2ERK7QStringP7QWidget [QtGui]	_ZN11QPushButtonC2ERK7QStringP7QWidgetPKc [QtGui]
_ZN11QPushButtonD0Ev [QtGui]	_ZN11QPushButtonD1Ev [QtGui]
_ZN11QPushButtonD2Ev [QtGui]	_ZN11QScrollArea10takeWidgetEv [QtGui]
_ZN11QScrollArea11eventFilterEP7QObjectP6QEvent [QtGui]	_ZN11QScrollArea11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN11QScrollArea11qt_metacastEPKc [QtGui]	_ZN11QScrollArea11resizeEventEP12QResizeEvent [QtGui]
_ZN11QScrollArea12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtXml]	_ZN11QScrollArea13ensureVisibleEiii [QtGui]
_ZN11QScrollArea16scrollContentsByEii [QtGui]	_ZN11QScrollArea18focusNextPrevChildEb [QtGui]
_ZN11QScrollArea18setWidgetResizableEb [QtGui]	_ZN11QScrollArea19ensureWidgetVisibleEP7QWidgetii [QtXml]
_ZN11QScrollArea5eventEP6QEvent [QtGui]	_ZN11QScrollArea9setWidgetEP7QWidget [QtGui]
_ZN11QScrollAreaC1EP7QWidget [QtGui]	_ZN11QScrollAreaC2EP7QWidget [QtGui]
_ZN11QScrollAreaD0Ev [QtGui]	_ZN11QScrollAreaD1Ev [QtGui]
_ZN11QScrollAreaD2Ev [QtGui]	_ZN11QToolButton10enterEventEP6QEvent [QtGui]
_ZN11QToolButton10leaveEventEP6QEvent [QtGui]	_ZN11QToolButton10paintEventEP11QPaintEvent [QtGui]
_ZN11QToolButton10setIconSetERK5QIconb [QtGui]	_ZN11QToolButton10timerEventEP11QTimerEvent [QtGui]
_ZN11QToolButton11actionEventEP12QActionEvent [QtGui]	_ZN11QToolButton11changeEventEP6QEvent [QtGui]
_ZN11QToolButton11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN11QToolButton11qt_metacastEPKc [QtGui]
_ZN11QToolButton12setArrowTypeEN2Qt9ArrowTypeE [QtGui]	_ZN11QToolButton12setAutoRaiseEb [QtGui]

_ZN11QToolButton12setOnIconSetERK5QIcon [QtGui]	_ZN11QToolButton12setPopupModeENS_19ToolButtonPopupModeE [QtGui]
_ZN11QToolButton13setOffIconSetERK5QIcon [QtGui]	_ZN11QToolButton13setPopupDelayEi [QtGui]
_ZN11QToolButton14nextCheckStateEv [QtGui]	_ZN11QToolButton15mousePressEventEP11QMouseEvent [QtGui]
_ZN11QToolButton16setDefaultActionEP7QAction [QtGui]	_ZN11QToolButton17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN11QToolButton18setToolButtonStyleEN2Qt15ToolButtonStyleE [QtGui]	_ZN11QToolButton5eventEP6QEvent [QtGui]
_ZN11QToolButton7setMenuEP5QMenu [QtGui]	_ZN11QToolButton8showMenuEv [QtGui]
_ZN11QToolButton9triggeredEP7QAction [QtGui]	_ZN11QToolButtonC1EN2Qt9ArrowTypeEP7QWidgetPKc [QtGui]
_ZN11QToolButtonC1EP7QWidget [QtGui]	_ZN11QToolButtonC1EP7QWidgetPKc [QtGui]
_ZN11QToolButtonC1ERK5QIconRK7QStringS5_P7QObjectPKcP7QWidgetS9_ [QtGui]	_ZN11QToolButtonC2EN2Qt9ArrowTypeEP7QWidgetPKc [QtGui]
_ZN11QToolButtonC2EP7QWidget [QtGui]	_ZN11QToolButtonC2EP7QWidgetPKc [QtGui]
_ZN11QToolButtonC2ERK5QIconRK7QStringS5_P7QObjectPKcP7QWidgetS9_ [QtGui]	_ZN11QToolButtonD0Ev [QtGui]
_ZN11QToolButtonD1Ev [QtGui]	_ZN11QToolButtonD2Ev [QtGui]
_ZN12QProgressBar10paintEventEP11QPaintEvent [QtGui]	_ZN12QProgressBar10setMaximumEi [QtGui]
_ZN12QProgressBar10setMinimumEi [QtGui]	_ZN12QProgressBar11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN12QProgressBar11qt_metacastEPKc [QtGui]	_ZN12QProgressBar12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN12QProgressBar12valueChangedEi [QtGui]	_ZN12QProgressBar13textDirectionEv [QtGui]
_ZN12QProgressBar14setOrientationEN2Qt11OrientationE [QtGui]	_ZN12QProgressBar14setTextVisibleEb [QtGui]
_ZN12QProgressBar16setTextDirectionENS_9DirectionE [QtGui]	_ZN12QProgressBar18invertedAppearanceEv [QtGui]
_ZN12QProgressBar21setInvertedAppearanceEb [QtGui]	_ZN12QProgressBar5eventEP6QEvent [QtGui]

_ZN12QProgressBar5resetEv [QtGui]	_ZN12QProgressBar8setRangeEii [QtGui]
_ZN12QProgressBar8setValueEi [QtGui]	_ZN12QProgressBar9setFormatERK7QString [QtXml]
_ZN12QProgressBarC1EP7QWidget [QtGui]	_ZN12QProgressBarC2EP7QWidget [QtGui]
_ZN12QRadioButton10paintEventEP11QPaintEvent [QtGui]	_ZN12QRadioButton11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN12QRadioButton11qt_metacastEPKc [QtGui]	_ZN12QRadioButton14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN12QRadioButton5eventEP6QEvent [QtGui]	_ZN12QRadioButtonC1EP7QWidget [QtGui]
_ZN12QRadioButtonC1EP7QWidgetPKc [QtGui]	_ZN12QRadioButtonC1ERK7QStringP7QWidget [QtGui]
_ZN12QRadioButtonC1ERK7QStringP7QWidgetPKc [QtGui]	_ZN12QRadioButtonC2EP7QWidget [QtGui]
_ZN12QRadioButtonC2EP7QWidgetPKc [QtGui]	_ZN12QRadioButtonC2ERK7QStringP7QWidget [QtGui]
_ZN12QRadioButtonC2ERK7QStringP7QWidgetPKc [QtGui]	_ZN12QTextBrowser10paintEventEP11QPaintEvent [QtGui]
_ZN12QTextBrowser11highlightedERK4QUrl [QtGui]	_ZN12QTextBrowser11highlightedERK7QString [QtGui]
_ZN12QTextBrowser11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN12QTextBrowser11qt_metacastEPKc [QtGui]
_ZN12QTextBrowser12clearHistoryEv [QtXml]	_ZN12QTextBrowser12loadResourceEiRK4QUrl [QtGui]
_ZN12QTextBrowser13anchorClickedERK4QUrl [QtGui]	_ZN12QTextBrowser13focusOutEventEP11QFocusEvent [QtGui]
_ZN12QTextBrowser13keyPressEventEP9QKeyEvent [QtGui]	_ZN12QTextBrowser13sourceChangedERK4QUrl [QtGui]
_ZN12QTextBrowser14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN12QTextBrowser14setSearchPathsERK11QStringList [QtGui]
_ZN12QTextBrowser15mousePressEventEP11QMouseEvent [QtGui]	_ZN12QTextBrowser16forwardAvailableEb [QtGui]
_ZN12QTextBrowser17backwardAvailableEb [QtGui]	_ZN12QTextBrowser17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN12QTextBrowser18focusNextPrevChildEb [QtGui]	_ZN12QTextBrowser20setOpenExternalLinksEb [QtXml]
_ZN12QTextBrowser4homeEv [QtGui]	_ZN12QTextBrowser5eventEP6QEvent [QtGui]
_ZN12QTextBrowser6reloadEv [QtGui]	_ZN12QTextBrowser7forwardEv [QtGui]

_ZN12QTextBrowser8backwardEv [QtGui]	_ZN12QTextBrowser9setSourceERK4QUrl [QtGui]
_ZN12QTextBrowserC1EP7QWidget [QtGui]	_ZN12QTextBrowserC1EP7QWidgetPKc [QtGui]
_ZN12QTextBrowserC2EP7QWidget [QtGui]	_ZN12QTextBrowserC2EP7QWidgetPKc [QtGui]
_ZN12QTextBrowserD0Ev [QtGui]	_ZN12QTextBrowserD1Ev [QtGui]
_ZN12QTextBrowserD2Ev [QtGui]	_ZN13QInputContext11filterEventEPK6QEvent [QtGui]
_ZN13QInputContext11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN13QInputContext11qt_metacastEPKc [QtGui]
_ZN13QInputContext12mouseHandlerEiP11QMouseEvent [QtGui]	_ZN13QInputContext14setFocusWidgetEP7QWidget [LSB]
_ZN13QInputContext14x11FilterEventEP7QWidgetP7_XEvent [QtGui]	_ZN13QInputContext15widgetDestroyedEP7QWidget [QtGui]
_ZN13QInputContext6updateEv [QtGui]	_ZN13QInputContext7actionsEv [QtGui]
_ZN13QInputContext9sendEventERK17QInputMethodEvent [QtGui]	_ZN13QInputContextC1EP7QObject [QtGui]
_ZN13QInputContextC2EP7QObject [QtGui]	_ZN13QInputContextD0Ev [QtGui]
_ZN13QInputContextD1Ev [QtGui]	_ZN13QInputContextD2Ev [QtGui]
_ZN14QDesktopWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN14QDesktopWidget11qt_metacastEPKc [QtGui]
_ZN14QDesktopWidget11resizeEventEP12QResizeEvent [QtGui]	_ZN14QDesktopWidget15workAreaResizedEi [QtGui]
_ZN14QDesktopWidget6screenEi [QtGui]	_ZN14QDesktopWidget7resizedEi [QtGui]
_ZN14QDesktopWidgetC1Ev [QtGui]	_ZN14QDesktopWidgetC2Ev [QtGui]
_ZN14QDesktopWidgetD0Ev [QtGui]	_ZN14QDesktopWidgetD1Ev [QtGui]
_ZN14QDesktopWidgetD2Ev [QtGui]	_ZN14QDoubleSpinBox10setMaximumEd [QtGui]
_ZN14QDoubleSpinBox10setMinimumEd [QtGui]	_ZN14QDoubleSpinBox11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN14QDoubleSpinBox11qt_metacastEPKc [QtGui]	_ZN14QDoubleSpinBox11setDecimalsEi [QtGui]
_ZN14QDoubleSpinBox12valueChangedERK7QString [QtGui]	_ZN14QDoubleSpinBox12valueChangedEd [QtGui]

_ZN14QDoubleSpinBox13setSingleStepEd [QtGui]	_ZN14QDoubleSpinBox8setRangeEd [QtGui]
_ZN14QDoubleSpinBox8setValueEd [QtGui]	_ZN14QDoubleSpinBox9setPrefixERK7QString [QtGui]
_ZN14QDoubleSpinBox9setSuffixERK7QString [QtGui]	_ZN14QDoubleSpinBoxC1EP7QWidget [QtGui]
_ZN14QDoubleSpinBoxC2EP7QWidget [QtGui]	_ZN14QStackedWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN14QStackedWidget11qt_metacastEPKc [QtGui]	_ZN14QStackedWidget12insertWidgetEiP7QWidget [QtGui]
_ZN14QStackedWidget12removeWidgetEP7QWidget [QtGui]	_ZN14QStackedWidget13widgetRemovedEi [QtGui]
_ZN14QStackedWidget14currentChangedEi [QtGui]	_ZN14QStackedWidget15setCurrentIndexEi [QtGui]
_ZN14QStackedWidget16setCurrentWidgetEP7QWidget [QtGui]	_ZN14QStackedWidget5eventEP6QEvent [QtGui]
_ZN14QStackedWidget9addWidgetEP7QWidget [QtGui]	_ZN14QStackedWidgetC1EP7QWidget [QtGui]
_ZN14QStackedWidgetC2EP7QWidget [QtGui]	_ZN14QStackedWidgetD0Ev [QtGui]
_ZN14QStackedWidgetD1Ev [QtGui]	_ZN14QStackedWidgetD2Ev [QtGui]
_ZN15QCalendarWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN15QCalendarWidget11qt_metacastEPKc [QtXml]
_ZN15QCalendarWidget11resizeEventEP12QResizeEvent [QtXml]	_ZN15QCalendarWidget12setDateRangeERK5QDateS2_ [QtXml]
_ZN15QCalendarWidget12showNextYearEv [QtXml]	_ZN15QCalendarWidget13keyPressEventEP9QKeyEvent [QtXml]
_ZN15QCalendarWidget13showNextMonthEv [QtXml]	_ZN15QCalendarWidget14setCurrentPageEii [QtXml]
_ZN15QCalendarWidget14setGridVisibleEb [QtXml]	_ZN15QCalendarWidget14setMaximumDateERK5QDate [QtXml]
_ZN15QCalendarWidget14setMinimumDateERK5QDate [QtXml]	_ZN15QCalendarWidget15mousePressEventEP11QMouseEvent [QtXml]
_ZN15QCalendarWidget15setSelectedDateERK5QDate [QtXml]	_ZN15QCalendarWidget16selectionChangedEv [QtXml]
_ZN15QCalendarWidget16setHeaderVisibleEb [QtXml]	_ZN15QCalendarWidget16setSelectionModeENS_13SelectionModeE [QtXml]
_ZN15QCalendarWidget16showPreviousYearEv [QtXml]	_ZN15QCalendarWidget16showSelectedDateEv [QtXml]

_ZN15QCalendarWidget17setDateTextFormatERK5QDateRK15QTextCharFormat [QtXml]	_ZN15QCalendarWidget17setFirstDayOfWeekEN2Qt9DayOfWeekE [QtXml]
_ZN15QCalendarWidget17showPreviousMonthEv [QtXml]	_ZN15QCalendarWidget18currentPageChangedEii [QtXml]
_ZN15QCalendarWidget19setHeaderTextFormatERK15QTextCharFormat [QtXml]	_ZN15QCalendarWidget20setWeekdayTextFormatEN2Qt9DayOfWeekERK15QTextCharFormat [QtXml]
_ZN15QCalendarWidget23setVerticalHeaderFormatENS_20VerticalHeaderFormatE [QtXml]	_ZN15QCalendarWidget25setHorizontalHeaderFormatENS_22HorizontalHeaderFormatE [QtXml]
_ZN15QCalendarWidget5eventEP6QEvent [QtXml]	_ZN15QCalendarWidget7clickedERK5QDate [QtXml]
_ZN15QCalendarWidget9activatedERK5QDate [QtXml]	_ZN15QCalendarWidget9showTodayEv [QtXml]
_ZN15QCalendarWidgetC1EP7QWidget [QtXml]	_ZN15QCalendarWidgetC2EP7QWidget [QtXml]
_ZN15QCalendarWidgetD0Ev [QtXml]	_ZN15QCalendarWidgetD1Ev [QtXml]
_ZN15QCalendarWidgetD2Ev [QtXml]	_ZN15QListWidgetItem4readER11QDataStream [QtGui]
_ZN15QListWidgetItem7setDataEiRK8QVariant [QtGui]	_ZN15QListWidgetItem8setFlagsE6QFlagsIN2Qt8ItemFlagEE [QtXml]
_ZN15QListWidgetItemC1EP11QListWidgeti [QtGui]	_ZN15QListWidgetItemC1ERK5QIconRK7QStringP11QListWidgeti [QtGui]
_ZN15QListWidgetItemC1ERK7QStringP11QListWidgeti [QtGui]	_ZN15QListWidgetItemC1ERKS_ [QtGui]
_ZN15QListWidgetItemC2EP11QListWidgeti [QtGui]	_ZN15QListWidgetItemC2ERK5QIconRK7QStringP11QListWidgeti [QtGui]
_ZN15QListWidgetItemC2ERK7QStringP11QListWidgeti [QtGui]	_ZN15QListWidgetItemC2ERKS_ [QtGui]
_ZN15QListWidgetItemD0Ev [QtGui]	_ZN15QListWidgetItemD1Ev [QtGui]
_ZN15QListWidgetItemD2Ev [QtGui]	_ZN15QListWidgetItemD3Ev [QtGui]
_ZN15QSystemTrayIcon10setToolTipERK7QString [QtXml]	_ZN15QSystemTrayIcon10setVisibleEb [QtXml]
_ZN15QSystemTrayIcon11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN15QSystemTrayIcon11qt_metacastEPKc [QtXml]

_ZN15QSystemTrayIcon11showMessageERK7QStringS2_NS_11MessageIconEi [QtXml]	_ZN15QSystemTrayIcon14messageClickedEv [QtXml]
_ZN15QSystemTrayIcon14setContextMenuEP5QMenu [QtXml]	_ZN15QSystemTrayIcon16supportsMessagesEv [QtXml]
_ZN15QSystemTrayIcon21isSystemTrayAvailableEv [QtXml]	_ZN15QSystemTrayIcon5eventEP6QEvent [QtXml]
_ZN15QSystemTrayIcon7setIconERK5QIcon [QtXml]	_ZN15QSystemTrayIcon9activatedENS_16ActivationReasonE [QtXml]
_ZN15QSystemTrayIconC1EP7QObject [QtXml]	_ZN15QSystemTrayIconC1ERK5QIconP7QObject [QtXml]
_ZN15QSystemTrayIconC2EP7QObject [QtXml]	_ZN15QSystemTrayIconC2ERK5QIconP7QObject [QtXml]
_ZN15QSystemTrayIconD0Ev [QtXml]	_ZN15QSystemTrayIconD1Ev [QtXml]
_ZN15QSystemTrayIconD2Ev [QtXml]	_ZN16QDesktopServices13setUrlHandlerERK7QStringP7QObjectPKc [QtXml]
_ZN16QDesktopServices15unsetUrlHandlerERK7QString [QtXml]	_ZN16QDesktopServices7openUrlERK4QUrl [QtXml]
_ZN16QDialogButtonBox11changeEventEP6QEvent [QtXml]	_ZN16QDialogButtonBox11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN16QDialogButtonBox11qt_metacastEPKc [QtXml]	_ZN16QDialogButtonBox12removeButtonEP15QAbstractButton [QtXml]
_ZN16QDialogButtonBox13helpRequestedEv [QtXml]	_ZN16QDialogButtonBox14setOrientationEN2Qt11OrientationE [QtXml]
_ZN16QDialogButtonBox16setCenterButtonsEb [QtXml]	_ZN16QDialogButtonBox18setStandardButtonsE6QFlagsINS_14StandardButtonEE [QtXml]
_ZN16QDialogButtonBox5clearEv [QtXml]	_ZN16QDialogButtonBox5eventEP6QEvent [QtXml]
_ZN16QDialogButtonBox7clickedEP15QAbstractButton [QtXml]	_ZN16QDialogButtonBox8acceptedEv [QtXml]
_ZN16QDialogButtonBox8rejectedEv [QtXml]	_ZN16QDialogButtonBox9addButtonENS_14StandardButtonE [QtXml]
_ZN16QDialogButtonBox9addButtonEP15QAbstractButtonNS_10ButtonRoleE [QtXml]	_ZN16QDialogButtonBox9addButtonERK7QStringNS_10ButtonRoleE [QtXml]
_ZN16QDialogButtonBoxC1E6QFlagsINS_14StandardButtonEEN2Qt11OrientationEP7QWidget [QtXml]	_ZN16QDialogButtonBoxC1EN2Qt11OrientationEP7QWidget [QtXml]

_ZN16QDialogButtonBoxC1EP7QWid dget [QtXml]	_ZN16QDialogButtonBoxC2E6QFlag sINS_14StandardButtonEEN2Qt11Or ientationEP7QWidget [QtXml]
_ZN16QDialogButtonBoxC2EN2Qt1 1OrientationEP7QWidget [QtXml]	_ZN16QDialogButtonBoxC2EP7QWid dget [QtXml]
_ZN16QDialogButtonBoxD0Ev [QtXml]	_ZN16QDialogButtonBoxD1Ev [QtXml]
_ZN16QDialogButtonBoxD2Ev [QtXml]	_ZN20QInputContextFactory11descr iptionERK7QString [QtGui]
_ZN20QInputContextFactory11displ ayNameERK7QString [QtGui]	_ZN20QInputContextFactory4keysE v [QtGui]
_ZN20QInputContextFactory6create ERK7QStringP7QObject [QtGui]	_ZN20QInputContextFactory9langua gesERK7QString [QtGui]
_ZN5QDial10paintEventEP11QPaint Event [QtGui]	_ZN5QDial11dialPressedEv [QtGui]
_ZN5QDial11qt_metacallEN11QMet aObject4CallEiPPv [QtGui]	_ZN5QDial11qt_metacastEPKc [QtGui]
_ZN5QDial11resizeEventEP12QResiz eEvent [QtGui]	_ZN5QDial11setWrappingEb [QtGui]
_ZN5QDial12dialReleasedEv [QtGui]	_ZN5QDial12sliderChangeEN15QAb stractSlider12SliderChangeE [QtGui]
_ZN5QDial14mouseMoveEventEP11 QMouseEvent [QtGui]	_ZN5QDial14setNotchTargetEd [QtGui]
_ZN5QDial15mousePressEventEP11 QMouseEvent [QtGui]	_ZN5QDial17mouseReleaseEventEP 11QMouseEvent [QtGui]
_ZN5QDial17setNotchesVisibleEb [QtGui]	_ZN5QDial5eventEP6QEvent [QtGui]
_ZN5QDial9dialMovedEi [QtGui]	_ZN5QDialC1EP7QWidget [QtGui]
_ZN5QDialC1EP7QWidgetPKc [QtGui]	_ZN5QDialC1EiiiiP7QWidgetPKc [QtGui]
_ZN5QDialC2EP7QWidget [QtGui]	_ZN5QDialC2EP7QWidgetPKc [QtGui]
_ZN5QDialC2EiiiiP7QWidgetPKc [QtGui]	_ZN5QDialD0Ev [QtGui]
_ZN5QDialD1Ev [QtGui]	_ZN5QDialD2Ev [QtGui]
_ZN6QFrame10paintEventEP11QPa intEvent [QtGui]	_ZN6QFrame11changeEventEP6QE vent [QtGui]
_ZN6QFrame11qt_metacallEN11QM etaObject4CallEiPPv [QtGui]	_ZN6QFrame11qt_metacastEPKc [QtGui]
_ZN6QFrame12setFrameRectERK5Q Rect [QtGui]	_ZN6QFrame12setLineWidthEi [QtGui]

_ZN6QFrame13setFrameShapeENS_5ShapeE [QtGui]	_ZN6QFrame13setFrameStyleEi [QtGui]
_ZN6QFrame14setFrameShadowENS_6ShadowE [QtGui]	_ZN6QFrame15setMidLineWidthEi [QtGui]
_ZN6QFrame5eventEP6QEvent [QtGui]	_ZN6QFrame9drawFrameEP8QPainter [QtGui]
_ZN6QFrameC1EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QFrameC1EP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QFrameC2EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QFrameC2EP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QFrameD0Ev [QtGui]	_ZN6QFrameD1Ev [QtGui]
_ZN6QFrameD2Ev [QtGui]	_ZN6QLabel10paintEventEP11QPaintEvent [QtGui]
_ZN6QLabel10setPictureERK8QPicture [QtGui]	_ZN6QLabel11changeEventEP6QEvent [QtGui]
_ZN6QLabel11linkHoveredERK7QString [QtXml]	_ZN6QLabel11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN6QLabel11qt_metacastEPKc [QtGui]	_ZN6QLabel11setWordWrapEb [QtGui]
_ZN6QLabel12focusInEventEP11QFocusEvent [QtXml]	_ZN6QLabel12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN6QLabel12setAlignmentEi [QtGui]	_ZN6QLabel13focusOutEventEP11QFocusEvent [QtXml]
_ZN6QLabel13keyPressEventEP9QKeyEvent [QtXml]	_ZN6QLabel13linkActivatedERK7QString [QtXml]
_ZN6QLabel13setTextFormatEN2Qt10TextFormatE [QtGui]	_ZN6QLabel14mouseMoveEventEP11QMouseEvent [QtXml]
_ZN6QLabel15mousePressEventEP11QMouseEvent [QtXml]	_ZN6QLabel16contextMenuEventEP17QContextMenuEvent [QtXml]
_ZN6QLabel17mouseReleaseEventEP11QMouseEvent [QtXml]	_ZN6QLabel17setScaledContentsEb [QtGui]
_ZN6QLabel18focusNextPrevChildEb [QtXml]	_ZN6QLabel20setOpenExternalLinksEb [QtXml]
_ZN6QLabel23setTextInteractionFlagsE6QFlagsIN2Qt19TextInteractionFlagEE [QtXml]	_ZN6QLabel5clearEv [QtGui]
_ZN6QLabel5eventEP6QEvent [QtGui]	_ZN6QLabel6setNumEd [QtGui]
_ZN6QLabel6setNumEi [QtGui]	_ZN6QLabel7setTextERK7QString [QtGui]

_ZN6QLabel8setBuddyEP7QWidget [QtGui]	_ZN6QLabel8setMovieEP6QMovie [QtGui]
_ZN6QLabel9setIndentEi [QtGui]	_ZN6QLabel9setMarginEi [QtGui]
_ZN6QLabel9setPixmapERK7QPixmap [QtGui]	_ZN6QLabelC1EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QLabelC1EP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QLabelC1EP7QWidgetRK7QStringS1_PKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QLabelC1ERK7QStringP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QLabelC1ERK7QStringP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QLabelC2EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QLabelC2EP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QLabelC2EP7QWidgetRK7QStringS1_PKc6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QLabelC2ERK7QStringP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN6QLabelC2ERK7QStringP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN6QLabelD0Ev [QtGui]
_ZN6QLabelD1Ev [QtGui]	_ZN6QLabelD2Ev [QtGui]
_ZN7QSlider10paintEventEP11QPaintEvent [QtGui]	_ZN7QSlider11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN7QSlider11qt_metacastEPKc [QtGui]	_ZN7QSlider14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN7QSlider15mousePressEventEP11QMouseEvent [QtGui]	_ZN7QSlider15setTickIntervalEi [QtGui]
_ZN7QSlider15setTickPositionENS_12TickPositionE [QtGui]	_ZN7QSlider17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN7QSlider5eventEP6QEvent [QtGui]	_ZN7QSliderC1EN2Qt11OrientationEP7QWidget [QtGui]
_ZN7QSliderC1EN2Qt11OrientationEP7QWidgetPKc [QtGui]	_ZN7QSliderC1EP7QWidget [QtGui]
_ZN7QSliderC1EP7QWidgetPKc [QtGui]	_ZN7QSliderC1EiiiiN2Qt11OrientationEP7QWidgetPKc [QtGui]
_ZN7QSliderC2EN2Qt11OrientationEP7QWidget [QtGui]	_ZN7QSliderC2EN2Qt11OrientationEP7QWidgetPKc [QtGui]
_ZN7QSliderC2EP7QWidget [QtGui]	_ZN7QSliderC2EP7QWidgetPKc [QtGui]
_ZN7QSliderC2EiiiiN2Qt11OrientationEP7QWidgetPKc [QtGui]	_ZN7QSliderD0Ev [QtGui]
_ZN7QSliderD1Ev [QtGui]	_ZN7QSliderD2Ev [QtGui]

_ZN7QTabBar10paintEventEP11QPaintEvent [QtGui]	_ZN7QTabBar10setTabDataEiRK8QVariant [QtGui]
_ZN7QTabBar10setTabIconEiRK5QIcon [QtGui]	_ZN7QTabBar10setTabTextEiRK7QString [QtGui]
_ZN7QTabBar10tabRemovedEi [QtGui]	_ZN7QTabBar11changeEventEP6QEvent [QtGui]
_ZN7QTabBar11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN7QTabBar11qt_metacastEPKc [QtGui]
_ZN7QTabBar11resizeEventEP12QRaiseEvent [QtGui]	_ZN7QTabBar11setDrawBaseEb [QtGui]
_ZN7QTabBar11setIconSizeERK5QSize [QtGui]	_ZN7QTabBar11tabInsertedEi [QtGui]
_ZN7QTabBar12setElideModeEN2Qt13TextElideModeE [QtXml]	_ZN7QTabBar13keyPressEventEP9QKeyEvent [QtGui]
_ZN7QTabBar13setTabEnabledEib [QtGui]	_ZN7QTabBar13setTabToolTipEiRK7QString [QtGui]
_ZN7QTabBar14currentChangedEi [QtGui]	_ZN7QTabBar14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN7QTabBar15mousePressEventEP11QMouseEvent [QtGui]	_ZN7QTabBar15setCurrentIndexEi [QtGui]
_ZN7QTabBar15setTabTextColorEiRK6QColor [QtGui]	_ZN7QTabBar15setTabWhatsThisEiRK7QString [QtGui]
_ZN7QTabBar15tabLayoutChangeEvent [QtGui]	_ZN7QTabBar17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN7QTabBar20setUsesScrollButtonsEb [QtXml]	_ZN7QTabBar5eventEP6QEvent [QtGui]
_ZN7QTabBar6addTabERK5QIconRK7QString [QtGui]	_ZN7QTabBar6addTabERK7QString [QtGui]
_ZN7QTabBar8selectedEi [QtGui]	_ZN7QTabBar8setShapeENS_5ShapeE [QtGui]
_ZN7QTabBar9insertTabEiRK5QIconRK7QString [QtGui]	_ZN7QTabBar9insertTabEiRK7QString [QtGui]
_ZN7QTabBar9removeTabEi [QtGui]	_ZN7QTabBar9showEventEP10QShowEvent [QtGui]
_ZN7QTabBarC1EP7QWidget [QtGui]	_ZN7QTabBarC2EP7QWidget [QtGui]
_ZN7QTabBarD0Ev [QtGui]	_ZN7QTabBarD1Ev [QtGui]
_ZN7QTabBarD2Ev [QtGui]	_ZN7QWidget10addActionE5QListI P7QActionE [QtGui]
_ZN7QWidget10adjustSizeEv [QtGui]	_ZN7QWidget10clearFocusEv [QtGui]

_ZN7QWidget10closeEventEP11QCloseEvent [QtGui]	_ZN7QWidget10enterEventEP6QEvent [QtGui]
_ZN7QWidget10fontChangeERK5QFont [QtGui]	_ZN7QWidget10leaveEventEP6QEvent [QtGui]
_ZN7QWidget10paintEventEP11QPaintEvent [QtGui]	_ZN7QWidget10setEnabledEb [QtGui]
_ZN7QWidget10setPaletteERK8QPalette [QtGui]	_ZN7QWidget10setToolTipERK7QString [QtGui]
_ZN7QWidget10setVisibleEb [QtGui]	_ZN7QWidget10showNormalEv [QtGui]
ZN7QWidget10stackUnderEPS [QtGui]	_ZN7QWidget10wheelEventEP11QWheelEvent [QtGui]
_ZN7QWidget11actionEventEP12QActionEvent [QtGui]	_ZN7QWidget11changeEventEP6QEvent [QtGui]
_ZN7QWidget11createWinIdEv [LSB]	_ZN7QWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN7QWidget11qt_metacastEPKc [QtGui]	_ZN7QWidget11resizeEventEP12QResizeEvent [QtGui]
_ZN7QWidget11setBaseSizeEii [QtGui]	_ZN7QWidget11setDisabledEb [QtGui]
_ZN7QWidget11setGeometryERK5QRect [QtGui]	_ZN7QWidget11setTabOrderEPS_S0_ [QtGui]
_ZN7QWidget11styleChangeERK6QStyle [QtGui]	_ZN7QWidget11tabletEventEP12QTabletEvent [QtGui]
_ZN7QWidget11unsetCursorEv [QtGui]	_ZN7QWidget12focusInEventEP11QFocusEvent [QtGui]
_ZN7QWidget12grabKeyboardEv [QtGui]	_ZN7QWidget12grabShortcutERK12QKeySequenceN2Qt15ShortcutContextE [QtGui]
_ZN7QWidget12inputContextEv [QtGui]	_ZN7QWidget12insertActionEP7QActionS1_ [QtGui]
_ZN7QWidget12mouseGrabberEv [QtGui]	_ZN7QWidget12releaseMouseEv [QtGui]
_ZN7QWidget12removeActionEP7QAction [QtGui]	_ZN7QWidget12setAttributeEN2Qt15WidgetAttributeEb [QtGui]
_ZN7QWidget12setFixedSizeERK5QSize [QtGui]	_ZN7QWidget12setFixedSizeEii [QtGui]
_ZN7QWidget12setStatusTipERK7QString [QtGui]	_ZN7QWidget12setWhatsThisERK7QString [QtGui]
_ZN7QWidget13dragMoveEventEP14QDragMoveEvent [QtGui]	_ZN7QWidget13setEnabledChangeEb [QtGui]

_ZN7QWidget13focusOutEventEP11QFocusEvent [QtGui]	_ZN7QWidget13insertActionsEP7QAction5QListIS1_E [QtGui]
_ZN7QWidget13keyPressEventEP9QKeyEvent [QtGui]	_ZN7QWidget13paletteChangeERK8QPalette [QtGui]
_ZN7QWidget13setFixedWidthEi [QtGui]	_ZN7QWidget13setFocusProxyEPS_ [QtGui]
_ZN7QWidget13setSizePolicyE11QSizePolicy [QtGui]	_ZN7QWidget13setStyleSheetERK7QString [QtXml]
_ZN7QWidget13setWindowIconERK5QIcon [QtGui]	_ZN7QWidget13setWindowRoleERK7QString [QtGui]
_ZN7QWidget13showMaximizedEv [QtGui]	_ZN7QWidget13showMinimizedEv [QtGui]
_ZN7QWidget14activateWindowEv [QtGui]	_ZN7QWidget14dragEnterEventEP15QDragEnterEvent [QtGui]
_ZN7QWidget14dragLeaveEventEP15QDragLeaveEvent [QtGui]	_ZN7QWidget14languageChangeEv [QtGui]
_ZN7QWidget14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN7QWidget14setAcceptDropsEb [QtGui]
_ZN7QWidget14setFixedHeightEi [QtGui]	_ZN7QWidget14setFocusPolicyEN2Qt11FocusPolicyE [QtGui]
_ZN7QWidget14setMaximumSizeEii [QtGui]	_ZN7QWidget14setMinimumSizeEii [QtGui]
_ZN7QWidget14setWindowFlagsE6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN7QWidget14setWindowStateE6QFlagsIN2Qt11WindowStateEE [QtGui]
_ZN7QWidget14setWindowTitleERK7QString [QtGui]	_ZN7QWidget14showFullScreenEv [QtGui]
_ZN7QWidget14updateGeometryEv [QtGui]	_ZN7QWidget15keyReleaseEventEP9QKeyEvent [QtGui]
_ZN7QWidget15keyboardGrabberEv [QtGui]	_ZN7QWidget15mousePressEventEP11QMouseEvent [QtGui]
_ZN7QWidget15releaseKeyboardEv [QtGui]	_ZN7QWidget15releaseShortcutEi [QtGui]
_ZN7QWidget15restoreGeometryERK10QByteArray [QtXml]	_ZN7QWidget15setInputContextEP13QInputContext [QtGui]
_ZN7QWidget15setMaximumWidthEi [QtGui]	_ZN7QWidget15setMinimumWidthEi [QtGui]
_ZN7QWidget16contextMenuEventEP17QContextMenuEvent [QtGui]	_ZN7QWidget16inputMethodEventEP17QInputMethodEvent [QtGui]
_ZN7QWidget16setMaximumHeightEi [QtGui]	_ZN7QWidget16setMinimumHeightEi [QtGui]

_ZN7QWidget16setSizeIncrementEii [QtGui]	_ZN7QWidget16setWindowOpacityEd [QtGui]
_ZN7QWidget16setWindowSurfaceEP14QWindowSurface [QtXml]	_ZN7QWidget16updateMicroFocusEv [QtGui]
_ZN7QWidget17mouseReleaseEventEP11QMouseEvent [QtGui]	_ZN7QWidget17resetInputContextEv [QtGui]
_ZN7QWidget17setAccessibleNameERK7QString [QtGui]	_ZN7QWidget17setBackgroundModeEN2Qt14BackgroundModeES1_ [QtGui]
_ZN7QWidget17setBackgroundRoleEN8QPalette9ColorRoleE [QtGui]	_ZN7QWidget17setForegroundRoleEN8QPalette9ColorRoleE [QtGui]
_ZN7QWidget17setUpdatesEnabledEb [QtGui]	_ZN7QWidget17setWindowIconTextERK7QString [QtGui]
_ZN7QWidget17setWindowModalityEN2Qt14WindowModalityE [QtGui]	_ZN7QWidget17setWindowModifiedEb [QtGui]
_ZN7QWidget18focusNextPrevChildEb [QtGui]	_ZN7QWidget18setContentsMarginsEiiii [QtGui]
_ZN7QWidget18setLayoutDirectionEN2Qt15LayoutDirectionE [QtGui]	_ZN7QWidget18setShortcutEnabledEib [QtGui]
_ZN7QWidget19overrideWindowFlagsE6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN7QWidget19overrideWindowStateE6QFlagsIN2Qt11WindowStateEE [LSB]
_ZN7QWidget20setContextMenuPolicyEN2Qt17ContextMenuPolicyE [QtGui]	_ZN7QWidget20unsetLayoutDirectionEv [QtGui]
_ZN7QWidget21mouseDoubleClickEventEP11QMouseEvent [QtGui]	_ZN7QWidget21setAutoFillBackgroundEb [QtGui]
_ZN7QWidget21setShortcutAutoRepeatEib [QtXml]	_ZN7QWidget22windowActivationChangeEb [QtGui]
_ZN7QWidget24setAccessibleDescriptionERK7QString [QtGui]	_ZN7QWidget26customContextMenuRequestedERK6QPoint [QtGui]
_ZN7QWidget4findEm [QtGui]	_ZN7QWidget4moveERK6QPoint [QtGui]
_ZN7QWidget5closeEb [QtGui]	_ZN7QWidget5closeEv [QtGui]
_ZN7QWidget5eraseERK7QRegion [QtGui]	_ZN7QWidget5eventEP6QEvent [QtGui]
_ZN7QWidget5lowerEv [QtGui]	_ZN7QWidget5raiseEv [QtGui]
_ZN7QWidget6createEmbb [QtGui]	_ZN7QWidget6resizeERK5QSize [QtGui]
_ZN7QWidget6scrollEii [QtGui]	_ZN7QWidget6scrollEiiRK5QRect [QtGui]

_ZN7QWidget6updateERK5QRect [QtGui]	_ZN7QWidget6updateERK7QRegion [QtGui]
_ZN7QWidget6updateEv [QtGui]	_ZN7QWidget7destroyEbb [QtGui]
_ZN7QWidget7repaintERK5QRect [QtGui]	_ZN7QWidget7repaintERK7QRegion [QtGui]
_ZN7QWidget7repaintEiiii [QtGui]	_ZN7QWidget7repaintEv [QtGui]
_ZN7QWidget7setFontERK5QFont [QtGui]	_ZN7QWidget7setIconERK7QPixmap [QtGui]
_ZN7QWidget7setMaskERK7QBitmap [QtGui]	_ZN7QWidget7setMaskERK7QRegion [QtGui]
_ZN7QWidget7wmapperEv [QtGui]	_ZN7QWidget8setFocusEN2Qt11FocusReasonE [QtGui]
_ZN7QWidget8setStyleEP6QStyle [QtGui]	_ZN7QWidget8setStyleERK7QString [QtGui]
_ZN7QWidget8x11EventEP7_XEvent [QtGui]	_ZN7QWidget9addActionEP7QAction [QtGui]
_ZN7QWidget9clearMaskEv [QtGui]	_ZN7QWidget9dropEventEP10QDropEvent [QtGui]
_ZN7QWidget9grabMouseERK7QCursor [QtGui]	_ZN7QWidget9grabMouseEv [QtGui]
_ZN7QWidget9hideEventEP10QHideEvent [QtGui]	_ZN7QWidget9moveEventEP10QMoveEvent [QtGui]
_ZN7QWidget9setCursorERK7QCursor [QtGui]	_ZN7QWidget9setLayoutEP7QLayout [QtGui]
ZN7QWidget9setParentEPS [QtGui]	_ZN7QWidget9setParentEPS_6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN7QWidget9showEventEP10QShowEvent [QtGui]	_ZN7QWidgetC1EPS_6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN7QWidgetC1EPS_PKc6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN7QWidgetC2EPS_6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN7QWidgetC2EPS_PKc6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN7QWidgetD0Ev [QtGui]
_ZN7QWidgetD1Ev [QtGui]	_ZN7QWidgetD2Ev [QtGui]
_ZN8QSpinBox10setMaximumEi [QtGui]	_ZN8QSpinBox10setMinimumEi [QtGui]
_ZN8QSpinBox11qt_metacallEN11QMetaObject4CalleiPPv [QtGui]	_ZN8QSpinBox11qt_metacastEPKc [QtGui]
_ZN8QSpinBox12valueChangedERK7QString [QtGui]	_ZN8QSpinBox12valueChangedEi [QtGui]
_ZN8QSpinBox13setSingleStepEi [QtGui]	_ZN8QSpinBox5eventEP6QEvent [QtGui]

_ZN8QSpinBox8setRangeEii [QtGui]	_ZN8QSpinBox8setValueEi [QtGui]
_ZN8QSpinBox9setPrefixERK7QString [QtGui]	_ZN8QSpinBox9setSuffixERK7QString [QtGui]
_ZN8QSpinBoxC1EP7QWidget [QtGui]	_ZN8QSpinBoxC1EP7QWidgetPKc [QtGui]
_ZN8QSpinBoxC1EiiiP7QWidgetPKc [QtGui]	_ZN8QSpinBoxC2EP7QWidget [QtGui]
_ZN8QSpinBoxC2EP7QWidgetPKc [QtGui]	_ZN8QSpinBoxC2EiiiP7QWidgetPKc [QtGui]
_ZN8QToolBox10insertItemEiP7QWidgetRK5QIconRK7QString [QtGui]	_ZN8QToolBox10removeItemEi [QtGui]
_ZN8QToolBox11changeEventEP6QEvent [QtGui]	_ZN8QToolBox11itemRemovedEi [QtGui]
_ZN8QToolBox11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN8QToolBox11qt_metacastEPKc [QtGui]
_ZN8QToolBox11setItemIconEiRK5QIcon [QtGui]	_ZN8QToolBox11setItemTextEiRK7QString [QtGui]
_ZN8QToolBox12itemInsertedEi [QtGui]	_ZN8QToolBox14currentChangedEi [QtGui]
_ZN8QToolBox14setItemEnabledEib [QtGui]	_ZN8QToolBox14setItemToolTipEiRK7QString [QtGui]
_ZN8QToolBox15setCurrentIndexEi [QtGui]	_ZN8QToolBox16setCurrentWidgetEP7QWidget [QtGui]
_ZN8QToolBox5eventEP6QEvent [QtGui]	_ZN8QToolBox9showEventEP10QShowEvent [QtGui]
_ZN8QToolBoxC1EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN8QToolBoxC1EP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN8QToolBoxC2EP7QWidget6QFlagsIN2Qt10WindowTypeEE [QtGui]	_ZN8QToolBoxC2EP7QWidgetPKc6QFlagsIN2Qt10WindowTypeEE [QtGui]
_ZN8QToolBoxD0Ev [QtGui]	_ZN8QToolBoxD1Ev [QtGui]
_ZN8QToolBoxD2Ev [QtGui]	_ZN8QToolTip10setPaletteERK8QPalette [QtGui]
_ZN8QToolTip4fontEv [QtGui]	_ZN8QToolTip7paletteEv [QtGui]
_ZN8QToolTip7setFontERK5QFont [QtGui]	_ZN8QToolTip8showTextERK6QPointRK7QStringP7QWidget [QtGui]
_ZN8QToolTip8showTextERK6QPointRK7QStringP7QWidgetRK5QRect [QtGui]	_ZN9QCheckBox10paintEventEP11QPaintEvent [QtGui]
_ZN9QCheckBox11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN9QCheckBox11qt_metacastEPKc [QtGui]

_ZN9QCheckBox11setTristateEb [QtGui]	_ZN9QCheckBox12stateChangedEi [QtGui]
_ZN9QCheckBox13checkStateSetEv [QtGui]	_ZN9QCheckBox13setCheckStateEN2Qt10CheckStateE [QtGui]
_ZN9QCheckBox14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN9QCheckBox14nextCheckStateEv [QtGui]
_ZN9QCheckBox5eventEP6QEvent [QtGui]	_ZN9QCheckBoxC1EP7QWidget [QtGui]
_ZN9QCheckBoxC1EP7QWidgetPKc [QtGui]	_ZN9QCheckBoxC1ERK7QStringP7QWidget [QtGui]
_ZN9QCheckBoxC1ERK7QStringP7QWidgetPKc [QtGui]	_ZN9QCheckBoxC2EP7QWidget [QtGui]
_ZN9QCheckBoxC2EP7QWidgetPKc [QtGui]	_ZN9QCheckBoxC2ERK7QStringP7QWidget [QtGui]
_ZN9QCheckBoxC2ERK7QStringP7QWidgetPKc [QtGui]	_ZN9QComboBox10insertItemEiRK5QIconRK7QStringRK8QVariant [QtGui]
_ZN9QComboBox10paintEventEP11QPaintEvent [QtGui]	_ZN9QComboBox10removeItemEi [QtGui]
_ZN9QComboBox10wheelEventEP11QWheelEvent [QtGui]	_ZN9QComboBox11changeEventEP6QEvent [QtGui]
_ZN9QComboBox11highlightedERK7QString [QtGui]	_ZN9QComboBox11highlightedEi [QtGui]
_ZN9QComboBox11insertItemsEiRK11QStringList [QtGui]	_ZN9QComboBox11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QComboBox11qt_metacastEPKc [QtGui]	_ZN9QComboBox11resizeEventEP12QResizeEvent [QtGui]
_ZN9QComboBox11setEditTextERK7QString [QtGui]	_ZN9QComboBox11setEditableEb [QtGui]
_ZN9QComboBox11setIconSizeERK5QSize [QtGui]	_ZN9QComboBox11setItemDataEiRK8QVarianti [QtGui]
_ZN9QComboBox11setItemIconEiRK5QIcon [QtGui]	_ZN9QComboBox11setItemTextEiRK7QString [QtGui]
_ZN9QComboBox11setLineEditEP9QLineEdit [QtGui]	_ZN9QComboBox11setMaxCountEi [QtGui]
_ZN9QComboBox11textChangedERK7QString [QtGui]	_ZN9QComboBox12focusInEventEP11QFocusEvent [QtGui]
_ZN9QComboBox12setCompleterEP10QCompleter [QtGui]	_ZN9QComboBox12setValidatorEPK10QValidator [QtGui]
_ZN9QComboBox13clearEditTextEv [QtGui]	_ZN9QComboBox13focusOutEventEP11QFocusEvent [QtGui]

_ZN9QComboBox13keyPressEventEP9QKeyEvent [QtGui]	_ZN9QComboBox14setModelColumnEi [QtGui]
_ZN9QComboBox15editTextChangedERK7QString [QtGui]	_ZN9QComboBox15keyReleaseEventEP9QKeyEvent [QtGui]
_ZN9QComboBox15mousePressEventEP11QMouseEvent [QtGui]	_ZN9QComboBox15setCurrentIndexEi [QtGui]
_ZN9QComboBox15setInsertPolicyENS_12InsertPolicyE [QtGui]	_ZN9QComboBox15setItemDelegateEP21QAbstractItemDelegate [QtGui]
_ZN9QComboBox16contextMenuEventEP17QContextMenuEvent [QtGui]	_ZN9QComboBox16inputMethodEventEP17QInputMethodEvent [QtGui]
_ZN9QComboBox17mouseReleaseEventEP11QMouseEvent [QtGui]	_ZN9QComboBox17setAutoCompletionEb [QtGui]
_ZN9QComboBox17setRootModelIndexERK11QModelIndex [QtGui]	_ZN9QComboBox18setMaxVisibleItemsEi [QtGui]
_ZN9QComboBox19currentIndexChangedERK7QString [QtGui]	_ZN9QComboBox19currentIndexChangedEi [QtGui]
_ZN9QComboBox19setSizeAdjustPolicyENS_16SizeAdjustPolicyE [QtGui]	_ZN9QComboBox20setDuplicatesEnabledEb [QtGui]
_ZN9QComboBox24setMinimumContentsLengthEi [QtGui]	_ZN9QComboBox32setAutoCompletionCaseSensitivityEN2Qt15CaseSensitivityE [QtGui]
_ZN9QComboBox5clearEv [QtGui]	_ZN9QComboBox5eventEP6QEvent [QtGui]
_ZN9QComboBox7setViewEP17QAbstractItemView [QtGui]	_ZN9QComboBox8 setFrameEb [QtGui]
_ZN9QComboBox8setModelEP18QAbstractItemModel [QtGui]	_ZN9QComboBox9activatedERK7QString [QtGui]
_ZN9QComboBox9activatedEi [QtGui]	_ZN9QComboBox9hideEventEP10QHideEvent [QtGui]
_ZN9QComboBox9hidePopupEv [QtGui]	_ZN9QComboBox9showEventEP10QShowEvent [QtGui]
_ZN9QComboBox9showPopupEv [QtGui]	_ZN9QComboBoxC1EP7QWidget [QtGui]
_ZN9QComboBoxC1EP7QWidgetPKc [QtGui]	_ZN9QComboBoxC1EbP7QWidgetPKc [QtGui]
_ZN9QComboBoxC2EP7QWidget [QtGui]	_ZN9QComboBoxC2EP7QWidgetPKc [QtGui]
_ZN9QComboBoxC2EbP7QWidgetPKc [QtGui]	_ZN9QComboBoxD0Ev [QtGui]
_ZN9QComboBoxD1Ev [QtGui]	_ZN9QComboBoxD2Ev [QtGui]

_ZN9QLineEdit10paintEventEP11QPaintEvent [QtGui]	_ZN9QLineEdit10textEditedERK7QString [QtGui]
_ZN9QLineEdit11changeEventEP6QEvent [QtGui]	_ZN9QLineEdit11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QLineEdit11qt_metacastEPKc [QtGui]	_ZN9QLineEdit11setEchoModeENS_8EchoModeE [QtGui]
_ZN9QLineEdit11setModifiedEb [QtGui]	_ZN9QLineEdit11setReadOnlyEb [QtGui]
_ZN9QLineEdit11textChangedERK7QString [QtGui]	_ZN9QLineEdit12focusInEventEP11QFocusEvent [QtGui]
ZN9QLineEdit12getSelectionEPiS0 [QtGui]	_ZN9QLineEdit12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN9QLineEdit12setCompleterEP10QCompleter [QtXml]	_ZN9QLineEdit12setInputMaskERK7QString [QtGui]
_ZN9QLineEdit12setMaxLengthEi [QtGui]	_ZN9QLineEdit12setSelectionEii [QtGui]
_ZN9QLineEdit12setValidatorEPK10QValidator [QtGui]	_ZN9QLineEdit13cursorForwardEbi [QtGui]
_ZN9QLineEdit13dragMoveEventEP14QDragMoveEvent [QtGui]	_ZN9QLineEdit13focusOutEventEP11QFocusEvent [QtGui]
_ZN9QLineEdit13keyPressEventEP9QKeyEvent [QtGui]	_ZN9QLineEdit13returnPressedEv [QtGui]
_ZN9QLineEdit14cursorBackwardEbi [QtGui]	_ZN9QLineEdit14dragEnterEventEP15QDragEnterEvent [QtGui]
_ZN9QLineEdit14dragLeaveEventEP15QDragLeaveEvent [QtGui]	_ZN9QLineEdit14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN9QLineEdit14setDragEnabledEb [QtGui]	_ZN9QLineEdit14validateAndSetERK7QStringiii [QtGui]
_ZN9QLineEdit15editingFinishedEv [QtGui]	_ZN9QLineEdit15mousePressEventEP11QMouseEvent [QtGui]
_ZN9QLineEdit16contextMenuEventEP17QContextMenuEvent [QtGui]	_ZN9QLineEdit16cursorPositionAtERK6QPoint [QtGui]
_ZN9QLineEdit16inputMethodEventEP17QInputMethodEvent [QtGui]	_ZN9QLineEdit16selectionChangedEv [QtGui]
_ZN9QLineEdit17cursorWordForwardEb [QtGui]	_ZN9QLineEdit17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN9QLineEdit17setCursorPositionEbi [QtGui]	_ZN9QLineEdit18cursorWordBackwardEb [QtGui]
_ZN9QLineEdit21cursorPositionChangedEii [QtGui]	_ZN9QLineEdit21mouseDoubleClickEventEP11QMouseEvent [QtGui]

_ZN9QLineEdit25createStandardCon textMenuEv [QtGui]	_ZN9QLineEdit3cutEv [QtGui]
_ZN9QLineEdit3delEv [QtGui]	_ZN9QLineEdit3endEb [QtGui]
_ZN9QLineEdit4homeEb [QtGui]	_ZN9QLineEdit4redoEv [QtGui]
_ZN9QLineEdit4undoEv [QtGui]	_ZN9QLineEdit5clearEv [QtGui]
_ZN9QLineEdit5eventEP6QEvent [QtGui]	_ZN9QLineEdit5pasteEv [QtGui]
_ZN9QLineEdit6insertERK7QString [QtGui]	_ZN9QLineEdit7setTextERK7QStrin g [QtGui]
_ZN9QLineEdit8deselectEv [QtGui]	_ZN9QLineEdit8setFrameEb [QtGui]
_ZN9QLineEdit9backspaceEv [QtGui]	_ZN9QLineEdit9dropEventEP10QDr opEvent [QtGui]
_ZN9QLineEdit9lostFocusEv [QtGui]	_ZN9QLineEdit9selectAllEv [QtGui]
_ZN9QLineEdit9setEditedEb [QtGui]	_ZN9QLineEditC1EP7QWidget [QtGui]
_ZN9QLineEditC1EP7QWidgetPKc [QtGui]	_ZN9QLineEditC1ERK7QStringP7Q Widget [QtGui]
_ZN9QLineEditC1ERK7QStringP7Q WidgetPKc [QtGui]	_ZN9QLineEditC1ERK7QStringS2_P 7QWidgetPKc [QtGui]
_ZN9QLineEditC2EP7QWidget [QtGui]	_ZN9QLineEditC2EP7QWidgetPKc [QtGui]
_ZN9QLineEditC2ERK7QStringP7Q Widget [QtGui]	_ZN9QLineEditC2ERK7QStringP7Q WidgetPKc [QtGui]
_ZN9QLineEditC2ERK7QStringS2_P 7QWidgetPKc [QtGui]	_ZN9QLineEditD0Ev [QtGui]
_ZN9QLineEditD1Ev [QtGui]	_ZN9QLineEditD2Ev [QtGui]
_ZN9QListView10moveCursorEN17 QAbstractItemView12CursorActionE 6QFlagsIN2Qt16KeyboardModifierE E [QtGui]	_ZN9QListView10paintEventEP11Q PaintEvent [QtGui]
_ZN9QListView10setSpacingEi [QtGui]	_ZN9QListView10timerEventEP11Q TimerEvent [QtGui]
ZN9QListView11dataChangedERK 11QModelIndexS2 [QtGui]	_ZN9QListView11qt_metacallEN11Q MetaObject4CallEiPPv [QtGui]
_ZN9QListView11qt_metacastEPKc [QtGui]	_ZN9QListView11resizeEventEP12Q ResizeEvent [QtGui]
_ZN9QListView11setGridSizeERK5Q Size [QtGui]	_ZN9QListView11setMovementENS _8MovementE [QtGui]
_ZN9QListView11setViewModeENS _8ViewModeE [QtGui]	_ZN9QListView11setWordWrapEb [QtGui]

_ZN9QListView11setWrappingEb [QtGui]	_ZN9QListView12indexesMovedERK5QList11QModelIndexE [QtXml]
_ZN9QListView12internalDragE6QFlagsIN2Qt10DropActionEE [QtGui]	_ZN9QListView12internalDropEP10QDropEvent [QtGui]
_ZN9QListView12rowsInsertedERK11QModelIndexii [QtGui]	_ZN9QListView12setBatchSizeEi [QtXml]
_ZN9QListView12setRootIndexERK11QModelIndex [QtGui]	_ZN9QListView12setRowHiddenEib [QtGui]
_ZN9QListView12setSelectionERK5QRect6QFlagsIN19QItemSelectionModel13SelectionFlagEE [QtGui]	_ZN9QListView13doItemsLayoutEv [LSB]
_ZN9QListView13dragMoveEventEP14QDragMoveEvent [QtGui]	_ZN9QListView13setLayoutModeENS_10LayoutModeE [QtGui]
_ZN9QListView13setResizeModeENS_10ResizeModeE [QtGui]	_ZN9QListView14dragLeaveEventEP15QDragLeaveEvent [QtGui]
_ZN9QListView14mouseMoveEventEP11QMouseEvent [QtGui]	_ZN9QListView14resizeContentsEii [LSB]
_ZN9QListView14setModelColumnEi [QtGui]	_ZN9QListView16scrollContentsByEii [QtGui]
_ZN9QListView16updateGeometriesEv [QtGui]	_ZN9QListView17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN9QListView18clearPropertyFlagsEv [QtGui]	_ZN9QListView19setPositionForIndexERK6QPointRK11QModelIndex [QtGui]
_ZN9QListView19setUniformItemSizesEb [QtGui]	_ZN9QListView20rowsAboutToBeRemovedERK11QModelIndexii [QtGui]
_ZN9QListView5eventEP6QEvent [QtGui]	_ZN9QListView5resetEv [QtGui]
_ZN9QListView7setFlowENS_4FlowE [QtGui]	_ZN9QListView8scrollToERK11QModelIndexN17QAbstractItemView10ScrollHintE [QtGui]
_ZN9QListView9dropEventEP10QDropEvent [QtGui]	_ZN9QListView9startDragE6QFlagsIN2Qt10DropActionEE [QtGui]
_ZN9QListViewC1EP7QWidget [QtGui]	_ZN9QListViewC2EP7QWidget [QtGui]
_ZN9QListViewD0Ev [QtGui]	_ZN9QListViewD1Ev [QtGui]
_ZN9QListViewD2Ev [QtGui]	_ZN9QTextEdit10insertHtmlERK7QString [QtGui]
_ZN9QTextEdit10moveCursorENS11QTextCursor13MoveOperationENS0_8MoveModeE [QtXml]	_ZN9QTextEdit10moveCursorENS_12CursorActionENS11QTextCursor8MoveModeE [QtGui]

_ZN9QTextEdit10moveCursorENS_12CursorActionEb [QtXml]	_ZN9QTextEdit10paintEventEP11QPaintEvent [QtGui]
_ZN9QTextEdit10timerEventEP11QTimerEvent [QtGui]	_ZN9QTextEdit10wheelEventEP11QWheelEvent [QtGui]
_ZN9QTextEdit11changeEventEP6QEvent [QtGui]	_ZN9QTextEdit11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]
_ZN9QTextEdit11qt_metacastEPKc [QtGui]	_ZN9QTextEdit11resizeEventEP12QResizeEvent [QtGui]
_ZN9QTextEdit11setDocumentEP13QTextDocument [QtGui]	_ZN9QTextEdit11setReadOnlyEb [QtGui]
_ZN9QTextEdit11textChangedEv [QtGui]	_ZN9QTextEdit12focusInEventEP11QFocusEvent [QtGui]
_ZN9QTextEdit12loadResourceEiRK4QUrl [QtGui]	_ZN9QTextEdit12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtGui]
_ZN9QTextEdit12setPlainTextERK7QString [QtGui]	_ZN9QTextEdit12setTextColorERK6QColor [QtGui]
_ZN9QTextEdit13copyAvailableEb [QtGui]	_ZN9QTextEdit13dragMoveEventEP14QDragMoveEvent [QtGui]
_ZN9QTextEdit13focusOutEventEP11QFocusEvent [QtGui]	_ZN9QTextEdit13keyPressEventEP9QKeyEvent [QtGui]
_ZN9QTextEdit13redoAvailableEb [QtGui]	_ZN9QTextEdit13setFontFamilyERK7QString [QtGui]
_ZN9QTextEdit13setFontItalicEb [QtGui]	_ZN9QTextEdit13setFontWeightEi [QtGui]
_ZN9QTextEdit13setTextCursorERK11QTextCursor [QtGui]	_ZN9QTextEdit13setTextFormatEN2Qt10TextFormatE [QtGui]
_ZN9QTextEdit13undoAvailableEb [QtGui]	_ZN9QTextEdit14dragEnterEventEP15QDragEnterEvent [QtGui]
_ZN9QTextEdit14dragLeaveEventEP15QDragLeaveEvent [QtGui]	_ZN9QTextEdit14mouseMoveEventEP11QMouseEvent [QtGui]
_ZN9QTextEdit14scrollToAnchorERK7QString [QtGui]	_ZN9QTextEdit14setCurrentFontERK5QFont [QtGui]
_ZN9QTextEdit14setCursorWidthEi [QtXml]	_ZN9QTextEdit15insertPlainTextERK7QString [QtGui]
_ZN9QTextEdit15keyReleaseEventEP9QKeyEvent [QtXml]	_ZN9QTextEdit15mousePressEventEP11QMouseEvent [QtGui]
_ZN9QTextEdit15setLineWrapModeENS_12LineWrapModeE [QtGui]	_ZN9QTextEdit15setTabStopWidthEi [QtGui]
_ZN9QTextEdit15setWordWrapModeEN11QTextOption8WrapModeE [QtGui]	_ZN9QTextEdit16contextMenuEventEP17QContextMenuEvent [QtGui]

_ZN9QTextEdit16doKeyboardActionENS_14KeyboardActionE [QtGui]	_ZN9QTextEdit16inputMethodEventEP17QInputMethodEvent [QtGui]
_ZN9QTextEdit16scrollContentsByEi [QtGui]	_ZN9QTextEdit16selectionChangedEv [QtGui]
_ZN9QTextEdit16setFontPointSizeEd [QtGui]	_ZN9QTextEdit16setFontUnderlineEb [QtGui]
_ZN9QTextEdit16setOverwriteModeEb [QtGui]	_ZN9QTextEdit17mouseReleaseEventEP11QMouseEvent [QtGui]
_ZN9QTextEdit17setAcceptRichTextEb [QtGui]	_ZN9QTextEdit17setAutoFormattingE6QFlagsINS_18AutoFormattingFlagEE [QtGui]
_ZN9QTextEdit18currentFontChangedERK5QFont [QtGui]	_ZN9QTextEdit18focusNextPrevChildEb [QtGui]
_ZN9QTextEdit18insertFromMimeDataEPK9QMimeData [QtGui]	_ZN9QTextEdit18setExtraSelectionsERK5QListINS_14ExtraSelectionEE [QtXml]
_ZN9QTextEdit18setTabChangesFocusEb [QtGui]	_ZN9QTextEdit19currentColorChangedERK6QColor [QtGui]
_ZN9QTextEdit19ensureCursorVisibleEv [QtGui]	_ZN9QTextEdit20setCurrentCharFormatERK15QTextCharFormat [QtGui]
_ZN9QTextEdit21cursorPositionChangedEv [QtGui]	_ZN9QTextEdit21mouseDoubleClickEventEP11QMouseEvent [QtGui]
_ZN9QTextEdit22mergeCurrentCharFormatERK15QTextCharFormat [QtGui]	_ZN9QTextEdit23setTextInteractionFlagsE6QFlagsIN2Qt19TextInteractionFlagEE [QtXml]
_ZN9QTextEdit24currentCharFormatChangedERK15QTextCharFormat [QtGui]	_ZN9QTextEdit24setLineWrapColumnOrWidthEi [QtGui]
_ZN9QTextEdit25createStandardContextMenuEv [QtGui]	_ZN9QTextEdit3cutEv [QtGui]
_ZN9QTextEdit4copyEv [QtGui]	_ZN9QTextEdit4findERK7QString6QFlagsIN13QTextDocument8FindFlagEE [QtGui]
_ZN9QTextEdit4redoEv [QtXml]	_ZN9QTextEdit4undoEv [QtXml]
_ZN9QTextEdit5clearEv [QtGui]	_ZN9QTextEdit5eventEP6QEvent [QtGui]
_ZN9QTextEdit5pasteEv [QtGui]	_ZN9QTextEdit6appendERK7QString [QtGui]
_ZN9QTextEdit6zoomInEi [QtGui]	_ZN9QTextEdit7setHtmlERK7QString [QtGui]
_ZN9QTextEdit7setTextERK7QString [QtGui]	_ZN9QTextEdit7zoomOutEi [QtGui]

_ZN9QTextEdit9dropEventEP10QDropEvent [QtGui]	_ZN9QTextEdit9selectAllEv [QtGui]
_ZN9QTextEdit9showEventEP10QShowEvent [QtGui]	_ZN9QTextEditC1EP7QWidget [QtGui]
_ZN9QTextEditC1EP7QWidgetPKc [QtGui]	_ZN9QTextEditC1ERK7QStringP7QWidget [QtGui]
_ZN9QTextEditC2EP7QWidget [QtGui]	_ZN9QTextEditC2EP7QWidgetPKc [QtGui]
_ZN9QTextEditC2ERK7QStringP7QWidget [QtGui]	_ZN9QTextEditD0Ev [QtGui]
_ZN9QTextEditD1Ev [QtGui]	_ZN9QTextEditD2Ev [QtGui]
_ZNK10QLCDNumber10metaObjectEv [QtGui]	_ZNK10QLCDNumber12segmentStyleEv [QtGui]
_ZNK10QLCDNumber13checkOverflowEd [QtGui]	_ZNK10QLCDNumber13checkOverflowEi [QtGui]
_ZNK10QLCDNumber17smallDecimalPointEv [QtGui]	_ZNK10QLCDNumber4modeEv [QtGui]
_ZNK10QLCDNumber5valueEv [QtGui]	_ZNK10QLCDNumber8intValueEv [QtGui]
_ZNK10QLCDNumber8sizeHintEv [QtGui]	_ZNK10QLCDNumber9numDigitsEv [QtGui]
_ZNK10QScrollBar10metaObjectEv [QtGui]	_ZNK10QScrollBar8sizeHintEv [QtGui]
_ZNK10QTabWidget10metaObjectEv [QtGui]	_ZNK10QTabWidget10tabToolTipEi [QtGui]
_ZNK10QTabWidget11tabPositionEv [QtGui]	_ZNK10QTabWidget12cornerWidgetEN2Qt6CornerE [QtGui]
_ZNK10QTabWidget12currentIndexEv [QtGui]	_ZNK10QTabWidget12isTabEnabledEi [QtGui]
_ZNK10QTabWidget12tabWhatsThisEi [QtGui]	_ZNK10QTabWidget13currentWidgetEv [QtGui]
_ZNK10QTabWidget15minimumSizeHintEv [QtGui]	_ZNK10QTabWidget17usesScrollButtonsEv [QtXml]
_ZNK10QTabWidget5countEv [QtGui]	_ZNK10QTabWidget6tabBarEv [QtGui]
_ZNK10QTabWidget6widgetEi [QtGui]	_ZNK10QTabWidget7indexOfEP7QWidget [QtGui]
_ZNK10QTabWidget7tabIconEi [QtGui]	_ZNK10QTabWidget7tabTextEi [QtGui]
_ZNK10QTabWidget8iconSizeEv [QtXml]	_ZNK10QTabWidget8sizeHintEv [QtGui]

_Znk10QTabWidget8tabShapeEv [QtGui]	_Znk10QTabWidget9elideModeEv [QtXml]
_Znk11QDockWidget10metaObjectEv [QtGui]	_Znk11QDockWidget12allowedAreasEv [QtGui]
_Znk11QDockWidget16toggleViewActionEv [QtGui]	_Znk11QDockWidget6widgetEv [QtGui]
_Znk11QDockWidget8featuresEv [QtGui]	_Znk11QFocusFrame10metaObjectEv [QtGui]
_Znk11QFocusFrame6widgetEv [QtGui]	_Znk11QListWidget10currentRowEv [QtGui]
_Znk11QListWidget10itemWidgetEP15QListWidgetItem [QtGui]	_Znk11QListWidget10metaObjectEv [QtGui]
_Znk11QListWidget11currentItemEv [QtGui]	_Znk11QListWidget12isItemHiddenEPK15QListWidgetItem [QtGui]
_Znk11QListWidget13indexFromItemEP15QListWidgetItem [QtGui]	_Znk11QListWidget13itemFromIndexEPK11QModelIndex [QtGui]
_Znk11QListWidget13selectedItemsEv [QtGui]	_Znk11QListWidget14isItemSelectedEPK15QListWidgetItem [QtGui]
_Znk11QListWidget14visualItemRectEPK15QListWidgetItem [QtGui]	_Znk11QListWidget16isSortingEnabledEv [QtXml]
_Znk11QListWidget20supportedDropActionsEv [QtGui]	_Znk11QListWidget3rowEPK15QListWidgetItem [QtGui]
_Znk11QListWidget4itemEi [QtGui]	_Znk11QListWidget5countEv [QtGui]
_Znk11QListWidget5itemsEPK9QMimeTypeData [QtGui]	_Znk11QListWidget6itemAtEPK6QPoint [QtGui]
_Znk11QListWidget8mimeDataE5QListIP15QListWidgetItemE [QtGui]	_Znk11QListWidget9findItemsEPK7QString6QFlagsIN2Qt9MatchFlagEE [QtGui]
_Znk11QListWidget9mimeTypesEv [QtGui]	_Znk11QMainWindow10isAnimatedEv [QtXml]
_Znk11QMainWindow10menuWidgetEv [QtXml]	_Znk11QMainWindow10metaObjectEv [QtGui]
_Znk11QMainWindow11isSeparatorEPK6QPoint [QtXml]	_Znk11QMainWindow11toolBarAreaEP8QToolBar [QtGui]
_Znk11QMainWindow13centralWidgetEv [QtGui]	_Znk11QMainWindow14dockWidgetAreaEP11QDockWidget [QtGui]
_Znk11QMainWindow15toolButtonStyleEv [QtGui]	_Znk11QMainWindow20isDockNestingEnabledEv [QtXml]
_Znk11QMainWindow6cornerEN2Qt6CornerE [QtGui]	_Znk11QMainWindow7menuBarEv [QtGui]

_Znk11QMainWindow8iconSizeEv [QtGui]	_Znk11QMainWindow9saveStateEv [QtGui]
_Znk11QMainWindow9statusBarEv [QtGui]	_Znk11QPushButton10metaObjectEv [QtGui]
_Znk11QPushButton11autoDefaultEv [QtGui]	_Znk11QPushButton4menuEv [QtGui]
_Znk11QPushButton6isFlatEv [QtGui]	_Znk11QPushButton8sizeHintEv [QtGui]
_Znk11QPushButton9isDefaultEv [QtGui]	_Znk11QScrollArea10metaObjectEv [QtGui]
_Znk11QScrollArea15widgetResizableEv [QtGui]	_Znk11QScrollArea6widgetEv [QtGui]
_Znk11QScrollArea8sizeHintEv [QtGui]	_Znk11QScrollArea9alignmentEv [QtXml]
_Znk11QToolButton10metaObjectEv [QtGui]	_Znk11QToolButton10offIconSetEv [QtGui]
_Znk11QToolButton10popupDelayEv [QtGui]	_Znk11QToolButton13defaultActionEv [QtGui]
_Znk11QToolButton15minimumSizeHintEv [QtGui]	_Znk11QToolButton15toolButtonStyleEv [QtGui]
_Znk11QToolButton4menuEv [QtGui]	_Znk11QToolButton7iconSetEv [QtGui]
_Znk11QToolButton8sizeHintEv [QtGui]	_Znk11QToolButton9arrowTypeEv [QtGui]
_Znk11QToolButton9autoRaiseEv [QtGui]	_Znk11QToolButton9hitButtonERK6QPoint [QtXml]
_Znk11QToolButton9onIconSetEv [QtGui]	_Znk11QToolButton9popupModeEv [QtGui]
_Znk12QProgressBar10metaObjectEv [QtGui]	_Znk12QProgressBar11orientationEv [QtGui]
_Znk12QProgressBar13isTextVisibleEv [QtGui]	_Znk12QProgressBar15minimumSizeHintEv [QtGui]
_Znk12QProgressBar4textEv [QtGui]	_Znk12QProgressBar5valueEv [QtGui]
_Znk12QProgressBar6formatEv [QtXml]	_Znk12QProgressBar7maximumEv [QtGui]
_Znk12QProgressBar7minimumEv [QtGui]	_Znk12QProgressBar8sizeHintEv [QtGui]
_Znk12QProgressBar9alignmentEv [QtGui]	_Znk12QRadioButton10metaObjectEv [QtGui]
_Znk12QRadioButton8sizeHintEv [QtGui]	_Znk12QRadioButton9hitButtonERK6QPoint [QtGui]

_ZNK12QTextBrowser10metaObjectEv [QtGui]	_ZNK12QTextBrowser11searchPathsEv [QtGui]
_ZNK12QTextBrowser17openExternalLinksEv [QtXml]	_ZNK12QTextBrowser18isForwardAvailableEv [QtXml]
_ZNK12QTextBrowser19isBackwardAvailableEv [QtXml]	_ZNK12QTextBrowser6sourceEv [QtGui]
_ZNK13QInputContext10metaObjectEv [QtGui]	_ZNK13QInputContext11focusWidgetEv [LSB]
_ZNK13QInputContext14standardFormatENS_14StandardFormatE [QtGui]	_ZNK13QInputContext4fontEv [QtGui]
_ZNK14QDesktopWidget10metaObjectEv [QtGui]	_ZNK14QDesktopWidget10numScreensEv [QtGui]
_ZNK14QDesktopWidget12screenNumberEPK7QWidget [QtGui]	_ZNK14QDesktopWidget12screenNumberERK6QPoint [QtGui]
_ZNK14QDesktopWidget13primaryScreenEv [QtGui]	_ZNK14QDesktopWidget14screenGeometryEi [QtGui]
_ZNK14QDesktopWidget16isVirtualDesktopEv [QtGui]	_ZNK14QDesktopWidget17availableGeometryEi [QtGui]
_ZNK14QDoubleSpinBox10metaObjectEv [QtGui]	_ZNK14QDoubleSpinBox10singleStepEv [QtGui]
_ZNK14QDoubleSpinBox13textFromValueEd [QtGui]	_ZNK14QDoubleSpinBox13valueFromTextERK7QString [QtGui]
_ZNK14QDoubleSpinBox5fixupER7QString [QtGui]	_ZNK14QDoubleSpinBox5valueEv [QtGui]
_ZNK14QDoubleSpinBox6prefixEv [QtGui]	_ZNK14QDoubleSpinBox6suffixEv [QtGui]
_ZNK14QDoubleSpinBox7maximumEv [QtGui]	_ZNK14QDoubleSpinBox7minimumEv [QtGui]
_ZNK14QDoubleSpinBox8decimalsEv [QtGui]	_ZNK14QDoubleSpinBox8validateER7QStringRi [QtGui]
_ZNK14QDoubleSpinBox9cleanTextEv [QtGui]	_ZNK14QStackedWidget10metaObjectEv [QtGui]
_ZNK14QStackedWidget12currentIndexEv [QtGui]	_ZNK14QStackedWidget13currentWidgetEv [QtGui]
_ZNK14QStackedWidget5countEv [QtGui]	_ZNK14QStackedWidget6widgetEi [QtGui]
_ZNK14QStackedWidget7indexOfEP7QWidget [QtGui]	_ZNK15QCalendarWidget10metaObjectEv [QtXml]
_ZNK15QCalendarWidget10monthsShownEv [QtXml]	_ZNK15QCalendarWidget11maximumDateEv [QtXml]

_Znk15QCalendarWidget11minimumDateEv [QtXml]	_Znk15QCalendarWidget12selectedDateEv [QtXml]
_Znk15QCalendarWidget13isGridView [QtXml]	_Znk15QCalendarWidget13selectionModeEv [QtXml]
_Znk15QCalendarWidget14dateTextFormatERK5QDate [QtXml]	_Znk15QCalendarWidget14dateTextFormatEv [QtXml]
_Znk15QCalendarWidget14firstDayOfWeekEv [QtXml]	_Znk15QCalendarWidget15isHeaderVisibleEv [QtXml]
_Znk15QCalendarWidget15minimumSizeHintEv [QtXml]	_Znk15QCalendarWidget16headerTextFormatEv [QtXml]
_Znk15QCalendarWidget17weekdayTextFormatEN2Qt9DayOfWeekE [QtXml]	_Znk15QCalendarWidget20verticalHeaderFormatEv [QtXml]
_Znk15QCalendarWidget22horizontalHeaderFormatEv [QtXml]	_Znk15QCalendarWidget8sizeHintEv [QtXml]
_Znk15QCalendarWidget9paintCellEP8QPainterRK5QRectRK5QDate [QtXml]	_Znk15QCalendarWidget9yearShownEv [QtXml]
_Znk15QListWidgetItem4dataEi [QtGui]	_Znk15QListWidgetItem5cloneEv [QtGui]
_Znk15QListWidgetItem5writeER11QDataStream [QtGui]	_Znk15QListWidgetItemltERKS_ [QtGui]
_Znk15QSystemTrayIcon10metaObjectEv [QtXml]	_Znk15QSystemTrayIcon11contextMenuEv [QtXml]
_Znk15QSystemTrayIcon4iconEv [QtXml]	_Znk15QSystemTrayIcon7toolTipEv [QtXml]
_Znk15QSystemTrayIcon9isVisibleEv [QtXml]	_Znk16QDialogButtonBox10buttonRoleEP15QAbstractButton [QtXml]
_Znk16QDialogButtonBox10metaObjectEv [QtXml]	_Znk16QDialogButtonBox11orientationEv [QtXml]
_Znk16QDialogButtonBox13centerButtonsEv [QtXml]	_Znk16QDialogButtonBox14standardButtonEP15QAbstractButton [QtXml]
_Znk16QDialogButtonBox15standardButtonsEv [QtXml]	_Znk16QDialogButtonBox6buttonENS_14StandardButtonE [QtXml]
_Znk16QDialogButtonBox7buttonsEv [QtXml]	_Znk5QDial10metaObjectEv [QtGui]
_Znk5QDial11notchTargetEv [QtGui]	_Znk5QDial14notchesVisibleEv [QtGui]
_Znk5QDial15minimumSizeHintEv [QtGui]	_Znk5QDial8sizeHintEv [QtGui]
_Znk5QDial8wrappingEv [QtGui]	_Znk5QDial9notchSizeEv [QtGui]

_ZNK6QFrame10frameShapeEv [QtGui]	_ZNK6QFrame10frameStyleEv [QtGui]
_ZNK6QFrame10frameWidthEv [QtGui]	_ZNK6QFrame10metaObjectEv [QtGui]
_ZNK6QFrame11frameShadowEv [QtGui]	_ZNK6QFrame12midLineWidthEv [QtGui]
_ZNK6QFrame8sizeHintEv [QtGui]	_ZNK6QFrame9frameRectEv [QtGui]
_ZNK6QFrame9lineWidthEv [QtGui]	_ZNK6QLabel10metaObjectEv [QtGui]
_ZNK6QLabel10textFormatEv [QtGui]	_ZNK6QLabel14heightForWidthEv [QtGui]
_ZNK6QLabel15minimumSizeHintEv [QtGui]	_ZNK6QLabel17hasScaledContentsEv [QtGui]
_ZNK6QLabel17openExternalLinksEv [QtXml]	_ZNK6QLabel20textInputInteractionFlagsEv [QtXml]
_ZNK6QLabel4textEv [QtGui]	_ZNK6QLabel5buddyEv [QtGui]
_ZNK6QLabel5movieEv [QtGui]	_ZNK6QLabel6indentEv [QtGui]
_ZNK6QLabel6marginEv [QtGui]	_ZNK6QLabel6pixmapEv [QtGui]
_ZNK6QLabel7pictureEv [QtGui]	_ZNK6QLabel8sizeHintEv [QtGui]
_ZNK6QLabel8wordWrapEv [QtGui]	_ZNK6QLabel9alignmentEv [QtGui]
_ZNK7QSlider10metaObjectEv [QtGui]	_ZNK7QSlider12tickIntervalEv [QtGui]
_ZNK7QSlider12tickPositionEv [QtGui]	_ZNK7QSlider15minimumSizeHintEv [QtGui]
_ZNK7QSlider8sizeHintEv [QtGui]	_ZNK7QTabBar10metaObjectEv [QtGui]
_ZNK7QTabBar10tabToolTipEi [QtGui]	_ZNK7QTabBar11tabSizeHintEi [QtGui]
_ZNK7QTabBar12currentIndexEv [QtGui]	_ZNK7QTabBar12isTabEnabledEi [QtGui]
_ZNK7QTabBar12tabTextColorEi [QtGui]	_ZNK7QTabBar12tabWhatsThisEi [QtGui]
_ZNK7QTabBar15minimumSizeHintEv [QtGui]	_ZNK7QTabBar17usesScrollButtonsEv [QtXml]
_ZNK7QTabBar5countEv [QtGui]	_ZNK7QTabBar5shapeEv [QtGui]
_ZNK7QTabBar7tabDataEi [QtGui]	_ZNK7QTabBar7tabIconEi [QtGui]
_ZNK7QTabBar7tabRectEi [QtGui]	_ZNK7QTabBar7tabTextEi [QtGui]
_ZNK7QTabBar8drawBaseEv [QtGui]	_ZNK7QTabBar8iconSizeEv [QtGui]

_Znk7QTabBar8sizeHintEv [QtGui]	_Znk7QTabBar9elideModeEv [QtXml]
_Znk7QWidget10focusProxyEv [QtGui]	_Znk7QWidget10metaObjectEv [QtGui]
_Znk7QWidget10sizePolicyEv [QtGui]	_Znk7QWidget10styleSheetEv [QtXml]
_Znk7QWidget10windowIconEv [QtGui]	_Znk7QWidget10windowRoleEv [QtGui]
_Znk7QWidget11acceptDropsEv [QtGui]	_Znk7QWidget11focusPolicyEv [QtGui]
_Znk7QWidget11focusWidgetEv [QtGui]	_Znk7QWidget11isEnabledToEPS [QtGui]
_Znk7QWidget11isMaximizedEv [QtGui]	_Znk7QWidget11isMinimizedEv [QtGui]
Znk7QWidget11isVisibleToEPS [QtGui]	_Znk7QWidget11mapToGlobalERK6QPoint [QtGui]
_Znk7QWidget11mapToParentERK6QPoint [QtGui]	_Znk7QWidget11maximumSizeEv [QtGui]
_Znk7QWidget11minimumSizeEv [QtGui]	_Znk7QWidget11paintEngineEv [QtGui]
_Znk7QWidget11visibleRectEv [QtGui]	_Znk7QWidget11windowStateEv [QtGui]
_Znk7QWidget11windowTitleEv [QtGui]	_Znk7QWidget12childrenRectEv [QtGui]
_Znk7QWidget12contentsRectEv [QtGui]	_Znk7QWidget12isAncestorOfEPKS_ [QtGui]
_Znk7QWidget12isFullScreenEv [QtGui]	_Znk7QWidget12saveGeometryEv [QtXml]
_Znk7QWidget13frameGeometryEv [QtGui]	_Znk7QWidget13mapFromGlobalERK6QPoint [QtGui]
_Znk7QWidget13mapFromParentERK6QPoint [QtGui]	_Znk7QWidget13sizeIncrementEv [QtGui]
_Znk7QWidget13visibleRegionEv [QtGui]	_Znk7QWidget13windowOpacityEv [QtGui]
_Znk7QWidget13windowSurfaceEv [QtXml]	_Znk7QWidget14accessibleNameEv [QtGui]
_Znk7QWidget14backgroundModeEv [QtGui]	_Znk7QWidget14backgroundRoleEv [QtGui]
_Znk7QWidget14childrenRegionEv [QtGui]	_Znk7QWidget14ensurePolishedEv [QtGui]
_Znk7QWidget14foregroundRoleEv [QtGui]	_Znk7QWidget14heightForWidthEv [QtGui]

_Znk7QWidget14isActiveWindowEv [QtGui]	_Znk7QWidget14normalGeometryEv [QtGui]
_Znk7QWidget14windowIconTextEv [QtGui]	_Znk7QWidget14windowModalityEv [QtGui]
_Znk7QWidget15layoutDirectionEv [QtGui]	_Znk7QWidget15minimumSizeHintEv [QtGui]
_Znk7QWidget16inputMethodQueryEN2Qt16InputMethodQueryE [QtGui]	_Znk7QWidget16isWindowModifiedEv [QtGui]
_Znk7QWidget16nextInFocusChainEv [QtGui]	_Znk7QWidget16x11PictureHandleEv [QtGui]
_Znk7QWidget17contextMenuPolicyEv [QtGui]	_Znk7QWidget18autoFillBackgroundEv [QtGui]
_Znk7QWidget18getContentsMarginsEPiS0_S0_S0_ [QtGui]	_Znk7QWidget1xEv [QtGui]
_Znk7QWidget1yEv [QtGui]	_Znk7QWidget20testAttribute_helperEN2Qt15WidgetAttributeE [QtGui]
_Znk7QWidget21accessibleDescriptionEv [QtGui]	_Znk7QWidget3posEv [QtGui]
_Znk7QWidget4iconEv [QtGui]	_Znk7QWidget4maskEv [QtGui]
_Znk7QWidget5mapToEPS_RK6QPoint [QtGui]	_Znk7QWidget5styleEv [QtGui]
_Znk7QWidget5winIdEv [QtXml]	_Znk7QWidget6cursorEv [QtGui]
_Znk7QWidget6handleEv [QtGui]	_Znk7QWidget6layoutEv [QtGui]
_Znk7QWidget6metricEN12QPaintDevice17PaintDeviceMetricE [QtGui]	_Znk7QWidget6windowEv [QtGui]
_Znk7QWidget7actionsEv [QtGui]	_Znk7QWidget7childAtERK6QPoint [QtGui]
_Znk7QWidget7devTypeEv [LSB]	_Znk7QWidget7mapFromEPS_RK6QPoint [QtGui]
_Znk7QWidget7paletteEv [QtGui]	_Znk7QWidget7toolTipEv [QtGui]
_Znk7QWidget7x11InfoEv [QtGui]	_Znk7QWidget8baseSizeEv [QtGui]
_Znk7QWidget8hasFocusEv [QtGui]	_Znk7QWidget8sizeHintEv [QtGui]
_Znk7QWidget9frameSizeEv [QtGui]	_Znk7QWidget9statusTipEv [QtGui]
_Znk7QWidget9whatsThisEv [QtGui]	_Znk8QSpinBox10metaObjectEv [QtGui]
_Znk8QSpinBox10singleStepEv [QtGui]	_Znk8QSpinBox13textFromValueEi [QtGui]

_Znk8QSpinBox13valueFromTextE RK7QString [QtGui]	_Znk8QSpinBox5fixupER7QString [QtGui]
_Znk8QSpinBox5valueEv [QtGui]	_Znk8QSpinBox6prefixEv [QtGui]
_Znk8QSpinBox6suffixEv [QtGui]	_Znk8QSpinBox7maximumEv [QtGui]
_Znk8QSpinBox7minimumEv [QtGui]	_Znk8QSpinBox8validateER7QStrin gRi [QtGui]
_Znk8QSpinBox9cleanTextEv [QtGui]	_Znk8QToolBox10metaObjectEv [QtGui]
_Znk8QToolBox11itemToolTipEi [QtGui]	_Znk8QToolBox12currentIndexEv [QtGui]
_Znk8QToolBox13currentWidgetEv [QtGui]	_Znk8QToolBox13isItemEnabledEi [QtGui]
_Znk8QToolBox5countEv [QtGui]	_Znk8QToolBox6widgetEi [QtGui]
_Znk8QToolBox7indexOfEP7QWid get [QtGui]	_Znk8QToolBox8itemIconEi [QtGui]
_Znk8QToolBox8itemTextEi [QtGui]	_Znk9QCheckBox10checkStateEv [QtGui]
_Znk9QCheckBox10isTristateEv [QtGui]	_Znk9QCheckBox10metaObjectEv [QtGui]
_Znk9QCheckBox8sizeHintEv [QtGui]	_Znk9QCheckBox9hitButtonERK6Q Point [QtGui]
_Znk9QComboBox10isEditableEv [QtGui]	_Znk9QComboBox10metaObjectEv [QtGui]
_Znk9QComboBox11currentTextEv [QtGui]	_Znk9QComboBox11modelColumn Ev [QtGui]
_Znk9QComboBox12currentIndexE v [QtGui]	_Znk9QComboBox12insertPolicyEv [QtGui]
_Znk9QComboBox12itemDelegateE v [QtGui]	_Znk9QComboBox14autoCompleti onEv [QtGui]
_Znk9QComboBox14rootModelIndex Ev [QtGui]	_Znk9QComboBox15maxVisibleIte msEv [QtGui]
_Znk9QComboBox15minimumSize HintEv [QtGui]	_Znk9QComboBox16inputMethodQ ueryEN2Qt16InputMethodQueryE [QtGui]
_Znk9QComboBox16sizeAdjustPoli cyEv [QtGui]	_Znk9QComboBox17duplicatesEna bledEv [QtGui]
_Znk9QComboBox21minimumCont entsLengthEv [QtGui]	_Znk9QComboBox29autoCompleti onCaseSensitivityEv [QtGui]
_Znk9QComboBox4viewEv [QtGui]	_Znk9QComboBox5countEv [QtGui]

_ZNK9QComboBox5modelEv [QtGui]	_ZNK9QComboBox8findDataERK8Q Varianti6QFlagsIN2Qt9MatchFlagEE [QtGui]
_ZNK9QComboBox8hasFrameEv [QtGui]	_ZNK9QComboBox8iconSizeEv [QtGui]
_ZNK9QComboBox8itemDataEii [QtGui]	_ZNK9QComboBox8itemIconEi [QtGui]
_ZNK9QComboBox8itemTextEi [QtGui]	_ZNK9QComboBox8lineEditEv [QtGui]
_ZNK9QComboBox8maxCountEv [QtGui]	_ZNK9QComboBox8sizeHintEv [QtGui]
_ZNK9QComboBox9completerEv [QtXml]	_ZNK9QComboBox9validatorEv [QtGui]
_ZNK9QLineEdit10isModifiedEv [QtGui]	_ZNK9QLineEdit10isReadOnlyEv [QtGui]
_ZNK9QLineEdit10metaObjectEv [QtGui]	_ZNK9QLineEdit11characterAtEiP5 QChar [QtGui]
_ZNK9QLineEdit11displayTextEv [QtGui]	_ZNK9QLineEdit11dragEnabledEv [QtGui]
_ZNK9QLineEdit12selectedTextEv [QtGui]	_ZNK9QLineEdit14cursorPositionEv [QtGui]
_ZNK9QLineEdit14selectionStartEv [QtGui]	_ZNK9QLineEdit15hasSelectedTextE v [QtGui]
_ZNK9QLineEdit15isRedoAvailable Ev [QtGui]	_ZNK9QLineEdit15isUndoAvailable Ev [QtGui]
_ZNK9QLineEdit15minimumSizeHi ntEv [QtGui]	_ZNK9QLineEdit16inputMethodQue ryEN2Qt16InputMethodQueryE [QtGui]
_ZNK9QLineEdit18hasAcceptableIn putEv [QtGui]	_ZNK9QLineEdit4copyEv [QtGui]
_ZNK9QLineEdit4textEv [QtGui]	_ZNK9QLineEdit6editedEv [QtGui]
_ZNK9QLineEdit8echoModeEv [QtGui]	_ZNK9QLineEdit8hasFrameEv [QtGui]
_ZNK9QLineEdit8sizeHintEv [QtGui]	_ZNK9QLineEdit9alignmentEv [QtGui]
_ZNK9QLineEdit9completerEv [QtXml]	_ZNK9QLineEdit9inputMaskEv [QtGui]
_ZNK9QLineEdit9maxLengthEv [QtGui]	_ZNK9QLineEdit9validatorEv [QtGui]
_ZNK9QListView10isWrappingEv [QtGui]	_ZNK9QListView10layoutModeEv [QtGui]

_ZNK9QListView10metaObjectEv [QtGui]	_ZNK9QListView10resizeModeEv [QtGui]
_ZNK9QListView10visualRectERK11QModelIndex [QtGui]	_ZNK9QListView11isRowHiddenEi [QtGui]
_ZNK9QListView11modelColumnEv [QtGui]	_ZNK9QListView11viewOptionsEv [QtGui]
_ZNK9QListView12contentsSizeEv [QtGui]	_ZNK9QListView12rectForIndexERK11QModelIndex [QtGui]
_ZNK9QListView13isIndexHiddenERK11QModelIndex [QtGui]	_ZNK9QListView14verticalOffsetEv [QtGui]
_ZNK9QListView15selectedIndexesEv [QtGui]	_ZNK9QListView16horizontalOffsetEv [QtGui]
_ZNK9QListView16uniformItemSizeEv [QtGui]	_ZNK9QListView24visualRegionForSelectionERK14QItemSelection [QtGui]
_ZNK9QListView4flowEv [QtGui]	_ZNK9QListView7indexAtERK6QPoint [QtGui]
_ZNK9QListView7spacingEv [QtGui]	_ZNK9QListView8gridSizeEv [QtGui]
_ZNK9QListView8movementEv [QtGui]	_ZNK9QListView8viewModeEv [QtGui]
_ZNK9QListView8wordWrapEv [QtXml]	_ZNK9QListView9batchSizeEv [QtXml]
_ZNK9QTextEdit10cursorRectERK11QTextCursor [QtGui]	_ZNK9QTextEdit10cursorRectEv [QtGui]
_ZNK9QTextEdit10fontFamilyEv [QtGui]	_ZNK9QTextEdit10fontItalicEv [QtGui]
_ZNK9QTextEdit10fontWeightEv [QtGui]	_ZNK9QTextEdit10isReadOnlyEv [QtGui]
_ZNK9QTextEdit10metaObjectEv [QtGui]	_ZNK9QTextEdit10textCursorEv [QtGui]
_ZNK9QTextEdit10textFormatEv [QtGui]	_ZNK9QTextEdit11currentFontEv [QtGui]
_ZNK9QTextEdit11cursorWidthEv [QtXml]	_ZNK9QTextEdit12lineWrapModeEv [QtGui]
_ZNK9QTextEdit12tabStopWidthEv [QtGui]	_ZNK9QTextEdit12wordWrapModeEv [QtGui]
_ZNK9QTextEdit13fontPointSizeEv [QtGui]	_ZNK9QTextEdit13fontUnderlineEv [QtGui]
_ZNK9QTextEdit13overwriteModeEv [QtGui]	_ZNK9QTextEdit14acceptRichTextEv [QtGui]

_ZNK9QTextEdit14autoFormattingEv [QtGui]	_ZNK9QTextEdit15extraSelectionsEv [QtXml]
_ZNK9QTextEdit15tabChangesFocusEv [QtGui]	_ZNK9QTextEdit16inputMethodQueryEN2Qt16InputMethodQueryE [QtGui]
_ZNK9QTextEdit17currentCharFormatEv [QtGui]	_ZNK9QTextEdit17cursorForPositionERK6QPoint [QtGui]
_ZNK9QTextEdit20textInteractionFlagsEv [QtXml]	_ZNK9QTextEdit21canInsertFromMimeDataEPK9QMimeData [QtGui]
_ZNK9QTextEdit21lineWrapColumnOrWidthEv [QtGui]	_ZNK9QTextEdit27createMimeDataFromSelectionEv [QtGui]
_ZNK9QTextEdit4textEv [QtGui]	_ZNK9QTextEdit8anchorAtERK6QPoint [QtGui]
_ZNK9QTextEdit8scanPasteEv [QtXml]	_ZNK9QTextEdit8documentEv [QtGui]
_ZNK9QTextEdit9alignmentEv [QtGui]	_ZNK9QTextEdit9textColorEv [QtGui]
_ZlsR11QDataStreamRK15QListWidgetItem [QtGui]	_ZrsR11QDataStreamR15QListWidgetItem [QtGui]

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 Widgets specified in Table 18-478, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-478 libQtGui - Qt4 Widgets Deprecated Function Interfaces

_ZN7QWidget10fontChangeERK5QFont [QtGui]	_ZN7QWidget11styleChangeERK6QStyle [QtGui]
_ZN7QWidget13enabledChangeEb [QtGui]	_ZN7QWidget13paletteChangeERK8QPalette [QtGui]
_ZN7QWidget14languageChangeEv [QtGui]	_ZN7QWidget22windowActivationChangeEb [QtGui]
_ZN9QComboBox17setAutoCompletionEb [QtGui]	_ZN9QComboBox32setAutoCompletionCaseSensitivityEN2Qt15CaseSensitivityE [QtGui]
_ZNK9QComboBox14autoCompletionEv [QtGui]	_ZNK9QComboBox29autoCompletionCaseSensitivityEv [QtGui]

18.5.21 Qt4 Drag and Drop

18.5.21.1 Class data for QMimeSource

The virtual table for the QMimeSource class is described by Table 18-479

Table 18-479 Primary vtable for QMimeSource

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QMimeSource
vfunc[0]:	QMimeSource::~QMimeSource()
vfunc[1]:	QMimeSource::~~QMimeSource()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	QMimeSource::provides(char const*) const
vfunc[4]:	__cxa_pure_virtual

The Run Time Type Information for the QMimeSource class is described by Table 18-480

Table 18-480 typeinfo for QMimeSource

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QMimeSource

18.5.21.2 Class data for QDrag

The virtual table for the QDrag class is described by Table 18-481

Table 18-481 Primary vtable for QDrag

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDrag
vfunc[0]:	QDrag::metaObject() const
vfunc[1]:	QDrag::qt_metacast(char const*)
vfunc[2]:	QDrag::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QDrag::~QDrag()
vfunc[4]:	QDrag::~~QDrag()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QDrag class is described by Table 18-482

Table 18-482 typeinfo for QDrag

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDrag
basetype:	typeinfo for QObject

18.5.21.3 Class data for QDropEvent

The virtual table for the QDropEvent class is described by Table 18-483

Table 18-483 Primary vtable for QDropEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDropEvent
vfunc[0]:	QDropEvent::~~QDropEvent()
vfunc[1]:	QDropEvent::~~QDropEvent()
vfunc[2]:	QDropEvent::format(int) const
vfunc[3]:	QDropEvent::encodedData(char const*) const
vfunc[4]:	QDropEvent::provides(char const*) const

The Run Time Type Information for the QDropEvent class is described by Table 18-484

Table 18-484 typeinfo for QDropEvent

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	
Name	typeinfo name for QDropEvent	
flags:	0	
basetype:	typeinfo for QEvent	2
basetype:	typeinfo for QMimeSource	3074

18.5.21.4 Class data for QDragMoveEvent

The virtual table for the QDragMoveEvent class is described by Table 18-485

Table 18-485 Primary vtable for QDragMoveEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDragMoveEvent
vfunc[0]:	QDragMoveEvent::~~QDragMoveEvent()
vfunc[1]:	QDragMoveEvent::~~QDragMoveEvent()
vfunc[2]:	QDropEvent::format(int) const
vfunc[3]:	QDropEvent::encodedData(char const*) const
vfunc[4]:	QDropEvent::provides(char const*) const

The Run Time Type Information for the QDragMoveEvent class is described by Table 18-486

Table 18-486 typeinfo for QDragMoveEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDragMoveEvent
basetype:	typeinfo for QDropEvent

18.5.21.5 Class data for QDragEnterEvent

The virtual table for the QDragEnterEvent class is described by Table 18-487

Table 18-487 Primary vtable for QDragEnterEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDragEnterEvent
vfunc[0]:	QDragEnterEvent::~~QDragEnterEvent()
vfunc[1]:	QDragEnterEvent::~~QDragEnterEvent()
vfunc[2]:	QDropEvent::format(int) const
vfunc[3]:	QDropEvent::encodedData(char const*) const
vfunc[4]:	QDropEvent::provides(char const*) const

The Run Time Type Information for the QDragEnterEvent class is described by Table 18-488

Table 18-488 typeinfo for QDragEnterEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDragEnterEvent
basetype:	typeinfo for QDragMoveEvent

18.5.21.6 Class data for QDragResponseEvent

The virtual table for the QDragResponseEvent class is described by Table 18-489

Table 18-489 Primary vtable for QDragResponseEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDragResponseEvent
vfunc[0]:	QDragResponseEvent::~~QDragResponseEvent()
vfunc[1]:	QDragResponseEvent::~~QDragResponseEvent()

The Run Time Type Information for the QDragResponseEvent class is described by Table 18-490

Table 18-490 typeinfo for QDragResponseEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDragResponseEvent
basetype:	typeinfo for QEvent

18.5.21.7 Class data for QDragLeaveEvent

The virtual table for the QDragLeaveEvent class is described by Table 18-491

Table 18-491 Primary vtable for QDragLeaveEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QDragLeaveEvent
vfunc[0]:	QDragLeaveEvent::~~QDragLeaveEvent()
vfunc[1]:	QDragLeaveEvent::~~QDragLeaveEvent()

The Run Time Type Information for the QDragLeaveEvent class is described by Table 18-492

Table 18-492 typeinfo for QDragLeaveEvent

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QDragLeaveEvent
basetype:	typeinfo for QEvent

18.5.21.8 Interfaces for Qt4 Drag and Drop

An LSB conforming implementation shall provide the generic functions for Qt4 Drag and Drop specified in Table 18-493, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-493 libQtGui - Qt4 Drag and Drop Function Interfaces

_ZN10QDropEvent13setDropAction EN2Qt10DropActionE [QtGui]	_ZN10QDropEventC1ERK6QPoint6 QFlagsIN2Qt10DropActionEEP K9QMimeDataS3_INS4_11MouseBut tonEES3_INS4_16KeyboardModifier EEN6QEvent4TypeE [QtGui]
_ZN10QDropEventC2ERK6QPoint6 QFlagsIN2Qt10DropActionEEP K9QMimeDataS3_INS4_11MouseBut tonEES3_INS4_16KeyboardModifier EEN6QEvent4TypeE [QtGui]	_ZN10QDropEventD0Ev [QtGui]
_ZN10QDropEventD1Ev [QtGui]	_ZN10QDropEventD2Ev [QtGui]
_ZN11QMimeSourceD0Ev [QtGui]	_ZN11QMimeSourceD1Ev [QtGui]
_ZN11QMimeSourceD2Ev [QtGui]	_ZN14QDragMoveEventC1ERK6QP oint6QFlagsIN2Qt10DropActionEEP K9QMimeDataS3_INS4_11MouseBut tonEES3_INS4_16KeyboardModifier EEN6QEvent4TypeE [QtGui]
_ZN14QDragMoveEventC2ERK6QP oint6QFlagsIN2Qt10DropActionEEP K9QMimeDataS3_INS4_11MouseBut tonEES3_INS4_16KeyboardModifier EEN6QEvent4TypeE [QtGui]	_ZN14QDragMoveEventD0Ev [QtGui]
_ZN14QDragMoveEventD1Ev [QtGui]	_ZN14QDragMoveEventD2Ev [QtGui]
_ZN15QDragEnterEventC1ERK6QP oint6QFlagsIN2Qt10DropActionEEP K9QMimeDataS3_INS4_11MouseBut tonEES3_INS4_16KeyboardModifier EE [QtGui]	_ZN15QDragEnterEventC2ERK6QP oint6QFlagsIN2Qt10DropActionEEP K9QMimeDataS3_INS4_11MouseBut tonEES3_INS4_16KeyboardModifier EE [QtGui]
_ZN15QDragEnterEventD0Ev [QtGui]	_ZN15QDragEnterEventD1Ev [QtGui]
_ZN15QDragEnterEventD2Ev [QtGui]	_ZN15QDragLeaveEventC1Ev [QtGui]

_ZN15QDragLeaveEventC2Ev [QtGui]	_ZN15QDragLeaveEventD0Ev [QtGui]
_ZN15QDragLeaveEventD1Ev [QtGui]	_ZN15QDragLeaveEventD2Ev [QtGui]
_ZN18QDragResponseEventC1Eb [LSB]	_ZN18QDragResponseEventC2Eb [LSB]
_ZN18QDragResponseEventD0Ev [QtGui]	_ZN18QDragResponseEventD1Ev [QtGui]
_ZN18QDragResponseEventD2Ev [QtGui]	_ZN5QDrag10setHotSpotERK6QPoint [QtGui]
_ZN5QDrag11qt_metacallEN11QMetaObject4CallEiPPv [QtGui]	_ZN5QDrag11qt_metacastEPKc [QtGui]
_ZN5QDrag11setMimeDataEP9QMimeData [QtGui]	_ZN5QDrag13actionChangedEN2Qt10DropActionE [QtGui]
_ZN5QDrag13setDragCursorERK7QPixmapN2Qt10DropActionE [QtGui]	_ZN5QDrag13targetChangedEP7QWidget [QtGui]
_ZN5QDrag5startE6QFlagsIN2Qt10DropActionEE [QtGui]	_ZN5QDrag9setPixmapERK7QPixmap [QtGui]
_ZN5QDragC1EP7QWidget [QtGui]	_ZN5QDragC2EP7QWidget [QtGui]
_ZN5QDragD0Ev [QtGui]	_ZN5QDragD1Ev [QtGui]
_ZN5QDragD2Ev [QtGui]	_ZNK10QDropEvent11encodedDataEPKc [QtGui]
_ZNK10QDropEvent6actionEv [QtGui]	_ZNK10QDropEvent6formatEi [QtGui]
_ZNK10QDropEvent6sourceEv [QtGui]	_ZNK10QDropEvent8providesEPKc [QtGui]
_ZNK11QMimeSource8providesEPKc [QtGui]	_ZNK5QDrag10metaObjectEv [QtGui]
_ZNK5QDrag6pixmapEv [QtGui]	_ZNK5QDrag6sourceEv [QtGui]
_ZNK5QDrag6targetEv [QtGui]	_ZNK5QDrag7hotSpotEv [QtGui]
_ZNK5QDrag8mimeDataEv [QtGui]	

18.5.22 Qt4 2D Graphics

18.5.22.1 Class data for QIconEngineFactoryInterface

The virtual table for the QIconEngineFactoryInterface class is described by Table 18-494

Table 18-494 Primary vtable for QIconEngineFactoryInterface

Base Offset	0
Virtual Base Offset	0

RTTI	typeinfo for QIconEngineFactoryInterface
vfunc[0]:	NULL or QIconEngineFactoryInterface::~~QIconEngineFactoryInterface()
vfunc[1]:	NULL or QIconEngineFactoryInterface::~~QIconEngineFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QIconEngineFactoryInterface class is described by Table 18-495

Table 18-495 typeinfo for QIconEngineFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QIconEngineFactoryInterface
basetype:	typeinfo for QFactoryInterface

18.5.22.2 Class data for QIconEngine

The virtual table for the QIconEngine class is described by Table 18-496

Table 18-496 Primary vtable for QIconEngine

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QIconEngine
vfunc[0]:	QIconEngine::~~QIconEngine()
vfunc[1]:	QIconEngine::~~QIconEngine()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	QIconEngine::actualSize(QSize const&, QIcon::Mode, QIcon::State)
vfunc[4]:	QIconEngine::pixmap(QSize const&, QIcon::Mode, QIcon::State)
vfunc[5]:	QIconEngine::addPixmap(QPixmap const&, QIcon::Mode, QIcon::State)
vfunc[6]:	QIconEngine::addFile(QString const&, QSize const&, QIcon::Mode, QIcon::State)

The Run Time Type Information for the QIconEngine class is described by Table 18-497

Table 18-497 typeinfo for QIconEngine

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QIconEngine

18.5.22.3 Class data for QPaintEngine

The virtual table for the QPaintEngine class is described by Table 18-498

Table 18-498 Primary vtable for QPaintEngine

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QPaintEngine
vfunc[0]:	QPaintEngine::~~QPaintEngine()
vfunc[1]:	QPaintEngine::~~QPaintEngine()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	QPaintEngine::drawRects(QRect const*, int)
vfunc[6]:	QPaintEngine::drawRects(QRectF const*, int)
vfunc[7]:	QPaintEngine::drawLines(QLine const*, int)
vfunc[8]:	QPaintEngine::drawLines(QLineF const*, int)
vfunc[9]:	QPaintEngine::drawEllipse(QRectF const&)
vfunc[10]:	QPaintEngine::drawEllipse(QRect const&)
vfunc[11]:	QPaintEngine::drawPath(QPainterPa th const&)
vfunc[12]:	QPaintEngine::drawPoints(QPointF const*, int)
vfunc[13]:	QPaintEngine::drawPoints(QPoint const*, int)
vfunc[14]:	QPaintEngine::drawPolygon(QPoint F const*, int, QPaintEngine::PolygonDrawMode)
vfunc[15]:	QPaintEngine::drawPolygon(QPoint const*, int, QPaintEngine::PolygonDrawMode)

vfunc[16]:	__cxa_pure_virtual
vfunc[17]:	QPaintEngine::drawTextItem(QPointF const&, QTextItem const&)
vfunc[18]:	QPaintEngine::drawTiledPixmap(QRectF const&, QPixmap const&, QPointF const&)
vfunc[19]:	QPaintEngine::drawImage(QRectF const&, QImage const&, QRectF const&, QFlags<Qt::ImageConversionFlag>)
vfunc[20]:	QPaintEngine::coordinateOffset() const
vfunc[21]:	__cxa_pure_virtual

The Run Time Type Information for the QPaintEngine class is described by Table 18-499

Table 18-499 typeinfo for QPaintEngine

Base Vtable	vtable for __cxxabi1::__class_type_info
Name	typeinfo name for QPaintEngine

18.5.22.4 Class data for QGraphicsEllipseItem

The virtual table for the QGraphicsEllipseItem class is described by Table 18-500

Table 18-500 Primary vtable for QGraphicsEllipseItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsEllipseItem
vfunc[0]:	QGraphicsEllipseItem::~~QGraphicsEllipseItem()
vfunc[1]:	QGraphicsEllipseItem::~~QGraphicsEllipseItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsEllipseItem::boundingRect() const
vfunc[4]:	QGraphicsEllipseItem::shape() const
vfunc[5]:	QGraphicsEllipseItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const

vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsEllipseItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsEllipseItem::opaqueArea() const
vfunc[10]:	QGraphicsEllipseItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)
vfunc[11]:	QGraphicsEllipseItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)

vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)
vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsEllipseItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsEllipseItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsEllipseItem::extension(QVariant const&) const

18.5.22.5 Class data for QGraphicsItem

The virtual table for the QGraphicsItem class is described by Table 18-501

Table 18-501 Primary vtable for QGraphicsItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsItem
vfunc[0]:	QGraphicsItem::~~QGraphicsItem()
vfunc[1]:	QGraphicsItem::~~QGraphicsItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	QGraphicsItem::shape() const
vfunc[5]:	QGraphicsItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const

vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsItem::opaqueArea() const
vfunc[10]:	__cxa_pure_virtual
vfunc[11]:	QGraphicsItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)

vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)
vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsItem::extension(QVariant const&) const

18.5.22.6 Class data for QGraphicsItemAnimation

The virtual table for the QGraphicsItemAnimation class is described by Table 18-502

Table 18-502 Primary vtable for QGraphicsItemAnimation

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsItemAnimation
vfunc[0]:	QGraphicsItemAnimation::metaObject() const
vfunc[1]:	QGraphicsItemAnimation::qt_metacast(char const*)
vfunc[2]:	QGraphicsItemAnimation::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QGraphicsItemAnimation::~QGraphicsItemAnimation()
vfunc[4]:	QGraphicsItemAnimation::~QGraphicsItemAnimation()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)

vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QGraphicsItemAnimation::beforeAnimationStep(double)
vfunc[13]:	QGraphicsItemAnimation::afterAnimationStep(double)

18.5.22.7 Class data for QGraphicsItemGroup

The virtual table for the QGraphicsItemGroup class is described by Table 18-503

Table 18-503 Primary vtable for QGraphicsItemGroup

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsItemGroup
vfunc[0]:	QGraphicsItemGroup::~~QGraphicsItemGroup()
vfunc[1]:	QGraphicsItemGroup::~~QGraphicsItemGroup()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsItemGroup::boundingRect() const
vfunc[4]:	QGraphicsItem::shape() const
vfunc[5]:	QGraphicsItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsItemGroup::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsItemGroup::opaqueArea() const
vfunc[10]:	QGraphicsItemGroup::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)

vfunc[11]:	QGraphicsItemGroup::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)

vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsItem::extension(QVariant const&) const

18.5.22.8 Class data for QGraphicsLineItem

The virtual table for the QGraphicsLineItem class is described by Table 18-504

Table 18-504 Primary vtable for QGraphicsLineItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsLineItem
vfunc[0]:	QGraphicsLineItem::~~QGraphicsLineItem()
vfunc[1]:	QGraphicsLineItem::~~QGraphicsLineItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsLineItem::boundingRect() const
vfunc[4]:	QGraphicsLineItem::shape() const
vfunc[5]:	QGraphicsLineItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsLineItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsLineItem::opaqueArea() const
vfunc[10]:	QGraphicsLineItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)

vfunc[11]:	QGraphicsLineItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)

vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsLineItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsLineItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsLineItem::extension(QVariant const&) const

18.5.22.9 Class data for QGraphicsPathItem

The virtual table for the QGraphicsPathItem class is described by Table 18-505

Table 18-505 Primary vtable for QGraphicsPathItem

Base Offset	0
Virtual Base Offset	0
RTTI	TypeInfo for QGraphicsPathItem
vfunc[0]:	QGraphicsPathItem::~~QGraphicsPathItem()
vfunc[1]:	QGraphicsPathItem::~~QGraphicsPathItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsPathItem::boundingRect() const
vfunc[4]:	QGraphicsPathItem::shape() const
vfunc[5]:	QGraphicsPathItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsPathItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsPathItem::opaqueArea() const
vfunc[10]:	QGraphicsPathItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)

vfunc[11]:	QGraphicsPathItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)

vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsPathItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsPathItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsPathItem::extension(QVariant const&) const

18.5.22.10 Class data for QGraphicsPixmapItem

The virtual table for the QGraphicsPixmapItem class is described by Table 18-506

Table 18-506 Primary vtable for QGraphicsPixmapItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsPixmapItem
vfunc[0]:	QGraphicsPixmapItem::~~QGraphicsPixmapItem()
vfunc[1]:	QGraphicsPixmapItem::~~QGraphicsPixmapItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsPixmapItem::boundingRect() const
vfunc[4]:	QGraphicsPixmapItem::shape() const
vfunc[5]:	QGraphicsPixmapItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsPixmapItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsPixmapItem::opaqueArea() const
vfunc[10]:	QGraphicsPixmapItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)

vfunc[11]:	QGraphicsPixmapItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)

vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsPixmapItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsPixmapItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsPixmapItem::extension(QVariant const&) const

18.5.22.11 Class data for QGraphicsPolygonItem

The virtual table for the QGraphicsPolygonItem class is described by Table 18-507

Table 18-507 Primary vtable for QGraphicsPolygonItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsPolygonItem
vfunc[0]:	QGraphicsPolygonItem::~~QGraphicsPolygonItem()
vfunc[1]:	QGraphicsPolygonItem::~~QGraphicsPolygonItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsPolygonItem::boundingRect() const
vfunc[4]:	QGraphicsPolygonItem::shape() const
vfunc[5]:	QGraphicsPolygonItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsPolygonItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsPolygonItem::opaqueArea() const

vfunc[10]:	QGraphicsPolygonItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)
vfunc[11]:	QGraphicsPolygonItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)

vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)
vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsPolygonItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsPolygonItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsPolygonItem::extension(QVariant const&) const

18.5.22.12 Class data for QGraphicsRectItem

The virtual table for the QGraphicsRectItem class is described by Table 18-508

Table 18-508 Primary vtable for QGraphicsRectItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsRectItem
vfunc[0]:	QGraphicsRectItem::~~QGraphicsRectItem()
vfunc[1]:	QGraphicsRectItem::~~QGraphicsRectItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsRectItem::boundingRect() const
vfunc[4]:	QGraphicsRectItem::shape() const
vfunc[5]:	QGraphicsRectItem::contains(QPointF const&) const
vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsRectItem::isObscuredBy(QGraphicsItem const*) const

vfunc[9]:	QGraphicsRectItem::opaqueArea() const
vfunc[10]:	QGraphicsRectItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)
vfunc[11]:	QGraphicsRectItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QG raphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGr aphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QG raphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QG raphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphi csSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocu sEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFoc usEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(Q GraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(Q GraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(Q GraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKe yEvent*)
vfunc[25]:	QGraphicsItem::keyReleaseEvent(QK eyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(Q GraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(Q GraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)

vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)
vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsRectItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsRectItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsRectItem::extension(QVariant const&) const

18.5.22.13 Class data for QGraphicsScene

The virtual table for the QGraphicsScene class is described by Table 18-509

Table 18-509 Primary vtable for QGraphicsScene

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsScene
vfunc[0]:	QGraphicsScene::metaObject() const
vfunc[1]:	QGraphicsScene::qt_metacast(char const*)
vfunc[2]:	QGraphicsScene::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QGraphicsScene::~~QGraphicsScene()
vfunc[4]:	QGraphicsScene::~~QGraphicsScene()
vfunc[5]:	QGraphicsScene::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)

vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QGraphicsScene::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[13]:	QGraphicsScene::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[14]:	QGraphicsScene::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[15]:	QGraphicsScene::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsScene::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsScene::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsScene::focusInEvent(QFocusEvent*)
vfunc[19]:	QGraphicsScene::focusOutEvent(QFocusEvent*)
vfunc[20]:	QGraphicsScene::helpEvent(QGraphicsSceneHelpEvent*)
vfunc[21]:	QGraphicsScene::keyPressEvent(QKeyEvent*)
vfunc[22]:	QGraphicsScene::keyReleaseEvent(QKeyEvent*)
vfunc[23]:	QGraphicsScene::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[24]:	QGraphicsScene::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[25]:	QGraphicsScene::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[26]:	QGraphicsScene::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsScene::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[28]:	QGraphicsScene::inputMethodEvent(QInputMethodEvent*)
vfunc[29]:	QGraphicsScene::drawBackground(QPainter*, QRectF const&)
vfunc[30]:	QGraphicsScene::drawForeground(QPainter*, QRectF const&)
vfunc[31]:	QGraphicsScene::drawItems(QPainter*, int, QGraphicsItem**,

	QStyleOptionGraphicsItem const*, QWidget*)

The Run Time Type Information for the QGraphicsScene class is described by Table 18-510

Table 18-510 typeinfo for QGraphicsScene

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QGraphicsScene
basetype:	typeinfo for QObject

18.5.22.14 Class data for QGraphicsSceneContextMenuEvent

The virtual table for the QGraphicsSceneContextMenuEvent class is described by Table 18-511

Table 18-511 Primary vtable for QGraphicsSceneContextMenuEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneContextMenuEvent
vfunc[0]:	QGraphicsSceneContextMenuEvent:: ~QGraphicsSceneContextMenuEvent ()
vfunc[1]:	QGraphicsSceneContextMenuEvent:: ~QGraphicsSceneContextMenuEvent ()

18.5.22.15 Class data for QGraphicsSceneDragDropEvent

The virtual table for the QGraphicsSceneDragDropEvent class is described by Table 18-512

Table 18-512 Primary vtable for QGraphicsSceneDragDropEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneDragDropEvent
vfunc[0]:	QGraphicsSceneDragDropEvent::~~Q GraphicsSceneDragDropEvent()
vfunc[1]:	QGraphicsSceneDragDropEvent::~~Q GraphicsSceneDragDropEvent()

18.5.22.16 Class data for QGraphicsSceneEvent

The virtual table for the QGraphicsSceneEvent class is described by Table 18-513

Table 18-513 Primary vtable for QGraphicsSceneEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneEvent
vfunc[0]:	QGraphicsSceneEvent::~~QGraphicsSceneEvent()
vfunc[1]:	QGraphicsSceneEvent::~~QGraphicsSceneEvent()

18.5.22.17 Class data for QGraphicsSceneHelpEvent

The virtual table for the QGraphicsSceneHelpEvent class is described by Table 18-514

Table 18-514 Primary vtable for QGraphicsSceneHelpEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneHelpEvent
vfunc[0]:	QGraphicsSceneHelpEvent::~~QGraphicsSceneHelpEvent()
vfunc[1]:	QGraphicsSceneHelpEvent::~~QGraphicsSceneHelpEvent()

18.5.22.18 Class data for QGraphicsSceneHoverEvent

The virtual table for the QGraphicsSceneHoverEvent class is described by Table 18-515

Table 18-515 Primary vtable for QGraphicsSceneHoverEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneHoverEvent
vfunc[0]:	QGraphicsSceneHoverEvent::~~QGraphicsSceneHoverEvent()
vfunc[1]:	QGraphicsSceneHoverEvent::~~QGraphicsSceneHoverEvent()

18.5.22.19 Class data for QGraphicsSceneMouseEvent

The virtual table for the QGraphicsSceneMouseEvent class is described by Table 18-516

Table 18-516 Primary vtable for QGraphicsSceneMouseEvent

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneMouseEvent
vfunc[0]:	QGraphicsSceneMouseEvent::~~QGraphicsSceneMouseEvent()
vfunc[1]:	QGraphicsSceneMouseEvent::~~QGraphicsSceneMouseEvent()

18.5.22.20 Class data for QGraphicsSceneWheelEvent

The virtual table for the QGraphicsSceneWheelEvent class is described by Table 18-517

Table 18-517 Primary vtable for QGraphicsSceneWheelEvent

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSceneWheelEvent
vfunc[0]:	QGraphicsSceneWheelEvent::~~QGraphicsSceneWheelEvent()
vfunc[1]:	QGraphicsSceneWheelEvent::~~QGraphicsSceneWheelEvent()

18.5.22.21 Class data for QGraphicsSimpleTextItem

The virtual table for the QGraphicsSimpleTextItem class is described by Table 18-518

Table 18-518 Primary vtable for QGraphicsSimpleTextItem

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGraphicsSimpleTextItem
vfunc[0]:	QGraphicsSimpleTextItem::~~QGraphicsSimpleTextItem()
vfunc[1]:	QGraphicsSimpleTextItem::~~QGraphicsSimpleTextItem()
vfunc[2]:	QGraphicsItem::advance(int)
vfunc[3]:	QGraphicsSimpleTextItem::boundingRect() const
vfunc[4]:	QGraphicsSimpleTextItem::shape() const
vfunc[5]:	QGraphicsSimpleTextItem::contains(QPointF const&) const

vfunc[6]:	QGraphicsItem::collidesWithItem(QGraphicsItem const*, Qt::ItemSelectionMode) const
vfunc[7]:	QGraphicsItem::collidesWithPath(QPainterPath const&, Qt::ItemSelectionMode) const
vfunc[8]:	QGraphicsSimpleTextItem::isObscuredBy(QGraphicsItem const*) const
vfunc[9]:	QGraphicsSimpleTextItem::opaqueArea() const
vfunc[10]:	QGraphicsSimpleTextItem::paint(QPainter*, QStyleOptionGraphicsItem const*, QWidget*)
vfunc[11]:	QGraphicsSimpleTextItem::type() const
vfunc[12]:	QGraphicsItem::sceneEventFilter(QGraphicsItem*, QEvent*)
vfunc[13]:	QGraphicsItem::sceneEvent(QEvent*)
vfunc[14]:	QGraphicsItem::contextMenuEvent(QGraphicsSceneContextMenuEvent*)
vfunc[15]:	QGraphicsItem::dragEnterEvent(QGraphicsSceneDragDropEvent*)
vfunc[16]:	QGraphicsItem::dragLeaveEvent(QGraphicsSceneDragDropEvent*)
vfunc[17]:	QGraphicsItem::dragMoveEvent(QGraphicsSceneDragDropEvent*)
vfunc[18]:	QGraphicsItem::dropEvent(QGraphicsSceneDragDropEvent*)
vfunc[19]:	QGraphicsItem::focusInEvent(QFocusEvent*)
vfunc[20]:	QGraphicsItem::focusOutEvent(QFocusEvent*)
vfunc[21]:	QGraphicsItem::hoverEnterEvent(QGraphicsSceneHoverEvent*)
vfunc[22]:	QGraphicsItem::hoverMoveEvent(QGraphicsSceneHoverEvent*)
vfunc[23]:	QGraphicsItem::hoverLeaveEvent(QGraphicsSceneHoverEvent*)
vfunc[24]:	QGraphicsItem::keyPressEvent(QKeyEvent*)

vfunc[25]:	QGraphicsItem::keyPressEvent(QKeyEvent*)
vfunc[26]:	QGraphicsItem::mousePressEvent(QGraphicsSceneMouseEvent*)
vfunc[27]:	QGraphicsItem::mouseMoveEvent(QGraphicsSceneMouseEvent*)
vfunc[28]:	QGraphicsItem::mouseReleaseEvent(QGraphicsSceneMouseEvent*)
vfunc[29]:	QGraphicsItem::mouseDoubleClickEvent(QGraphicsSceneMouseEvent*)
vfunc[30]:	QGraphicsItem::wheelEvent(QGraphicsSceneWheelEvent*)
vfunc[31]:	QGraphicsItem::inputMethodEvent(QInputMethodEvent*)
vfunc[32]:	QGraphicsItem::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[33]:	QGraphicsItem::itemChange(QGraphicsItem::GraphicsItemChange, QVariant const&)
vfunc[34]:	QGraphicsSimpleTextItem::supportsExtension(QGraphicsItem::Extension) const
vfunc[35]:	QGraphicsSimpleTextItem::setExtension(QGraphicsItem::Extension, QVariant const&)
vfunc[36]:	QGraphicsSimpleTextItem::extension(QVariant const&) const

18.5.22.22 Interfaces for Qt4 2D Graphics

An LSB conforming implementation shall provide the generic functions for Qt4 2D Graphics specified in Table 18-519, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-519 libQtGui - Qt4 2D Graphics Function Interfaces

_Z13qDrawWinPanelP8QPainteriiiiRK8QPalettebPK6QBrush [QtGui]	_Z14qDrawPlainRectP8QPainterRK5QRectRK6QColoriPK6QBrush [QtGui]
_Z14qDrawPlainRectP8QPainteriiiiRK6QColoriPK6QBrush [QtGui]	_Z14qDrawShadeLineP8QPainterRK6QPointS3_RK8QPalettebii [QtGui]
_Z14qDrawShadeLineP8QPainteriiiiRK8QPalettebii [QtGui]	_Z14qDrawShadeRectP8QPainterRK5QRectRK8QPalettebiiPK6QBrush [QtGui]

_Z14qDrawShadeRectP8QPainteriii RK8QPalettedbiiPK6QBrush [QtGui]	_Z14qDrawWinButtonP8QPainterRK 5QRectRK8QPalettedbPK6QBrush [QtGui]
_Z14qDrawWinButtonP8QPainteriii RK8QPalettedbPK6QBrush [QtGui]	_Z15qDrawShadePanelP8QPainterR K5QRectRK8QPalettedbPK6QBrush [QtGui]
_Z15qDrawShadePanelP8QPainteriii RK8QPalettedbPK6QBrush [QtGui]	_ZN11QIconEngine10actualSizeERK 5QSizeN5QIcon4ModeENS3_5StateE [QtGui]
_ZN11QIconEngine6pixmapERK5QS izeN5QIcon4ModeENS3_5StateE [QtGui]	_ZN11QIconEngine7addFileERK7QS tringRK5QSizeN5QIcon4ModeENS6 _5StateE [QtGui]
_ZN11QIconEngine9addPixmapERK 7QPixmapN5QIcon4ModeENS3_5Sta teE [QtGui]	_ZN11QIconEngineD0Ev [QtGui]
_ZN11QIconEngineD1Ev [QtGui]	_ZN11QIconEngineD2Ev [QtGui]
_ZN12QPaintEngine10drawPointsEP K6QPointi [QtGui]	_ZN12QPaintEngine10drawPointsEP K7QPointFi [QtGui]
_ZN12QPaintEngine11drawEllipseE RK5QRect [QtGui]	_ZN12QPaintEngine11drawEllipseE RK6QRectF [QtGui]
_ZN12QPaintEngine11drawPolygon EPK6QPointiNS_15PolygonDrawMo deE [QtGui]	_ZN12QPaintEngine11drawPolygon EPK7QPointFiNS_15PolygonDrawM odeE [QtGui]
_ZN12QPaintEngine12drawTextItem ERK7QPointFRK9QTextItem [QtGui]	_ZN12QPaintEngine13setSystemClip ERK7QRegion [LSB]
_ZN12QPaintEngine13setSystemRect ERK5QRect [LSB]	_ZN12QPaintEngine14setPaintDevic eEP12QPaintDevice [LSB]
_ZN12QPaintEngine15drawTiledPix mapERK6QRectFRK7QPixmapRK7Q PointF [QtGui]	_ZN12QPaintEngine8drawPathERK1 2QPainterPath [QtGui]
_ZN12QPaintEngine9drawImageER K6QRectFRK6QImageS2_6QFlagsIN 2Qt19ImageConversionFlagEE [QtGui]	_ZN12QPaintEngine9drawLinesEPK 5QLinei [QtGui]
_ZN12QPaintEngine9drawLinesEPK 6QLineFi [QtGui]	_ZN12QPaintEngine9drawRectsEPK 5QRecti [QtGui]
_ZN12QPaintEngine9drawRectsEPK 6QRectFi [QtGui]	_ZN12QPaintEngineC1E6QFlagsINS _18PaintEngineFeatureEE [QtGui]
_ZN12QPaintEngineC2E6QFlagsINS _18PaintEngineFeatureEE [QtGui]	_ZN12QPaintEngineD0Ev [QtGui]
_ZN12QPaintEngineD1Ev [QtGui]	_ZN12QPaintEngineD2Ev [QtGui]
_ZN12QPainterPath10addEllipseER K6QRectF [QtGui]	_ZN12QPainterPath10addPolygonER K9QPolygonF [QtGui]

ZN12QPainterPath11connectPathERKS [QtGui]	_ZN12QPainterPath11setFillRuleEN2Qt8FillRuleE [QtGui]
_ZN12QPainterPath12closeSubpathEv [QtGui]	_ZN12QPainterPath5arcToERK6QRectFdd [QtGui]
_ZN12QPainterPath6lineToERK7QPointF [QtGui]	_ZN12QPainterPath6moveToERK7QPointF [QtGui]
ZN12QPainterPath6quadToERK7QPointF5S2 [QtGui]	_ZN12QPainterPath7addPathERKS_ [QtGui]
_ZN12QPainterPath7addRectERK6QRectF [QtGui]	_ZN12QPainterPath7addTextERK7QPointF5RK5QFontRK7QString [QtGui]
_ZN12QPainterPath7cubicToERK7QPointF5S2_S2_ [QtGui]	_ZN12QPainterPath9addRegionERK7QRegion [QtGui]
_ZN12QPainterPath9arcMoveToERK6QRectFd [QtXml]	_ZN12QPainterPathC1ERK7QPointF [QtGui]
ZN12QPainterPathC1ERKS [QtGui]	_ZN12QPainterPathC1Ev [QtGui]
_ZN12QPainterPathC2ERK7QPointF [QtGui]	_ZN12QPainterPathC2ERKS_ [QtGui]
_ZN12QPainterPathC2Ev [QtGui]	_ZN12QPainterPathD1Ev [QtGui]
_ZN12QPainterPathD2Ev [QtGui]	_ZN12QPainterPathaSERKS_ [QtGui]
_ZN13QGraphicsItem10addToIndexEv [LSB]	_ZN13QGraphicsItem10clearFocusEv [QtXml]
_ZN13QGraphicsItem10itemChangeENS_18GraphicsItemChangeERK8QVariant [QtXml]	_ZN13QGraphicsItem10sceneEventEP6QEvent [QtXml]
_ZN13QGraphicsItem10setEnabledEb [QtXml]	_ZN13QGraphicsItem10setToolTipERK7QString [QtXml]
_ZN13QGraphicsItem10setVisibleEb [QtXml]	_ZN13QGraphicsItem10wheelEventEP24QGraphicsSceneWheelEvent [QtXml]
_ZN13QGraphicsItem11resetMatrixEv [QtXml]	_ZN13QGraphicsItem11setSelectedEb [QtXml]
_ZN13QGraphicsItem11unsetCursorEv [QtXml]	_ZN13QGraphicsItem12focusInEventEP11QFocusEvent [QtXml]
_ZN13QGraphicsItem12setExtensionENS_9ExtensionERK8QVariant [LSB]	_ZN13QGraphicsItem13dragMoveEventEP27QGraphicsSceneDragDropEvent [QtXml]
_ZN13QGraphicsItem13ensureVisibleERK6QRectFii [QtXml]	_ZN13QGraphicsItem13focusOutEventEP11QFocusEvent [QtXml]
_ZN13QGraphicsItem13keyPressEventEP9QKeyEvent [QtXml]	_ZN13QGraphicsItem13setParentItemEPS_ [QtXml]

_ZN13QGraphicsItem14dragEnterEventEP27QGraphicsSceneDragDropEvent [QtXml]	_ZN13QGraphicsItem14dragLeaveEventEP27QGraphicsSceneDragDropEvent [QtXml]
_ZN13QGraphicsItem14hoverMoveEventEP24QGraphicsSceneHoverEvent [QtXml]	_ZN13QGraphicsItem14mouseMoveEventEP24QGraphicsSceneMouseEvent [QtXml]
_ZN13QGraphicsItem14setAcceptDropsEb [QtXml]	_ZN13QGraphicsItem15hoverEnterEventEP24QGraphicsSceneHoverEvent [QtXml]
_ZN13QGraphicsItem15hoverLeaveEventEP24QGraphicsSceneHoverEvent [QtXml]	_ZN13QGraphicsItem15keyReleaseEventEP9QKeyEvent [QtXml]
_ZN13QGraphicsItem15mousePressEventEP24QGraphicsSceneMouseEvent [QtXml]	_ZN13QGraphicsItem15removeFromIndexEv [LSB]
_ZN13QGraphicsItem16contextMenuEventEP30QGraphicsSceneContextMenuEvent [QtXml]	_ZN13QGraphicsItem16inputMethodEventEP17QInputMethodEvent [QtXml]
_ZN13QGraphicsItem16sceneEventFilterEPS_P6QEvent [QtXml]	_ZN13QGraphicsItem17mouseReleaseEventEP24QGraphicsSceneMouseEvent [QtXml]
_ZN13QGraphicsItem21mouseDoubleClickEventEP24QGraphicsSceneMouseEvent [QtXml]	_ZN13QGraphicsItem21prepareGeometryChangeEv [QtXml]
_ZN13QGraphicsItem21setAcceptsHoverEventsEb [QtXml]	_ZN13QGraphicsItem21setHandlesChildEventsEb [QtXml]
ZN13QGraphicsItem22removeSceneEventFilterEPS [QtXml]	_ZN13QGraphicsItem23installSceneEventFilterEPS_ [QtXml]
_ZN13QGraphicsItem23setAcceptedMouseButtonsE6QFlagsIN2Qt11MouseButtonEE [QtXml]	_ZN13QGraphicsItem5scaleEdd [QtXml]
_ZN13QGraphicsItem5shearEdd [QtXml]	_ZN13QGraphicsItem6rotateEd [QtXml]
_ZN13QGraphicsItem6setPosERK7QPointF [QtXml]	_ZN13QGraphicsItem6updateERK6QRectF [QtXml]
_ZN13QGraphicsItem7advanceEi [QtXml]	_ZN13QGraphicsItem7setDataEiRK8QVariant [QtXml]
_ZN13QGraphicsItem7setFlagENS_16GraphicsItemFlagEb [QtXml]	_ZN13QGraphicsItem8setFlagsE6QFlagsINS_16GraphicsItemFlagEE [QtXml]
_ZN13QGraphicsItem8setFocusEN2Qt11FocusReasonE [QtXml]	_ZN13QGraphicsItem8setGroupEP18QGraphicsItemGroup [QtXml]
_ZN13QGraphicsItem9dropEventEP27QGraphicsSceneDragDropEvent [QtXml]	_ZN13QGraphicsItem9setCursorERK7QCursor [QtXml]

_ZN13QGraphicsItem9setMatrixERK7QMatrixb [QtXml]	_ZN13QGraphicsItem9setZValueEd [QtXml]
_ZN13QGraphicsItem9translateEdd [QtXml]	_ZN13QGraphicsItemC1EPS_P14QGraphicsScene [QtXml]
_ZN13QGraphicsItemC2EPS_P14QGraphicsScene [QtXml]	_ZN13QGraphicsItemD0Ev [QtXml]
_ZN13QGraphicsItemD1Ev [QtXml]	_ZN13QGraphicsItemD2Ev [QtXml]
_ZN13QGraphicsView10paintEventEP11QPaintEvent [QtXml]	_ZN13QGraphicsView10wheelEventEP11QWheelEvent [QtXml]
_ZN13QGraphicsView11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]	_ZN13QGraphicsView11qt_metacastEPKc [QtXml]
_ZN13QGraphicsView11resetMatrixEv [QtXml]	_ZN13QGraphicsView11resizeEventEP12QResizeEvent [QtXml]
_ZN13QGraphicsView11setDragModeENS_8DragModeE [QtXml]	_ZN13QGraphicsView11updateSceneERK5QListI6QRectFE [QtXml]
_ZN13QGraphicsView12focusInEventEP11QFocusEvent [QtXml]	_ZN13QGraphicsView12setAlignmentE6QFlagsIN2Qt13AlignmentFlagEE [QtXml]
_ZN13QGraphicsView12setCacheModeE6QFlagsINS_13CacheModeFlagEE [QtXml]	_ZN13QGraphicsView12setSceneRectERK6QRectF [QtXml]
_ZN13QGraphicsView13dragMoveEventEP14QDragMoveEvent [QtXml]	_ZN13QGraphicsView13ensureVisibleEPK13QGraphicsItemii [QtXml]
_ZN13QGraphicsView13ensureVisibleERK6QRectFii [QtXml]	_ZN13QGraphicsView13focusOutEventEP11QFocusEvent [QtXml]
_ZN13QGraphicsView13keyPressEventEP9QKeyEvent [QtXml]	_ZN13QGraphicsView13setRenderHintEN8QPainter10RenderHintEb [QtXml]
_ZN13QGraphicsView13setupViewportEP7QWidget [QtXml]	_ZN13QGraphicsView13viewportEventEP6QEvent [QtXml]
_ZN13QGraphicsView14dragEnterEventEP15QDragEnterEvent [QtXml]	_ZN13QGraphicsView14dragLeaveEventEP15QDragLeaveEvent [QtXml]
_ZN13QGraphicsView14drawBackgroudEP8QPainterRK6QRectF [QtXml]	_ZN13QGraphicsView14drawForegroundEP8QPainterRK6QRectF [QtXml]
_ZN13QGraphicsView14mouseMoveEventEP11QMouseEvent [QtXml]	_ZN13QGraphicsView14setInteractiveEb [QtXml]
_ZN13QGraphicsView14setRenderHintsE6QFlagsIN8QPainter10RenderHintEE [QtXml]	_ZN13QGraphicsView15keyReleaseEventEP9QKeyEvent [QtXml]

_ZN13QGraphicsView15mousePressEventEP11QMouseEvent [QtXml]	_ZN13QGraphicsView15setResizeAnchorENS_14ViewportAnchorE [QtXml]
_ZN13QGraphicsView15updateSceneRectERK6QRectF [QtXml]	_ZN13QGraphicsView16contextMenuEventEP17QContextMenuEvent [QtXml]
_ZN13QGraphicsView16inputMethodEventEP17QInputMethodEvent [QtXml]	_ZN13QGraphicsView16scrollContentsByEii [QtXml]
_ZN13QGraphicsView17mouseReleaseEventEP11QMouseEvent [QtXml]	_ZN13QGraphicsView18resetCachedContentEv [QtXml]
_ZN13QGraphicsView18setBackgroundBrushERK6QBrush [QtXml]	_ZN13QGraphicsView18setForegroundBrushERK6QBrush [QtXml]
_ZN13QGraphicsView21mouseDoubleClickEventEP11QMouseEvent [QtXml]	_ZN13QGraphicsView23setTransformationAnchorENS_14ViewportAnchorE [QtXml]
_ZN13QGraphicsView5eventEP6QEvent [QtXml]	_ZN13QGraphicsView5scaleEdd [QtXml]
_ZN13QGraphicsView5shearEdd [QtXml]	_ZN13QGraphicsView6renderEP8QPainterRK6QRectFRK5QRectN2Qt15AspectRatioModeE [QtXml]
_ZN13QGraphicsView6rotateEd [QtXml]	_ZN13QGraphicsView8centerOnEPK13QGraphicsItem [QtXml]
_ZN13QGraphicsView8centerOnERK7QPointF [QtXml]	_ZN13QGraphicsView8setSceneEP14QGraphicsScene [QtXml]
_ZN13QGraphicsView9drawItemsEP8QPainteriPP13QGraphicsItemPK24QStyleOptionGraphicsItem [QtXml]	_ZN13QGraphicsView9dropEventEP10QDropEvent [QtXml]
_ZN13QGraphicsView9fitInViewEPK13QGraphicsItemN2Qt15AspectRatioModeE [QtXml]	_ZN13QGraphicsView9fitInViewERK6QRectFN2Qt15AspectRatioModeE [QtXml]
_ZN13QGraphicsView9setMatrixERK7QMatrixb [QtXml]	_ZN13QGraphicsView9showEventEP10QShowEvent [QtXml]
_ZN13QGraphicsView9translateEdd [QtXml]	_ZN13QGraphicsViewC1EP14QGraphicsSceneP7QWidget [QtXml]
_ZN13QGraphicsViewC1EP7QWidget [QtXml]	_ZN13QGraphicsViewC2EP14QGraphicsSceneP7QWidget [QtXml]
_ZN13QGraphicsViewC2EP7QWidget [QtXml]	_ZN13QGraphicsViewD0Ev [QtXml]
_ZN13QGraphicsViewD1Ev [QtXml]	_ZN13QGraphicsViewD2Ev [QtXml]
_ZN14QGraphicsScene10addEllipseERK6QRectFRK4QPenRK6QBrush [QtXml]	_ZN14QGraphicsScene10addPolygonERK9QPolygonFRK4QPenRK6QBrush [QtXml]

_ZN14QGraphicsScene10clearFocusEvent [QtXml]	_ZN14QGraphicsScene10removeItemEP13QGraphicsItem [QtXml]
_ZN14QGraphicsScene10wheelEventEP24QGraphicsSceneWheelEvent [QtXml]	_ZN14QGraphicsScene11qt_metacallEN11QMetaObject4CallEiPPv [QtXml]
_ZN14QGraphicsScene11qt_metacastEPKc [QtXml]	_ZN14QGraphicsScene12focusInEventEP11QFocusEvent [QtXml]
_ZN14QGraphicsScene12setFocusItemEP13QGraphicsItemN2Qt11FocusReasonE [QtXml]	_ZN14QGraphicsScene12setSceneRectERK6QRectF [QtXml]
_ZN14QGraphicsScene13dragMoveEventEP27QGraphicsSceneDragDropEvent [QtXml]	_ZN14QGraphicsScene13focusOutEventEP11QFocusEvent [QtXml]
_ZN14QGraphicsScene13keyPressEventEP9QKeyEvent [QtXml]	_ZN14QGraphicsScene14clearSelectionEv [QtXml]
_ZN14QGraphicsScene14dragEnterEventEP27QGraphicsSceneDragDropEvent [QtXml]	_ZN14QGraphicsScene14dragLeaveEventEP27QGraphicsSceneDragDropEvent [QtXml]
_ZN14QGraphicsScene14drawBackgroundEP8QPainterRK6QRectF [QtXml]	_ZN14QGraphicsScene14drawForegroundEP8QPainterRK6QRectF [QtXml]
_ZN14QGraphicsScene14mouseMoveEventEP24QGraphicsSceneMouseEvent [QtXml]	_ZN14QGraphicsScene15createItemGroupERK5QListIP13QGraphicsItemE [QtXml]
_ZN14QGraphicsScene15keyReleaseEventEP9QKeyEvent [QtXml]	_ZN14QGraphicsScene15mousePressEventEP24QGraphicsSceneMouseEvent [QtXml]
_ZN14QGraphicsScene16contextMenuEventEP30QGraphicsSceneContextMenuEvent [QtXml]	_ZN14QGraphicsScene16destroyItemGroupEP18QGraphicsItemGroup [QtXml]
_ZN14QGraphicsScene16inputMethodEventEP17QInputMethodEvent [QtXml]	_ZN14QGraphicsScene16sceneRectChangedERK6QRectF [QtXml]
_ZN14QGraphicsScene16setSelectionAreaERK12QPainterPath [QtXml]	_ZN14QGraphicsScene17mouseReleaseEventEP24QGraphicsSceneMouseEvent [QtXml]
_ZN14QGraphicsScene18setBackgroundBrushERK6QBrush [QtXml]	_ZN14QGraphicsScene18setForegroundBrushERK6QBrush [QtXml]
_ZN14QGraphicsScene18setItemIndexMethodENS_15ItemIndexMethodE [QtXml]	_ZN14QGraphicsScene21mouseDoubleClickEventEP24QGraphicsSceneMouseEvent [QtXml]
_ZN14QGraphicsScene5eventEP6QEvent [QtXml]	_ZN14QGraphicsScene6renderEP8QPainterRK6QRectFS4_N2Qt15AspectRatioModeE [QtXml]

_ZN14QGraphicsScene6updateERK6QRectF [QtXml]	_ZN14QGraphicsScene7addItemEP13QGraphicsItem [QtXml]
_ZN14QGraphicsScene7addLineERK6QLineFRK4QPen [QtXml]	_ZN14QGraphicsScene7addPathERK12QPainterPathRK4QPenRK6QBrush [QtXml]
_ZN14QGraphicsScene7addRectERK6QRectFRK4QPenRK6QBrush [QtXml]	_ZN14QGraphicsScene7addTextERK7QStringRK5QFont [QtXml]
_ZN14QGraphicsScene7advanceEv [QtXml]	_ZN14QGraphicsScene7changedERK5QListI6QRectFE [QtXml]
_ZN14QGraphicsScene8setFocusEN2Qt11FocusReasonE [QtXml]	_ZN14QGraphicsScene9addPixmapERK7QPixmap [QtXml]
_ZN14QGraphicsScene9drawItemsEP8QPainterIP13QGraphicsItemPK24QStyleOptionGraphicsItemP7QWidget [QtXml]	_ZN14QGraphicsScene9dropEventEP27QGraphicsSceneDragDropEvent [QtXml]
_ZN14QGraphicsScene9helpEventEP23QGraphicsSceneHelpEvent [QtXml]	_ZN14QGraphicsSceneC1EP7QObject [QtXml]
_ZN14QGraphicsSceneC1ERK6QRectFP7QObject [QtXml]	_ZN14QGraphicsSceneC1EddddP7QObject [QtXml]
_ZN14QGraphicsSceneC2EP7QObject [QtXml]	_ZN14QGraphicsSceneC2ERK6QRectFP7QObject [QtXml]
_ZN14QGraphicsSceneC2EddddP7QObject [QtXml]	_ZN14QGraphicsSceneD0Ev [QtXml]
_ZN14QGraphicsSceneD1Ev [QtXml]	_ZN14QGraphicsSceneD2Ev [QtXml]
_ZN15QLinearGradient12setFinalStopERK7QPointF [QtXml]	_ZN15QLinearGradient8setStartERK7QPointF [QtXml]
ZN15QLinearGradientC1ERK7QPointF5S2 [QtGui]	_ZN15QLinearGradientC1Edddd [QtGui]
_ZN15QLinearGradientC1Ev [QtXml]	_ZN15QLinearGradientC2ERK7QPointF5S2_ [QtGui]
_ZN15QLinearGradientC2Edddd [QtGui]	_ZN15QLinearGradientC2Ev [QtXml]
_ZN15QRadialGradient13setFocalPointERK7QPointF [QtXml]	_ZN15QRadialGradient9setCenterERK7QPointF [QtXml]
_ZN15QRadialGradient9setRadiusEd [QtXml]	_ZN15QRadialGradientC1ERK7QPointF5S2_ [QtXml]
ZN15QRadialGradientC1ERK7QPointF5S2 [QtGui]	_ZN15QRadialGradientC1Eddd [QtXml]
_ZN15QRadialGradientC1Edddd [QtGui]	_ZN15QRadialGradientC1Ev [QtXml]

_ZN15QRadialGradientC2ERK7QPointFd [QtXml]	_ZN15QRadialGradientC2ERK7QPointFdS2_ [QtGui]
_ZN15QRadialGradientC2Eddd [QtXml]	_ZN15QRadialGradientC2Edddd [QtGui]
_ZN15QRadialGradientC2Ev [QtXml]	_ZN16QConicalGradient8setAngleEd [QtXml]
_ZN16QConicalGradient9setCenterERK7QPointF [QtXml]	_ZN16QConicalGradientC1ERK7QPointFd [QtGui]
_ZN16QConicalGradientC1Eddd [QtGui]	_ZN16QConicalGradientC1Ev [QtXml]
_ZN16QConicalGradientC2ERK7QPointFd [QtGui]	_ZN16QConicalGradientC2Eddd [QtGui]
_ZN16QConicalGradientC2Ev [QtXml]	_ZN17QGraphicsLineItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]
_ZN17QGraphicsLineItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]	_ZN17QGraphicsLineItem6setPenERK4QPen [QtXml]
_ZN17QGraphicsLineItem7setLineERK6QLineF [QtXml]	_ZN17QGraphicsLineItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsLineItemC1ERK6QLineFP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsLineItemC1EdddP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsLineItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsLineItemC2ERK6QLineFP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsLineItemC2EdddP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsLineItemD0Ev [QtXml]
_ZN17QGraphicsLineItemD1Ev [QtXml]	_ZN17QGraphicsLineItemD2Ev [QtXml]
_ZN17QGraphicsPathItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]	_ZN17QGraphicsPathItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]
_ZN17QGraphicsPathItem7setPathERK12QPainterPath [QtXml]	_ZN17QGraphicsPathItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsPathItemC1ERK12QPainterPathP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsPathItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsPathItemC2ERK12QPainterPathP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsPathItemD0Ev [QtXml]

_ZN17QGraphicsPathItemD1Ev [QtXml]	_ZN17QGraphicsPathItemD2Ev [QtXml]
_ZN17QGraphicsRectItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]	_ZN17QGraphicsRectItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]
_ZN17QGraphicsRectItem7setRectERK6QRectF [QtXml]	_ZN17QGraphicsRectItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsRectItemC1ERK6QRectFP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsRectItemC1EddddP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsRectItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsRectItemC2ERK6QRectFP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsRectItemC2EddddP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsRectItemD0Ev [QtXml]
_ZN17QGraphicsRectItemD1Ev [QtXml]	_ZN17QGraphicsRectItemD2Ev [QtXml]
_ZN17QGraphicsTextItem10adjustSizeEv [QtXml]	_ZN17QGraphicsTextItem10sceneEventEP6QEvent [QtXml]
_ZN17QGraphicsTextItem11linkHoveredERK7QString [QtXml]	_ZN17QGraphicsTextItem11qt_metaCallEN11QMetaObject4CallEiPPv [QtXml]
_ZN17QGraphicsTextItem11qt_metaCastEPKc [QtXml]	_ZN17QGraphicsTextItem11setDocumentEP13QTextDocument [QtXml]
_ZN17QGraphicsTextItem12focusInEventEP11QFocusEvent [QtXml]	_ZN17QGraphicsTextItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]
_ZN17QGraphicsTextItem12setPlainTextERK7QString [QtXml]	_ZN17QGraphicsTextItem12setTextWidthEd [QtXml]
_ZN17QGraphicsTextItem13dragMoveEventEP27QGraphicsSceneDragDropEvent [QtXml]	_ZN17QGraphicsTextItem13focusOutEventEP11QFocusEvent [QtXml]
_ZN17QGraphicsTextItem13keyPressEventEP9QKeyEvent [QtXml]	_ZN17QGraphicsTextItem13linkActivatedERK7QString [QtXml]
_ZN17QGraphicsTextItem13setTextCursorERK11QTextCursor [QtXml]	_ZN17QGraphicsTextItem14dragEnterEventEP27QGraphicsSceneDragDropEvent [QtXml]
_ZN17QGraphicsTextItem14dragLeaveEventEP27QGraphicsSceneDragDropEvent [QtXml]	_ZN17QGraphicsTextItem14hoverMoveEventEP24QGraphicsSceneHoverEvent [QtXml]
_ZN17QGraphicsTextItem14mouseMoveEventEP24QGraphicsSceneMouseEvent [QtXml]	_ZN17QGraphicsTextItem15hoverEnterEventEP24QGraphicsSceneHoverEvent [QtXml]

_ZN17QGraphicsTextItem15hoverLeaveEventEP24QGraphicsSceneHoverEvent [QtXml]	_ZN17QGraphicsTextItem15keyReleaseEventEP9QKeyEvent [QtXml]
_ZN17QGraphicsTextItem15mousePressEventEP24QGraphicsSceneMouseEvent [QtXml]	_ZN17QGraphicsTextItem16contextMenuEventEP30QGraphicsSceneContextMenuEvent [QtXml]
_ZN17QGraphicsTextItem16inputMethodEventEP17QInputMethodEvent [QtXml]	_ZN17QGraphicsTextItem17mouseReleaseEventEP24QGraphicsSceneMouseEvent [QtXml]
_ZN17QGraphicsTextItem19setDefaultTextColorERK6QColor [QtXml]	_ZN17QGraphicsTextItem20setOpenExternalLinksEb [QtXml]
_ZN17QGraphicsTextItem21mouseDoubleClickEventEP24QGraphicsSceneMouseEvent [QtXml]	_ZN17QGraphicsTextItem23setTextInteractionFlagsE6QFlagsIN2Qt19TextInteractionFlagEE [QtXml]
_ZN17QGraphicsTextItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]	_ZN17QGraphicsTextItem7setFontERK5QFont [QtXml]
_ZN17QGraphicsTextItem7setHtmlERK7QString [QtXml]	_ZN17QGraphicsTextItem9dropEventEP27QGraphicsSceneDragDropEvent [QtXml]
_ZN17QGraphicsTextItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsTextItemC1ERK7QStringP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsTextItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN17QGraphicsTextItemC2ERK7QStringP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN17QGraphicsTextItemD0Ev [QtXml]	_ZN17QGraphicsTextItemD1Ev [QtXml]
_ZN17QGraphicsTextItemD2Ev [QtXml]	_ZN18QGraphicsItemGroup10addToGroupEP13QGraphicsItem [QtXml]
_ZN18QGraphicsItemGroup15removeFromGroupEP13QGraphicsItem [QtXml]	_ZN18QGraphicsItemGroup5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]
_ZN18QGraphicsItemGroupC1EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN18QGraphicsItemGroupC2EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN18QGraphicsItemGroupD0Ev [QtXml]	_ZN18QGraphicsItemGroupD1Ev [QtXml]
_ZN18QGraphicsItemGroupD2Ev [QtXml]	_ZN19QGraphicsPixmapItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]
_ZN19QGraphicsPixmapItem12setShapeModeENS_9ShapeModeE [QtXml]	_ZN19QGraphicsPixmapItem21setTransformationModeEN2Qt18TransformationModeE [QtXml]

_ZN19QGraphicsPixmapItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]	_ZN19QGraphicsPixmapItem9setOffsetERK7QPointF [QtXml]
_ZN19QGraphicsPixmapItem9setPixmapERK7QPixmap [QtXml]	_ZN19QGraphicsPixmapItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN19QGraphicsPixmapItemC1ERK7QPixmapP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN19QGraphicsPixmapItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN19QGraphicsPixmapItemC2ERK7QPixmapP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN19QGraphicsPixmapItemD0Ev [QtXml]
_ZN19QGraphicsPixmapItemD1Ev [QtXml]	_ZN19QGraphicsPixmapItemD2Ev [QtXml]
_ZN19QGraphicsSceneEvent9setWidthEP7QWidget [LSB]	_ZN19QGraphicsSceneEventC1EN6QEvent4TypeE [LSB]
_ZN19QGraphicsSceneEventC2EN6QEvent4TypeE [LSB]	_ZN19QGraphicsSceneEventD0Ev [QtXml]
_ZN19QGraphicsSceneEventD1Ev [QtXml]	_ZN19QGraphicsSceneEventD2Ev [QtXml]
_ZN19QPainterPathStroker11setCapStyleEN2Qt11PenCapStyleE [QtGui]	_ZN19QPainterPathStroker12setJoinStyleEN2Qt12PenJoinStyleE [QtGui]
_ZN19QPainterPathStroker13setMiterLimitEd [QtGui]	_ZN19QPainterPathStroker14setDashPatternEN2Qt8PenStyleE [QtGui]
_ZN19QPainterPathStroker14setDashPatternERK7QVectorIdE [QtGui]	_ZN19QPainterPathStroker17setCurveThresholdEd [QtGui]
_ZN19QPainterPathStroker8setWidthEd [QtGui]	_ZN19QPainterPathStrokerC1Ev [QtGui]
_ZN19QPainterPathStrokerC2Ev [QtGui]	_ZN19QPainterPathStrokerD1Ev [QtGui]
_ZN19QPainterPathStrokerD2Ev [QtGui]	_ZN20QGraphicsEllipseItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]
_ZN20QGraphicsEllipseItem12setSpanAngleEi [QtXml]	_ZN20QGraphicsEllipseItem13setStartAngleEi [QtXml]
_ZN20QGraphicsEllipseItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]	_ZN20QGraphicsEllipseItem7setRectERK6QRectF [QtXml]
_ZN20QGraphicsEllipseItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN20QGraphicsEllipseItemC1ERK6QRectFP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN20QGraphicsEllipseItemC1EdddP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN20QGraphicsEllipseItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]

_ZN20QGraphicsEllipseItemC2ERK6QRectFP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN20QGraphicsEllipseItemC2EdddP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN20QGraphicsEllipseItemD0Ev [QtXml]	_ZN20QGraphicsEllipseItemD1Ev [QtXml]
_ZN20QGraphicsEllipseItemD2Ev [QtXml]	_ZN20QGraphicsPolygonItem10setPolygonERK9QPolygonF [QtXml]
_ZN20QGraphicsPolygonItem11setFillRuleEN2Qt8FillRuleE [QtXml]	_ZN20QGraphicsPolygonItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]
_ZN20QGraphicsPolygonItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]	_ZN20QGraphicsPolygonItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN20QGraphicsPolygonItemC1ERK9QPolygonFP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN20QGraphicsPolygonItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN20QGraphicsPolygonItemC2ERK9QPolygonFP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN20QGraphicsPolygonItemD0Ev [QtXml]
_ZN20QGraphicsPolygonItemD1Ev [QtXml]	_ZN20QGraphicsPolygonItemD2Ev [QtXml]
_ZN22QGraphicsItemAnimation10setScaleAtEddd [QtXml]	_ZN22QGraphicsItemAnimation10setShearAtEddd [QtXml]
_ZN22QGraphicsItemAnimation11qt_metacallEN11QMetaObject4CallEiPv [QtXml]	_ZN22QGraphicsItemAnimation11qt_metacastEPKc [QtXml]
_ZN22QGraphicsItemAnimation11setTimeLineEP9QTimeLine [QtXml]	_ZN22QGraphicsItemAnimation13setRotationAtEdd [QtXml]
_ZN22QGraphicsItemAnimation16setTranslationAtEddd [QtXml]	_ZN22QGraphicsItemAnimation18afterAnimationStepEd [QtXml]
_ZN22QGraphicsItemAnimation19beforeAnimationStepEd [QtXml]	_ZN22QGraphicsItemAnimation5clearEv [QtXml]
_ZN22QGraphicsItemAnimation5resetEv [QtXml]	_ZN22QGraphicsItemAnimation7setItemEP13QGraphicsItem [QtXml]
_ZN22QGraphicsItemAnimation7setStepEd [QtXml]	_ZN22QGraphicsItemAnimation8setPositionAtEdRK7QPointF [QtXml]
_ZN22QGraphicsItemAnimationC1EP7QObject [QtXml]	_ZN22QGraphicsItemAnimationC2EP7QObject [QtXml]
_ZN22QGraphicsItemAnimationD0Ev [QtXml]	_ZN22QGraphicsItemAnimationD1Ev [QtXml]
_ZN22QGraphicsItemAnimationD2Ev [QtXml]	_ZN23QGraphicsSceneHelpEvent11setScenePosERK7QPointF [LSB]

_ZN23QGraphicsSceneHelpEvent12setScreenPosERK6QPoint [LSB]	_ZN23QGraphicsSceneHelpEventC1EN6QEvent4TypeE [LSB]
_ZN23QGraphicsSceneHelpEventC2EN6QEvent4TypeE [LSB]	_ZN23QGraphicsSceneHelpEventD0Ev [QtXml]
_ZN23QGraphicsSceneHelpEventD1Ev [QtXml]	_ZN23QGraphicsSceneHelpEventD2Ev [QtXml]
_ZN23QGraphicsSimpleTextItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant [LSB]	_ZN23QGraphicsSimpleTextItem5paintEP8QPainterPK24QStyleOptionGraphicsItemP7QWidget [QtXml]
_ZN23QGraphicsSimpleTextItem7setFontERK5QFont [QtXml]	_ZN23QGraphicsSimpleTextItem7setTextERK7QString [QtXml]
_ZN23QGraphicsSimpleTextItemC1EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN23QGraphicsSimpleTextItemC1ERK7QStringP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN23QGraphicsSimpleTextItemC2EP13QGraphicsItemP14QGraphicsScene [QtXml]	_ZN23QGraphicsSimpleTextItemC2ERK7QStringP13QGraphicsItemP14QGraphicsScene [QtXml]
_ZN23QGraphicsSimpleTextItemD0Ev [QtXml]	_ZN23QGraphicsSimpleTextItemD1Ev [QtXml]
_ZN23QGraphicsSimpleTextItemD2Ev [QtXml]	_ZN24QGraphicsSceneHoverEvent11setScreenPosERK7QPointF [LSB]
_ZN24QGraphicsSceneHoverEvent12setScreenPosERK6QPoint [LSB]	_ZN24QGraphicsSceneHoverEvent6setPosERK7QPointF [LSB]
_ZN24QGraphicsSceneHoverEventC1EN6QEvent4TypeE [LSB]	_ZN24QGraphicsSceneHoverEventC2EN6QEvent4TypeE [LSB]
_ZN24QGraphicsSceneHoverEventD0Ev [QtXml]	_ZN24QGraphicsSceneHoverEventD1Ev [QtXml]
_ZN24QGraphicsSceneHoverEventD2Ev [QtXml]	_ZN24QGraphicsSceneMouseEvent10setButtonsE6QFlagsIN2Qt11MouseButtonEE [LSB]
_ZN24QGraphicsSceneMouseEvent10setLastPosERK7QPointF [LSB]	_ZN24QGraphicsSceneMouseEvent11setScreenPosERK7QPointF [LSB]
_ZN24QGraphicsSceneMouseEvent12setModifiersE6QFlagsIN2Qt16KeyboardModifierEE [LSB]	_ZN24QGraphicsSceneMouseEvent12setScreenPosERK6QPoint [LSB]
_ZN24QGraphicsSceneMouseEvent15setLastScenePosERK7QPointF [LSB]	_ZN24QGraphicsSceneMouseEvent16setButtonDownPosEN2Qt11MouseButtonERK7QPointF [LSB]
_ZN24QGraphicsSceneMouseEvent16setLastScreenPosERK6QPoint [LSB]	_ZN24QGraphicsSceneMouseEvent21setButtonDownScenePosEN2Qt11MouseButtonERK7QPointF [LSB]
_ZN24QGraphicsSceneMouseEvent22setButtonDownScreenPosEN2Qt11MouseButtonERK6QPoint [LSB]	_ZN24QGraphicsSceneMouseEvent6setPosERK7QPointF [LSB]

_ZN24QGraphicsSceneMouseEvent9 setButtonEN2Qt11MouseButtonE [LSB]	_ZN24QGraphicsSceneMouseEventC 1EN6QEvent4TypeE [LSB]
_ZN24QGraphicsSceneMouseEventC 2EN6QEvent4TypeE [LSB]	_ZN24QGraphicsSceneMouseEventD 0Ev [QtXml]
_ZN24QGraphicsSceneMouseEventD 1Ev [QtXml]	_ZN24QGraphicsSceneMouseEventD 2Ev [QtXml]
_ZN24QGraphicsSceneWheelEvent1 0setButtonsE6QFlagsIN2Qt11Mouse ButtonEE [LSB]	_ZN24QGraphicsSceneWheelEvent1 1setScenePosERK7QPointF [LSB]
_ZN24QGraphicsSceneWheelEvent1 2setModifiersE6QFlagsIN2Qt16Keyb oardModifierEE [LSB]	_ZN24QGraphicsSceneWheelEvent1 2setScreenPosERK6QPoint [LSB]
_ZN24QGraphicsSceneWheelEvent6s etPosERK7QPointF [LSB]	_ZN24QGraphicsSceneWheelEvent8s etDeltaEi [LSB]
_ZN24QGraphicsSceneWheelEventC 1EN6QEvent4TypeE [LSB]	_ZN24QGraphicsSceneWheelEventC 2EN6QEvent4TypeE [LSB]
_ZN24QGraphicsSceneWheelEventD 0Ev [QtXml]	_ZN24QGraphicsSceneWheelEventD 1Ev [QtXml]
_ZN24QGraphicsSceneWheelEventD 2Ev [QtXml]	_ZN27QGraphicsSceneDragDropEve nt10setButtonsE6QFlagsIN2Qt11Mou seButtonEE [LSB]
_ZN27QGraphicsSceneDragDropEve nt11setMimeDataEPK9QMimeData [LSB]	_ZN27QGraphicsSceneDragDropEve nt11setScenePosERK7QPointF [LSB]
_ZN27QGraphicsSceneDragDropEve nt12setModifiersE6QFlagsIN2Qt16K eyboardModifierEE [LSB]	_ZN27QGraphicsSceneDragDropEve nt12setScreenPosERK6QPoint [LSB]
_ZN27QGraphicsSceneDragDropEve nt13setDropActionEN2Qt10DropActi onE [QtXml]	_ZN27QGraphicsSceneDragDropEve nt17setProposedActionEN2Qt10Dro pActionE [LSB]
_ZN27QGraphicsSceneDragDropEve nt18setPossibleActionsE6QFlagsIN2 Qt10DropActionEE [LSB]	_ZN27QGraphicsSceneDragDropEve nt20acceptProposedActionEv [QtXml]
_ZN27QGraphicsSceneDragDropEve nt6setPosERK7QPointF [LSB]	_ZN27QGraphicsSceneDragDropEve nt9setSourceEP7QWidget [LSB]
_ZN27QGraphicsSceneDragDropEve ntC1EN6QEvent4TypeE [LSB]	_ZN27QGraphicsSceneDragDropEve ntC2EN6QEvent4TypeE [LSB]
_ZN27QGraphicsSceneDragDropEve ntD0Ev [QtXml]	_ZN27QGraphicsSceneDragDropEve ntD1Ev [QtXml]
_ZN27QGraphicsSceneDragDropEve ntD2Ev [QtXml]	_ZN30QGraphicsSceneContextMenu Event11setScenePosERK7QPointF [LSB]

_ZN30QGraphicsSceneContextMenuEvent12setModifiersE6QFlagsIN2Qt16KeyboardModifierEE [LSB]	_ZN30QGraphicsSceneContextMenuEvent12setScreenPosERK6QPoint [LSB]
_ZN30QGraphicsSceneContextMenuEvent6setPosERK7QPointF [LSB]	_ZN30QGraphicsSceneContextMenuEvent9setReasonENS_6ReasonE [LSB]
_ZN30QGraphicsSceneContextMenuEventC1EN6QEvent4TypeE [LSB]	_ZN30QGraphicsSceneContextMenuEventC2EN6QEvent4TypeE [LSB]
_ZN30QGraphicsSceneContextMenuEventD0Ev [QtXml]	_ZN30QGraphicsSceneContextMenuEventD1Ev [QtXml]
_ZN30QGraphicsSceneContextMenuEventD2Ev [QtXml]	_ZN4QPen10isDetachedEv [QtGui]
_ZN4QPen11setCapStyleEN2Qt11PenCapStyleE [QtGui]	_ZN4QPen12setJoinStyleEN2Qt12PenJoinStyleE [QtGui]
_ZN4QPen13setMiterLimitEd [QtGui]	_ZN4QPen14setDashPatternERK7QVectorIdE [QtGui]
_ZN4QPen8setBrushERK6QBrush [QtGui]	_ZN4QPen8setColorERK6QColor [QtGui]
_ZN4QPen8setStyleEN2Qt8PenStyleE [QtGui]	_ZN4QPen8setWidthEi [QtGui]
_ZN4QPen9setWidthFE [QtGui]	_ZN4QPenC1EN2Qt8PenStyleE [QtGui]
_ZN4QPenC1ERK6QBrushdN2Qt8PenStyleENS3_11PenCapStyleENS3_12PenJoinStyleE [QtGui]	_ZN4QPenC1ERK6QColor [QtGui]
ZN4QPenC1ERKS [QtGui]	_ZN4QPenC1Ev [QtGui]
_ZN4QPenC2EN2Qt8PenStyleE [QtGui]	_ZN4QPenC2ERK6QBrushdN2Qt8PenStyleENS3_11PenCapStyleENS3_12PenJoinStyleE [QtGui]
_ZN4QPenC2ERK6QColor [QtGui]	_ZN4QPenC2ERKS_ [QtGui]
_ZN4QPenC2Ev [QtGui]	_ZN4QPenD1Ev [QtGui]
_ZN4QPenD2Ev [QtGui]	_ZN4QPenASERKS_ [QtGui]
_ZN5QIcon10pixmapSizeENS_4SizeE [QtGui]	_ZN5QIcon13setPixmapSizeENS_4SizeERK5QSize [QtGui]
_ZN5QIcon7addFileERK7QStringRK5QSizeNS_4ModeENS_5StateE [QtGui]	_ZN5QIcon9addPixmapERK7QPixmapNS_4ModeENS_5StateE [QtGui]
_ZN5QIconC1EP11QIconEngine [QtGui]	_ZN5QIconC1ERK7QPixmap [QtGui]
_ZN5QIconC1ERK7QString [QtGui]	_ZN5QIconC1ERKS_ [QtGui]
_ZN5QIconC1Ev [QtGui]	_ZN5QIconC2EP11QIconEngine [QtGui]

_ZN5QIconC2ERK7QPixmap [QtGui]	_ZN5QIconC2ERK7QString [QtGui]
ZN5QIconC2ERKS [QtGui]	_ZN5QIconC2Ev [QtGui]
_ZN5QIconD1Ev [QtGui]	_ZN5QIconD2Ev [QtGui]
ZN5QIconaSERKS [QtGui]	_ZN6QBrush10setTextureERK7QPixmap [QtGui]
_ZN6QBrush15setTextureImageERK6QImage [QtXml]	_ZN6QBrush8setColorERK6QColor [QtGui]
_ZN6QBrush8setStyleEN2Qt10BrushStyleE [QtGui]	_ZN6QBrush9setMatrixERK7QMatrix [QtXml]
_ZN6QBrushC1EN2Qt10BrushStyleE [QtGui]	_ZN6QBrushC1EN2Qt11GlobalColorENS0_10BrushStyleE [QtGui]
_ZN6QBrushC1EN2Qt11GlobalColorERK7QPixmap [QtGui]	_ZN6QBrushC1ERK6QColorN2Qt10BrushStyleE [QtGui]
_ZN6QBrushC1ERK6QColorRK7QPixmap [QtGui]	_ZN6QBrushC1ERK6QImage [QtXml]
_ZN6QBrushC1ERK7QPixmap [QtGui]	_ZN6QBrushC1ERK9QGradient [QtGui]
ZN6QBrushC1ERKS [QtGui]	_ZN6QBrushC1Ev [QtGui]
_ZN6QBrushC2EN2Qt10BrushStyleE [QtGui]	_ZN6QBrushC2EN2Qt11GlobalColorENS0_10BrushStyleE [QtGui]
_ZN6QBrushC2EN2Qt11GlobalColorERK7QPixmap [QtGui]	_ZN6QBrushC2ERK6QColorN2Qt10BrushStyleE [QtGui]
_ZN6QBrushC2ERK6QColorRK7QPixmap [QtGui]	_ZN6QBrushC2ERK6QImage [QtXml]
_ZN6QBrushC2ERK7QPixmap [QtGui]	_ZN6QBrushC2ERK9QGradient [QtGui]
ZN6QBrushC2ERKS [QtGui]	_ZN6QBrushC2Ev [QtGui]
_ZN6QBrushD1Ev [QtGui]	_ZN6QBrushD2Ev [QtGui]
ZN6QBrushaSERKS [QtGui]	_ZN7QMatrix5resetEv [QtGui]
_ZN7QMatrix5scaleEdd [QtGui]	_ZN7QMatrix5shearEdd [QtGui]
_ZN7QMatrix6rotateEd [QtGui]	_ZN7QMatrix9setMatrixEddddd [QtGui]
_ZN7QMatrix9translateEdd [QtGui]	_ZN7QMatrixC1ERKS_ [QtGui]
_ZN7QMatrixC1Eddddd [QtGui]	_ZN7QMatrixC1Ev [QtGui]
ZN7QMatrixC2ERKS [QtGui]	_ZN7QMatrixC2Eddddd [QtGui]
_ZN7QMatrixC2Ev [QtGui]	_ZN7QMatrixaSERKS_ [QtGui]
ZN7QMatrixmLERKS [QtGui]	_ZN7QRegion8setRectsEPK5QRecti [QtGui]

_ZN7QRegion9translateEii [QtGui]	_ZN7QRegionC1ERK5QRectNS_10RegionTypeE [QtGui]
_ZN7QRegionC1ERK7QBitmap [QtGui]	_ZN7QRegionC1ERK8QPolygonN2Qt8FillRuleE [QtGui]
_ZN7QRegionC1ERK8QPolygonb [QtGui]	_ZN7QRegionC1ERKS_ [QtGui]
_ZN7QRegionC1EiiiiNS_10RegionTypeE [QtGui]	_ZN7QRegionC1Ev [QtGui]
_ZN7QRegionC2ERK5QRectNS_10RegionTypeE [QtGui]	_ZN7QRegionC2ERK7QBitmap [QtGui]
_ZN7QRegionC2ERK8QPolygonN2Qt8FillRuleE [QtGui]	_ZN7QRegionC2ERK8QPolygonb [QtGui]
ZN7QRegionC2ERKS [QtGui]	_ZN7QRegionC2EiiiiNS_10RegionTypeE [QtGui]
_ZN7QRegionC2Ev [QtGui]	_ZN7QRegionD1Ev [QtGui]
_ZN7QRegionD2Ev [QtGui]	_ZN7QRegionaNERKS_ [QtGui]
ZN7QRegionaSERKS [QtGui]	_ZN7QRegioneOERKS_ [QtGui]
ZN7QRegionmLERKS [QtGui]	_ZN7QRegionoRERKS_ [QtGui]
ZN7QRegionpLERKS [QtGui]	_ZN8QPainter10drawPixmapERK6QRectFRK7QPixmapS2_ [QtGui]
_ZN8QPainter10drawPointsEPK6QPointi [QtGui]	_ZN8QPainter10drawPointsEPK7QPointFi [QtGui]
_ZN8QPainter10redirectedEPK12QPaintDeviceP6QPoint [QtGui]	_ZN8QPainter10setOpacityEd [QtGui]
_ZN8QPainter10strokePathERK12QPainterPathRK4QPen [QtGui]	_ZN8QPainter11drawEllipseERK5QRect [QtGui]
_ZN8QPainter11drawEllipseERK6QRectF [QtGui]	_ZN8QPainter11drawPictureERK7QPointFRK8QPicture [QtGui]
_ZN8QPainter11drawPolygonEPK6QPointiN2Qt8FillRuleE [QtGui]	_ZN8QPainter11drawPolygonEPK7QPointFiN2Qt8FillRuleE [QtGui]
_ZN8QPainter11resetMatrixEv [QtGui]	_ZN8QPainter11setClipPathERK12QPainterPathN2Qt13ClipOperationE [QtGui]
_ZN8QPainter11setClipRectERK6QRectFN2Qt13ClipOperationE [QtGui]	_ZN8QPainter11setClippingEb [QtGui]
_ZN8QPainter11setViewportERK5QRect [QtGui]	_ZN8QPainter12boundingRectERK5QRectiRK7QString [QtGui]
_ZN8QPainter12boundingRectERK6QRectFRK7QStringRK11QTextOption [QtGui]	_ZN8QPainter12boundingRectERK6QRectFiRK7QString [QtGui]
_ZN8QPainter12drawPolylineEPK6QPointi [QtGui]	_ZN8QPainter12drawPolylineEPK7QPointFi [QtGui]

_ZN8QPainter13drawRoundRectERK6QRectFii [QtGui]	_ZN8QPainter13setBackgroundERK6QBrush [QtGui]
_ZN8QPainter13setClipRegionERK7QRegionN2Qt13ClipOperationE [QtGui]	_ZN8QPainter13setRedirectedEPK12QPaintDevicePS0_RK6QPoint [QtGui]
_ZN8QPainter13setRenderHintENS_10RenderHintEb [QtGui]	_ZN8QPainter14setBrushOriginERK7QPointF [QtGui]
_ZN8QPainter14setRenderHintsE6QFlagsINS_10RenderHintEEb [QtXml]	_ZN8QPainter14setWorldMatrixERK7QMatrixb [QtXml]
_ZN8QPainter15drawCubicBezierERK8QPolygoni [QtGui]	_ZN8QPainter15drawTiledPixmapERK6QRectFRK7QPixmapRK7QPointF [QtGui]
_ZN8QPainter16drawLineSegmentsERK8QPolygonii [QtGui]	_ZN8QPainter16setMatrixEnabledEb [QtGui]
_ZN8QPainter17drawConvexPolygonEPK6QPointi [QtGui]	_ZN8QPainter17drawConvexPolygonEPK7QPointFi [QtGui]
_ZN8QPainter17restoreRedirectedEPK12QPaintDevice [QtGui]	_ZN8QPainter17setBackgroundModeEN2Qt6BGModeE [QtGui]
_ZN8QPainter18setCompositionModeENS_15CompositionModeE [QtGui]	_ZN8QPainter18setLayoutDirectionEN2Qt15LayoutDirectionE [QtGui]
_ZN8QPainter21setWorldMatrixEnabledEb [QtXml]	_ZN8QPainter23setViewTransformEnabledEb [QtGui]
_ZN8QPainter3endEv [QtGui]	_ZN8QPainter4saveEv [QtGui]
_ZN8QPainter5beginEP12QPaintDevice [QtGui]	_ZN8QPainter5scaleEdd [QtGui]
_ZN8QPainter5shearEdd [QtGui]	_ZN8QPainter6rotateEd [QtGui]
_ZN8QPainter6setPenEN2Qt8PenStyleE [QtGui]	_ZN8QPainter6setPenERK4QPen [QtGui]
_ZN8QPainter6setPenERK6QColor [QtGui]	_ZN8QPainter7drawArcERK6QRectFii [QtGui]
_ZN8QPainter7drawPieERK6QRectFii [QtGui]	_ZN8QPainter7restoreEv [QtGui]
_ZN8QPainter7setFontERK5QFont [QtGui]	_ZN8QPainter8drawPathERK12QPainterPath [QtGui]
ZN8QPainter8drawTextERK5QRectiRK7QStringPS0 [QtGui]	_ZN8QPainter8drawTextERK6QRectFRK7QStringRK11QTextOption [QtGui]
ZN8QPainter8drawTextERK6QRectFiRK7QStringPS0 [QtGui]	_ZN8QPainter8drawTextERK7QPointFRK7QString [QtGui]
_ZN8QPainter8fillPathERK12QPainterPathRK6QBrush [QtGui]	_ZN8QPainter8fillRectERK5QRectRK6QBrush [QtGui]

_ZN8QPainter8fillRectERK6QRectFRK6QBrush [QtGui]	_ZN8QPainter8initFromEPK7QWidg et [QtGui]
_ZN8QPainter8setBrushEN2Qt10BrushStyleE [QtGui]	_ZN8QPainter8setBrushERK6QBrush [QtGui]
_ZN8QPainter9drawChordERK6QRectFii [QtGui]	_ZN8QPainter9drawImageERK6QRectFRK6QImageS2_6QFlagsIN2Qt19ImageConversionFlagEE [QtGui]
_ZN8QPainter9drawLinesEPK5QLinei [QtGui]	_ZN8QPainter9drawLinesEPK6QLineFi [QtGui]
_ZN8QPainter9drawLinesEPK6QPointi [QtGui]	_ZN8QPainter9drawLinesEPK7QPointFi [QtGui]
_ZN8QPainter9drawRectsEPK5QRecti [QtGui]	_ZN8QPainter9drawRectsEPK6QRectFi [QtGui]
_ZN8QPainter9eraseRectERK6QRectF [QtGui]	_ZN8QPainter9setMatrixERK7QMatrixb [QtGui]
_ZN8QPainter9setWindowERK5QRect [QtGui]	_ZN8QPainter9translateERK7QPointF [QtGui]
_ZN8QPainterC1EP12QPaintDevice [QtGui]	_ZN8QPainterC1Ev [QtGui]
_ZN8QPainterC2EP12QPaintDevice [QtGui]	_ZN8QPainterC2Ev [QtGui]
_ZN8QPainterD1Ev [QtGui]	_ZN8QPainterD2Ev [QtGui]
_ZN8QPolygon9putPointsEiiPKi [QtGui]	_ZN8QPolygon9putPointsEiiRKSi [QtGui]
_ZN8QPolygon9putPointsEiiiiz [QtGui]	_ZN8QPolygon9setPointsEiiPKi [QtGui]
_ZN8QPolygon9setPointsEiiiiz [QtGui]	_ZN8QPolygon9translateEii [QtGui]
_ZN8QPolygonC1ERK5QRectb [QtGui]	_ZN8QPolygonC1EiPKi [QtGui]
_ZN8QPolygonC2ERK5QRectb [QtGui]	_ZN8QPolygonC2EiPKi [QtGui]
_ZN9QGradient10setColorAtEdRK6QColor [QtGui]	_ZN9QGradient17setCoordinateModeENS_14CoordinateModeE [QtXml]
_ZN9QGradient8setStopsERK7QVectorI5QPairId6QColorEE [QtGui]	_ZN9QGradientC1Ev [QtGui]
_ZN9QGradientC2Ev [QtGui]	_ZN9QGradienteqERKS_ [QtGui]
_ZN9QPolygonF9translateERK7QPointF [QtGui]	_ZN9QPolygonFC1ERK6QRectF [QtGui]
_ZN9QPolygonFC1ERK8QPolygon [QtGui]	_ZN9QPolygonFC2ERK6QRectF [QtGui]

_ZN9QPolygonFC2ERK8QPolygon [QtGui]	_ZNK12QPaintEngine10systemClipEv [LSB]
_ZNK12QPaintEngine10systemRectEv [LSB]	_ZNK12QPaintEngine11paintDeviceEv [QtGui]
_ZNK12QPaintEngine16coordinateOffsetEv [LSB]	_ZNK12QPaintEngine7painterEv [QtGui]
_ZNK12QPainterPath10intersectsERK6QRectF [QtGui]	_ZNK12QPainterPath10toReversedEv [QtGui]
_ZNK12QPainterPath12boundingRectEv [QtGui]	_ZNK12QPainterPath13toFillPolygonERK7QMatrix [QtGui]
_ZNK12QPainterPath14toFillPolygonsERK7QMatrix [QtGui]	_ZNK12QPainterPath15currentPositionEv [QtGui]
_ZNK12QPainterPath16controlPointRectEv [QtGui]	_ZNK12QPainterPath17toSubpathPolygonsERK7QMatrix [QtGui]
_ZNK12QPainterPath8containsERK6QRectF [QtGui]	_ZNK12QPainterPath8containsERK7QPointF [QtGui]
_ZNK12QPainterPath8fillRuleEv [QtGui]	_ZNK12QPainterPatheqERKS_ [QtGui]
ZNK12QPainterPathneERKS [QtGui]	_ZNK13QGraphicsItem10isObscuredEv [QtXml]
_ZNK13QGraphicsItem10isSelectedEv [QtXml]	_ZNK13QGraphicsItem10mapToSceneERK12QPainterPath [QtXml]
_ZNK13QGraphicsItem10mapToSceneERK6QRectF [QtXml]	_ZNK13QGraphicsItem10mapToSceneERK7QPointF [QtXml]
_ZNK13QGraphicsItem10mapToSceneERK9QPolygonF [QtXml]	_ZNK13QGraphicsItem10opaqueAreaEv [QtXml]
_ZNK13QGraphicsItem10parentItemEv [QtXml]	_ZNK13QGraphicsItem11acceptDropEv [QtXml]
_ZNK13QGraphicsItem11mapFromItemEPKS_RK12QPainterPath [QtXml]	_ZNK13QGraphicsItem11mapFromItemEPKS_RK6QRectF [QtXml]
_ZNK13QGraphicsItem11mapFromItemEPKS_RK7QPointF [QtXml]	_ZNK13QGraphicsItem11mapFromItemEPKS_RK9QPolygonF [QtXml]
_ZNK13QGraphicsItem11mapToParentERK12QPainterPath [QtXml]	_ZNK13QGraphicsItem11mapToParentERK6QRectF [QtXml]
_ZNK13QGraphicsItem11mapToParentERK7QPointF [QtXml]	_ZNK13QGraphicsItem11mapToParentERK9QPolygonF [QtXml]
_ZNK13QGraphicsItem11sceneMatrixEv [QtXml]	_ZNK13QGraphicsItem12isAncestorOfEPKS_ [QtXml]
ZNK13QGraphicsItem12isObscuredByEPKS [QtXml]	_ZNK13QGraphicsItem12mapFromSceneERK12QPainterPath [QtXml]
_ZNK13QGraphicsItem12mapFromSceneERK6QRectF [QtXml]	_ZNK13QGraphicsItem12mapFromSceneERK7QPointF [QtXml]

_Znk13QGraphicsItem12mapFromSceneERK9QPolygonF [QtXml]	_Znk13QGraphicsItem12topLevelItemEv [QtXml]
_Znk13QGraphicsItem13mapFromParentERK12QPainterPath [QtXml]	_Znk13QGraphicsItem13mapFromParentERK6QRectF [QtXml]
_Znk13QGraphicsItem13mapFromParentERK7QPointF [QtXml]	_Znk13QGraphicsItem13mapFromParentERK9QPolygonF [QtXml]
_Znk13QGraphicsItem14collidingItemsEN2Qt17ItemSelectionMode [QtXml]	_Znk13QGraphicsItem16collidesWithItemEPKS_N2Qt17ItemSelectionMode [QtXml]
_Znk13QGraphicsItem16collidesWithPathERK12QPainterPathN2Qt17ItemSelectionMode [QtXml]	_Znk13QGraphicsItem16inputMethodQueryEN2Qt16InputMethodQueryE [QtXml]
_Znk13QGraphicsItem17sceneBoundingRectEv [QtXml]	_Znk13QGraphicsItem17supportsExtensionENS_9ExtensionE [LSB]
_Znk13QGraphicsItem18acceptsHoverEventsEv [QtXml]	_Znk13QGraphicsItem18handlesChildEventsEv [QtXml]
_Znk13QGraphicsItem20acceptedMouseButtonsEv [QtXml]	_Znk13QGraphicsItem20childrenBoundingRectEv [QtXml]
_Znk13QGraphicsItem3posEv [QtXml]	_Znk13QGraphicsItem4dataEi [QtXml]
_Znk13QGraphicsItem4typeEv [QtXml]	_Znk13QGraphicsItem5flagsEv [QtXml]
_Znk13QGraphicsItem5groupEv [QtXml]	_Znk13QGraphicsItem5sceneEv [QtXml]
_Znk13QGraphicsItem5shapeEv [QtXml]	_Znk13QGraphicsItem6cursorEv [QtXml]
_Znk13QGraphicsItem6matrixEv [QtXml]	_Znk13QGraphicsItem6zValueEv [QtXml]
_Znk13QGraphicsItem7toolTipEv [QtXml]	_Znk13QGraphicsItem8childrenEv [QtXml]
_Znk13QGraphicsItem8containsERK7QPointF [QtXml]	_Znk13QGraphicsItem8hasFocusEv [QtXml]
_Znk13QGraphicsItem8scenePosEv [QtXml]	_Znk13QGraphicsItem9extensionERK8QVariant [LSB]
_Znk13QGraphicsItem9hasCursorEv [QtXml]	_Znk13QGraphicsItem9isEnabledEv [QtXml]
_Znk13QGraphicsItem9isVisibleEv [QtXml]	_Znk13QGraphicsItem9mapToItemEPKS_RK12QPainterPath [QtXml]
_Znk13QGraphicsItem9mapToItemEPKS_RK6QRectF [QtXml]	_Znk13QGraphicsItem9mapToItemEPKS_RK7QPointF [QtXml]
_Znk13QGraphicsItem9mapToItemEPKS_RK9QPolygonF [QtXml]	_Znk13QGraphicsView10mapToSceneERK12QPainterPath [QtXml]

_Znk13QGraphicsView10mapToSceneERK5QRect [QtXml]	_Znk13QGraphicsView10mapToSceneERK6QPoint [QtXml]
_Znk13QGraphicsView10mapToSceneERK8QPolygon [QtXml]	_Znk13QGraphicsView10metaObjectEv [QtXml]
_Znk13QGraphicsView11renderHintsEv [QtXml]	_Znk13QGraphicsView12mapFromSceneERK12QPainterPath [QtXml]
_Znk13QGraphicsView12mapFromSceneERK6QRectF [QtXml]	_Znk13QGraphicsView12mapFromSceneERK7QPointF [QtXml]
_Znk13QGraphicsView12mapFromSceneERK9QPolygonF [QtXml]	_Znk13QGraphicsView12resizeAnchorEv [QtXml]
_Znk13QGraphicsView13isInteractiveEv [QtXml]	_Znk13QGraphicsView15backgroundBrushEv [QtXml]
_Znk13QGraphicsView15foregroundBrushEv [QtXml]	_Znk13QGraphicsView16inputMethodQueryEN2Qt16InputMethodQueryE [QtXml]
_Znk13QGraphicsView20transformationAnchorEv [QtXml]	_Znk13QGraphicsView5itemsERK12QPainterPathN2Qt17ItemSelectionModeE [QtXml]
_Znk13QGraphicsView5itemsERK5QRectN2Qt17ItemSelectionModeE [QtXml]	_Znk13QGraphicsView5itemsERK6QPoint [QtXml]
_Znk13QGraphicsView5itemsERK8QPolygonN2Qt17ItemSelectionModeE [QtXml]	_Znk13QGraphicsView5itemsEv [QtXml]
_Znk13QGraphicsView5sceneEv [QtXml]	_Znk13QGraphicsView6itemAtERK6QPoint [QtXml]
_Znk13QGraphicsView6matrixEv [QtXml]	_Znk13QGraphicsView8dragModeEv [QtXml]
_Znk13QGraphicsView8sizeHintEv [QtXml]	_Znk13QGraphicsView9alignmentEv [QtXml]
_Znk13QGraphicsView9cacheModeEv [QtXml]	_Znk13QGraphicsView9sceneRectEv [QtXml]
_Znk14QGraphicsScene10metaObjectEv [QtXml]	_Znk14QGraphicsScene13selectedItemsEv [QtXml]
_Znk14QGraphicsScene14collidingItemsEPK13QGraphicsItemN2Qt17ItemSelectionModeE [QtXml]	_Znk14QGraphicsScene15backgroundBrushEv [QtXml]
_Znk14QGraphicsScene15foregroundBrushEv [QtXml]	_Znk14QGraphicsScene15itemIndexMethodEv [QtXml]
_Znk14QGraphicsScene16inputMethodQueryEN2Qt16InputMethodQueryE [QtXml]	_Znk14QGraphicsScene16mouseGrabberItemEv [QtXml]

_ZNK14QGraphicsScene17itemsBoundingRectEv [QtXml]	_ZNK14QGraphicsScene5itemsERK12QPainterPathN2Qt17ItemSelectionModeE [QtXml]
_ZNK14QGraphicsScene5itemsERK6QRectFN2Qt17ItemSelectionModeE [QtXml]	_ZNK14QGraphicsScene5itemsERK7QPointF [QtXml]
_ZNK14QGraphicsScene5itemsERK9QPolygonFN2Qt17ItemSelectionModeE [QtXml]	_ZNK14QGraphicsScene5itemsEv [QtXml]
_ZNK14QGraphicsScene5viewsEv [QtXml]	_ZNK14QGraphicsScene6itemAtERK7QPointF [QtXml]
_ZNK14QGraphicsScene8hasFocusEv [QtXml]	_ZNK14QGraphicsScene9focusItemEv [QtXml]
_ZNK14QGraphicsScene9sceneRectEv [QtXml]	_ZNK15QLinearGradient5startEv [QtGui]
_ZNK15QLinearGradient9finalStopEv [QtGui]	_ZNK15QRadialGradient10focalPointEv [QtGui]
_ZNK15QRadialGradient6centerEv [QtGui]	_ZNK15QRadialGradient6radiusEv [QtGui]
_ZNK16QConicalGradient5angleEv [QtGui]	_ZNK16QConicalGradient6centerEv [QtGui]
_ZNK17QGraphicsLineItem10opaqueAreaEv [QtXml]	_ZNK17QGraphicsLineItem12boundingRectEv [QtXml]
_ZNK17QGraphicsLineItem12isObscuredByEPK13QGraphicsItem [QtXml]	_ZNK17QGraphicsLineItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]
_ZNK17QGraphicsLineItem3penEv [QtXml]	_ZNK17QGraphicsLineItem4lineEv [QtXml]
_ZNK17QGraphicsLineItem4typeEv [QtXml]	_ZNK17QGraphicsLineItem5shapeEv [QtXml]
_ZNK17QGraphicsLineItem8containsERK7QPointF [QtXml]	_ZNK17QGraphicsLineItem9extensionERK8QVariant [LSB]
_ZNK17QGraphicsPathItem10opaqueAreaEv [QtXml]	_ZNK17QGraphicsPathItem12boundingRectEv [QtXml]
_ZNK17QGraphicsPathItem12isObscuredByEPK13QGraphicsItem [QtXml]	_ZNK17QGraphicsPathItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]
_ZNK17QGraphicsPathItem4pathEv [QtXml]	_ZNK17QGraphicsPathItem4typeEv [QtXml]
_ZNK17QGraphicsPathItem5shapeEv [QtXml]	_ZNK17QGraphicsPathItem8containsERK7QPointF [QtXml]
_ZNK17QGraphicsPathItem9extensionERK8QVariant [LSB]	_ZNK17QGraphicsRectItem10opaqueAreaEv [QtXml]

_ZNK17QGraphicsRectItem12boundingRectEv [QtXml]	_ZNK17QGraphicsRectItem12isObscuredByEPK13QGraphicsItem [QtXml]
_ZNK17QGraphicsRectItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]	_ZNK17QGraphicsRectItem4rectEv [QtXml]
_ZNK17QGraphicsRectItem4typeEv [QtXml]	_ZNK17QGraphicsRectItem5shapeEv [QtXml]
_ZNK17QGraphicsRectItem8containsERK7QPointF [QtXml]	_ZNK17QGraphicsRectItem9extensionERK8QVariant [LSB]
_ZNK17QGraphicsTextItem10metaObjectEv [QtXml]	_ZNK17QGraphicsTextItem10opaqueAreaEv [QtXml]
_ZNK17QGraphicsTextItem10textCursorEv [QtXml]	_ZNK17QGraphicsTextItem11toPlainTextEv [QtXml]
_ZNK17QGraphicsTextItem12boundingRectEv [QtXml]	_ZNK17QGraphicsTextItem12isObscuredByEPK13QGraphicsItem [QtXml]
_ZNK17QGraphicsTextItem16defaultTextColorEv [QtXml]	_ZNK17QGraphicsTextItem16inputMethodQueryEN2Qt16InputMethodQueryE [QtXml]
_ZNK17QGraphicsTextItem17openExternalLinksEv [QtXml]	_ZNK17QGraphicsTextItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]
_ZNK17QGraphicsTextItem20textInteractionFlagsEv [QtXml]	_ZNK17QGraphicsTextItem4fontEv [QtXml]
_ZNK17QGraphicsTextItem4typeEv [QtXml]	_ZNK17QGraphicsTextItem5shapeEv [QtXml]
_ZNK17QGraphicsTextItem6toHtmlEv [QtXml]	_ZNK17QGraphicsTextItem8containsERK7QPointF [QtXml]
_ZNK17QGraphicsTextItem8documentEv [QtXml]	_ZNK17QGraphicsTextItem9extensionERK8QVariant [LSB]
_ZNK17QGraphicsTextItem9textWidthEv [QtXml]	_ZNK17QPaintEngineState10clipRegionEv [QtGui]
_ZNK17QPaintEngineState11brushOriginEv [QtGui]	_ZNK17QPaintEngineState11renderHintsEv [QtGui]
_ZNK17QPaintEngineState13clipOperationEv [QtGui]	_ZNK17QPaintEngineState13isClipEnabledEv [QtGui]
_ZNK17QPaintEngineState14backgroundModeEv [QtGui]	_ZNK17QPaintEngineState15backgroundBrushEv [QtGui]
_ZNK17QPaintEngineState15compositionModeEv [QtGui]	_ZNK17QPaintEngineState3penEv [QtGui]
_ZNK17QPaintEngineState4fontEv [QtGui]	_ZNK17QPaintEngineState5brushEv [QtGui]

_Znk17QPaintEngineState6matrixEv [QtGui]	_Znk17QPaintEngineState7opacityEv [QtXml]
_Znk17QPaintEngineState7painterEv [QtGui]	_Znk17QPaintEngineState8clipPathEv [QtGui]
_Znk18QGraphicsItemGroup10opaqueAreaEv [QtXml]	_Znk18QGraphicsItemGroup12boundingRectEv [QtXml]
_Znk18QGraphicsItemGroup12isObscuredByEPK13QGraphicsItem [QtXml]	_Znk18QGraphicsItemGroup4typeEv [QtXml]
_Znk19QGraphicsPixmapItem10opaqueAreaEv [QtXml]	_Znk19QGraphicsPixmapItem12boundingRectEv [QtXml]
_Znk19QGraphicsPixmapItem12isObscuredByEPK13QGraphicsItem [QtXml]	_Znk19QGraphicsPixmapItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]
_Znk19QGraphicsPixmapItem18transformationModeEv [QtXml]	_Znk19QGraphicsPixmapItem4typeEv [QtXml]
_Znk19QGraphicsPixmapItem5shapeEv [QtXml]	_Znk19QGraphicsPixmapItem6offsetEv [QtXml]
_Znk19QGraphicsPixmapItem6pixmapEv [QtXml]	_Znk19QGraphicsPixmapItem8containsERK7QPointF [QtXml]
_Znk19QGraphicsPixmapItem9extensionERK8QVariant [LSB]	_Znk19QGraphicsPixmapItem9shapeModeEv [QtXml]
_Znk19QGraphicsSceneEvent6widgetEv [QtXml]	_Znk19QPainterPathStroker10miterLimitEv [QtGui]
_Znk19QPainterPathStroker11dashPatternEv [QtGui]	_Znk19QPainterPathStroker12createStrokeERK12QPainterPath [QtGui]
_Znk19QPainterPathStroker14curveThresholdEv [QtGui]	_Znk19QPainterPathStroker5widthEv [QtGui]
_Znk19QPainterPathStroker8capStyleEv [QtGui]	_Znk19QPainterPathStroker9joinStyleEv [QtGui]
_Znk20QGraphicsEllipseItem10opaqueAreaEv [QtXml]	_Znk20QGraphicsEllipseItem10startAngleEv [QtXml]
_Znk20QGraphicsEllipseItem12boundingRectEv [QtXml]	_Znk20QGraphicsEllipseItem12isObscuredByEPK13QGraphicsItem [QtXml]
_Znk20QGraphicsEllipseItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]	_Znk20QGraphicsEllipseItem4rectEv [QtXml]
_Znk20QGraphicsEllipseItem4typeEv [QtXml]	_Znk20QGraphicsEllipseItem5shapeEv [QtXml]
_Znk20QGraphicsEllipseItem8containsERK7QPointF [QtXml]	_Znk20QGraphicsEllipseItem9extensionERK8QVariant [LSB]

_ZNK20QGraphicsEllipseItem9spanAngleEv [QtXml]	_ZNK20QGraphicsPolygonItem10opaqueAreaEv [QtXml]
_ZNK20QGraphicsPolygonItem12boundingRectEv [QtXml]	_ZNK20QGraphicsPolygonItem12isObscuredByEPK13QGraphicsItem [QtXml]
_ZNK20QGraphicsPolygonItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]	_ZNK20QGraphicsPolygonItem4typeEv [QtXml]
_ZNK20QGraphicsPolygonItem5shapeEv [QtXml]	_ZNK20QGraphicsPolygonItem7polygonEv [QtXml]
_ZNK20QGraphicsPolygonItem8containsERK7QPointF [QtXml]	_ZNK20QGraphicsPolygonItem8fillRuleEv [QtXml]
_ZNK20QGraphicsPolygonItem9extensionERK8QVariant [LSB]	_ZNK22QGraphicsItemAnimation10metaObjectEv [QtXml]
_ZNK22QGraphicsItemAnimation10rotationAtEd [QtXml]	_ZNK22QGraphicsItemAnimation14xTranslationAtEd [QtXml]
_ZNK22QGraphicsItemAnimation14yTranslationAtEd [QtXml]	_ZNK22QGraphicsItemAnimation15verticalScaleAtEd [QtXml]
_ZNK22QGraphicsItemAnimation15verticalShearAtEd [QtXml]	_ZNK22QGraphicsItemAnimation17horizontalScaleAtEd [QtXml]
_ZNK22QGraphicsItemAnimation17horizontalShearAtEd [QtXml]	_ZNK22QGraphicsItemAnimation4itemEv [QtXml]
_ZNK22QGraphicsItemAnimation5posAtEd [QtXml]	_ZNK22QGraphicsItemAnimation8matrixAtEd [QtXml]
_ZNK22QGraphicsItemAnimation8timeLineEv [QtXml]	_ZNK23QGraphicsSceneHelpEvent8scenePosEv [QtXml]
_ZNK23QGraphicsSceneHelpEvent9screenPosEv [QtXml]	_ZNK23QGraphicsSimpleTextItem10opaqueAreaEv [QtXml]
_ZNK23QGraphicsSimpleTextItem12boundingRectEv [QtXml]	_ZNK23QGraphicsSimpleTextItem12isObscuredByEPK13QGraphicsItem [QtXml]
_ZNK23QGraphicsSimpleTextItem17supportsExtensionEN13QGraphicsItem9ExtensionE [LSB]	_ZNK23QGraphicsSimpleTextItem4fontEv [QtXml]
_ZNK23QGraphicsSimpleTextItem4textEv [QtXml]	_ZNK23QGraphicsSimpleTextItem4typeEv [QtXml]
_ZNK23QGraphicsSimpleTextItem5shapeEv [QtXml]	_ZNK23QGraphicsSimpleTextItem8containsERK7QPointF [QtXml]
_ZNK23QGraphicsSimpleTextItem9extensionERK8QVariant [LSB]	_ZNK24QGraphicsSceneHoverEvent3posEv [QtXml]
_ZNK24QGraphicsSceneHoverEvent8scenePosEv [QtXml]	_ZNK24QGraphicsSceneHoverEvent9screenPosEv [QtXml]

_ZNK24QGraphicsSceneMouseEvent 12lastScenePosEv [QtXml]	_ZNK24QGraphicsSceneMouseEvent 13buttonDownPosEN2Qt11MouseBu ttonE [QtXml]
_ZNK24QGraphicsSceneMouseEvent 13lastScreenPosEv [QtXml]	_ZNK24QGraphicsSceneMouseEvent 18buttonDownScenePosEN2Qt11Mo useButtonE [QtXml]
_ZNK24QGraphicsSceneMouseEvent 19buttonDownScreenPosEN2Qt11M ouseButtonE [QtXml]	_ZNK24QGraphicsSceneMouseEvent 3posEv [QtXml]
_ZNK24QGraphicsSceneMouseEvent 6buttonEv [QtXml]	_ZNK24QGraphicsSceneMouseEvent 7buttonsEv [QtXml]
_ZNK24QGraphicsSceneMouseEvent 7lastPosEv [QtXml]	_ZNK24QGraphicsSceneMouseEvent 8scenePosEv [QtXml]
_ZNK24QGraphicsSceneMouseEvent 9modifiersEv [QtXml]	_ZNK24QGraphicsSceneMouseEvent 9screenPosEv [QtXml]
_ZNK24QGraphicsSceneWheelEvent 3posEv [QtXml]	_ZNK24QGraphicsSceneWheelEvent 5deltaEv [QtXml]
_ZNK24QGraphicsSceneWheelEvent 7buttonsEv [QtXml]	_ZNK24QGraphicsSceneWheelEvent 8scenePosEv [QtXml]
_ZNK24QGraphicsSceneWheelEvent 9modifiersEv [QtXml]	_ZNK24QGraphicsSceneWheelEvent 9screenPosEv [QtXml]
_ZNK27QGraphicsSceneDragDropE vent10dropActionEv [QtXml]	_ZNK27QGraphicsSceneDragDropE vent14proposedActionEv [QtXml]
_ZNK27QGraphicsSceneDragDropE vent15possibleActionsEv [QtXml]	_ZNK27QGraphicsSceneDragDropE vent3posEv [QtXml]
_ZNK27QGraphicsSceneDragDropE vent6sourceEv [QtXml]	_ZNK27QGraphicsSceneDragDropE vent7buttonsEv [QtXml]
_ZNK27QGraphicsSceneDragDropE vent8mimeDataEv [QtXml]	_ZNK27QGraphicsSceneDragDropE vent8scenePosEv [QtXml]
_ZNK27QGraphicsSceneDragDropE vent9modifiersEv [QtXml]	_ZNK27QGraphicsSceneDragDropE vent9screenPosEv [QtXml]
_ZNK30QGraphicsSceneContextMen uEvent3posEv [QtXml]	_ZNK30QGraphicsSceneContextMen uEvent6reasonEv [QtXml]
_ZNK30QGraphicsSceneContextMen uEvent8scenePosEv [QtXml]	_ZNK30QGraphicsSceneContextMen uEvent9modifiersEv [QtXml]
_ZNK30QGraphicsSceneContextMen uEvent9screenPosEv [QtXml]	_ZNK4QPen10miterLimitEv [QtGui]
_ZNK4QPen11dashPatternEv [QtGui]	_ZNK4QPen5brushEv [QtGui]
_ZNK4QPen5colorEv [QtGui]	_ZNK4QPen5styleEv [QtGui]
_ZNK4QPen5widthEv [QtGui]	_ZNK4QPen6widthFEv [QtGui]
_ZNK4QPen7isSolidEv [QtGui]	_ZNK4QPen8capStyleEv [QtGui]

_ZNK4QPen9joinStyleEv [QtGui]	_ZNK4QPencv8QVariantEv [QtGui]
ZNK4QPeneqERKS [QtGui]	_ZNK5QIcon10actualSizeERK5QSizeNS_4ModeENS_5StateE [QtGui]
_ZNK5QIcon10isDetachedEv [QtGui]	_ZNK5QIcon12serialNumberEv [QtGui]
_ZNK5QIcon5paintEP8QPainterRK5QRect6QFlagsIN2Qt13AlignmentFlagEENS_4ModeENS_5StateE [QtGui]	_ZNK5QIcon6isNullEv [QtGui]
_ZNK5QIcon6pixmapENS_4SizeENS_4ModeENS_5StateE [QtGui]	_ZNK5QIcon6pixmapENS_4SizeEbNS_5StateE [QtGui]
_ZNK5QIcon6pixmapERK5QSizeNS_4ModeENS_5StateE [QtGui]	_ZNK5QIcon6pixmapEv [QtGui]
_ZNK5QIconcv8QVariantEv [QtGui]	_ZNK6QBrush12textureImageEv [QtXml]
_ZNK6QBrush6pixmapEv [QtGui]	_ZNK6QBrush7textureEv [QtGui]
_ZNK6QBrush8gradientEv [QtGui]	_ZNK6QBrush8isOpaqueEv [QtGui]
_ZNK6QBrushcv8QVariantEv [QtGui]	_ZNK6QBrushEqERKS_ [QtGui]
_ZNK7QMatrix11mapToRegionERK5QRect [QtGui]	_ZNK7QMatrix12mapToPolygonERK5QRect [QtGui]
_ZNK7QMatrix3mapERK12QPainterPath [QtGui]	_ZNK7QMatrix3mapERK5QLine [QtGui]
_ZNK7QMatrix3mapERK6QPointF [QtGui]	_ZNK7QMatrix3mapERK6QPoint [QtGui]
_ZNK7QMatrix3mapERK7QPointF [QtGui]	_ZNK7QMatrix3mapERK7QRegion [QtGui]
_ZNK7QMatrix3mapERK8QPolygon [QtGui]	_ZNK7QMatrix3mapERK9QPolygonF [QtGui]
ZNK7QMatrix3mapEddPdS0 [QtGui]	_ZNK7QMatrix3mapEiiPiS0_ [QtGui]
_ZNK7QMatrix7mapRectERK5QRect [QtGui]	_ZNK7QMatrix7mapRectERK6QRectF [QtGui]
_ZNK7QMatrix8invertedEPb [QtGui]	_ZNK7QMatrixcv8QVariantEv [QtXml]
ZNK7QMatrixEqERKS [QtGui]	_ZNK7QMatrixmlERKS_ [QtGui]
ZNK7QMatrixneERKS [QtGui]	_ZNK7QRegion10intersectsERK5QRect [QtXml]
ZNK7QRegion10intersectsERKS [QtXml]	_ZNK7QRegion10translatedEii [QtGui]
_ZNK7QRegion12boundingRectEv [QtGui]	_ZNK7QRegion3eorERKS_ [QtGui]

_Znk7QRegion5rectsEv [QtGui]	_Znk7QRegion5uniteERKS_ [QtGui]
_Znk7QRegion7isEmptyEv [QtGui]	_Znk7QRegion8containsERK5QRect [QtGui]
_Znk7QRegion8containsERK6QPoint [QtGui]	_Znk7QRegion8subtractERKS_ [QtGui]
Znk7QRegion9intersectERKS [QtGui]	_Znk7QRegionanERKS_ [QtGui]
_Znk7QRegioncv8QVariantEv [QtGui]	_Znk7QRegioneoERKS_ [QtGui]
Znk7QRegioneqERKS [QtGui]	_Znk7QRegionmiERKS_ [QtGui]
Znk7QRegionorERKS [QtGui]	_Znk7QRegionplERKS_ [QtGui]
_Znk8QPainter10backgroundEv [QtGui]	_Znk8QPainter10clipRegionEv [QtGui]
_Znk8QPainter11brushOriginEv [QtGui]	_Znk8QPainter11fontMetricsEv [QtGui]
_Znk8QPainter11hasClippingEv [QtGui]	_Znk8QPainter11paintEngineEv [QtGui]
_Znk8QPainter11renderHintsEv [QtGui]	_Znk8QPainter11worldMatrixEv [QtGui]
_Znk8QPainter12deviceMatrixEv [QtGui]	_Znk8QPainter12translationXEv [QtGui]
_Znk8QPainter12translationYEv [QtGui]	_Znk8QPainter13matrixEnabledEv [QtGui]
_Znk8QPainter14backgroundModeEv [QtGui]	_Znk8QPainter14combinedMatrixEv [QtGui]
_Znk8QPainter15compositionModeEv [QtGui]	_Znk8QPainter15layoutDirectionEv [QtGui]
_Znk8QPainter18worldMatrixEnabledEv [QtGui]	_Znk8QPainter20viewTransformEnabledEv [QtGui]
Znk8QPainter3mapEiiPiS0 [LSB]	_Znk8QPainter3penEv [QtGui]
_Znk8QPainter4fontEv [QtGui]	_Znk8QPainter5brushEv [QtGui]
_Znk8QPainter5xFormERK5QRect [QtGui]	_Znk8QPainter5xFormERK6QPoint [QtGui]
_Znk8QPainter5xFormERK8QPolygon [QtGui]	_Znk8QPainter5xFormERK8QPolygon [QtGui]
_Znk8QPainter6deviceEv [QtGui]	_Znk8QPainter6matrixEv [QtGui]
_Znk8QPainter6windowEv [QtGui]	_Znk8QPainter7opacityEv [QtGui]
_Znk8QPainter8clipPathEv [QtGui]	_Znk8QPainter8fontInfoEv [QtGui]
_Znk8QPainter8isActiveEv [QtGui]	_Znk8QPainter8viewportEv [QtGui]

_ZNK8QPainter8xFormDevERK5QRect [QtGui]	_ZNK8QPainter8xFormDevERK6QPoint [QtGui]
_ZNK8QPainter8xFormDevERK8QPolygon [QtGui]	_ZNK8QPainter8xFormDevERK8QPolygonii [QtGui]
_ZNK8QPolygon12boundingRectEv [QtGui]	_ZNK8QPolygon5pointEiPiS0_ [QtGui]
_ZNK8QPolygoncv8QVariantEv [QtGui]	_ZNK9QGradient14coordinateModeEv [QtXml]
_ZNK9QGradient5stopsEv [QtGui]	_ZNK9QGradienreqERKS_ [QtGui]
_ZNK9QPolygonF12boundingRectEv [QtGui]	_ZNK9QPolygonF9toPolygonEv [QtGui]
_Zls6QDebugP13QGraphicsItem [QtGui]	_Zls6QDebugRK4QPen [QtGui]
_Zls6QDebugRK6QBrush [QtGui]	_Zls6QDebugRK7QMatrix [QtGui]
_Zls6QDebugRK7QRegion [QtGui]	_Zls6QDebugRK8QPolygon [QtGui]
_Zls6QDebugRK9QPolygonF [QtGui]	_ZlsR11QDataStreamRK12QPainterPath [QtGui]
_ZlsR11QDataStreamRK4QPen [QtGui]	_ZlsR11QDataStreamRK5QIcon [QtXml]
_ZlsR11QDataStreamRK6QBrush [QtGui]	_ZlsR11QDataStreamRK7QMatrix [QtGui]
_ZlsR11QDataStreamRK7QRegion [QtGui]	_ZlsR11QDataStreamRK9QPolygonF [QtGui]
_ZmlRK12QPainterPathRK7QMatrix [QtGui]	_ZrsR11QDataStreamR12QPainterPath [QtGui]
_ZrsR11QDataStreamR4QPen [QtGui]	_ZrsR11QDataStreamR5QIcon [QtXml]
_ZrsR11QDataStreamR6QBrush [QtGui]	_ZrsR11QDataStreamR7QMatrix [QtGui]
_ZrsR11QDataStreamR7QRegion [QtGui]	_ZrsR11QDataStreamR9QPolygonF [QtGui]

An LSB conforming implementation shall provide the generic deprecated functions for Qt4 2D Graphics specified in Table 18-520, with the full mandatory functionality as described in the referenced underlying specification.

Note: These interfaces are deprecated, and applications should avoid using them. These interfaces may be withdrawn in future releases of this specification.

Table 18-520 libQtGui - Qt4 2D Graphics Deprecated Function Interfaces

_ZN22QGraphicsItemAnimation5resetEv [QtXml]	

18.6 Data Definitions for libQtGui

This section defines global identifiers and their values that are associated with interfaces contained in libQtGui. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

18.6.1 QtGui/qabstractbutton.h

```
class QAbstractButton;
```

18.6.2 QtGui/qabstractitemdelegate.h

```
class QAbstractItemDelegate;
enum QAbstractItemDelegate::EndEditHint {
    NoHint = 0,
    EditNextItem = 1,
    EditPreviousItem = 2,
    SubmitModelCache = 3,
    RevertModelCache = 4
};
```

18.6.3 QtGui/qabstractitemview.h

```
class QAbstractItemView;
enum QAbstractItemView::SelectionMode {
    NoSelection = 0,
    SingleSelection = 1,
    MultiSelection = 2,
    ExtendedSelection = 3,
    ContiguousSelection = 4
};
enum QAbstractItemView::SelectionBehavior {
    SelectItems = 0,
    SelectRows = 1,
    SelectColumns = 2
};
enum QAbstractItemView::ScrollHint {
    EnsureVisible = 0,
    PositionAtTop = 1,
    PositionAtBottom = 2
};
enum QAbstractItemView::EditTrigger {
    NoEditTriggers = 0,
    CurrentChanged = 1,
    DoubleClicked = 2,
    SelectedClicked = 4,
    EditKeyPressed = 8,
```

```

        AnyKeyPressed = 16,
        AllEditTriggers = 31
};
class QFlags < QAbstractItemView::EditTrigger >;
typedef class QFlags < QAbstractItemView::EditTrigger >
    QAbstractItemView::EditTriggers;
enum QAbstractItemView::CursorAction {
    MoveUp = 0,
    MoveDown = 1,
    MoveLeft = 2,
    MoveRight = 3,
    MoveHome = 4,
    MoveEnd = 5,
    MovePageUp = 6,
    MovePageDown = 7,
    MoveNext = 8,
    MovePrevious = 9
};
enum QAbstractItemView::State {
    NoState = 0,
    DraggingState = 1,
    DragSelectingState = 2,
    EditingState = 3,
    ExpandingState = 4,
    CollapsingState = 5
};
enum QAbstractItemView::DropIndicatorPosition {
    OnItem = 0,
    AboveItem = 1,
    BelowItem = 2,
    OnViewport = 3
};

```

18.6.4 QtGui/qabstractpagesetupdialog.h

```
class QAbstractPageSetupDialog;
```

18.6.5 QtGui/qabstractprintdialog.h

```

class QAbstractPrintDialog;
enum QAbstractPrintDialog::PrintRange {
    AllPages = 0,
    Selection = 1,
    PageRange = 2
};
enum QAbstractPrintDialog::PrintDialogOption {
    None = 0,
    PrintToFile = 1,
    PrintSelection = 2,
    PrintPageRange = 4,
    PrintCollateCopies = 16
};
class QFlags < QAbstractPrintDialog::PrintDialogOption >;
typedef class QFlags < QAbstractPrintDialog::PrintDialogOption >
    QAbstractPrintDialog::PrintDialogOptions;

```

18.6.6 QtGui/qabstractproxymodel.h

```
class QAbstractProxyModel;
```


18.6.7 QtGui/qabstractscrollarea.h

```
class QAbstractScrollArea;
```

18.6.8 QtGui/qabstractslider.h

```
class QAbstractSlider;
enum QAbstractSlider::SliderAction {
    SliderNoAction = 0,
    SliderSingleStepAdd = 1,
    SliderSingleStepSub = 2,
    SliderPageStepAdd = 3,
    SliderPageStepSub = 4,
    SliderToMinimum = 5,
    SliderToMaximum = 6,
    SliderMove = 7
};
enum QAbstractSlider::SliderChange {
    SliderRangeChange = 0,
    SliderOrientationChange = 1,
    SliderStepsChange = 2,
    SliderValueChange = 3
};
```

18.6.9 QtGui/qabstractspinbox.h

```
class QAbstractSpinBox;
enum QAbstractSpinBox::StepEnabledFlag {
    StepNone = 0,
    StepUpEnabled = 1,
    StepDownEnabled = 2
};
class QFlags < QAbstractSpinBox::StepEnabledFlag >;
typedef class QFlags < QAbstractSpinBox::StepEnabledFlag >
    QAbstractSpinBox::StepEnabled;
enum QAbstractSpinBox::ButtonSymbols {
    UpDownArrows = 0,
    PlusMinus = 1
};
```

18.6.10 QtGui/qabstracttextdocumentlayout.h

```
class QAbstractTextDocumentLayout;
struct QAbstractTextDocumentLayout::Selection;
struct QAbstractTextDocumentLayout::PaintContext;
class QTextObjectInterface;
```

18.6.11 QtGui/qaccessible.h

```
class QAccessible;
enum QAccessible::Event {
    SoundPlayed = 1,
    Alert = 2,
    ForegroundChanged = 3,
    MenuStart = 4,
    MenuEnd = 5,
    PopupMenuStart = 6,
    PopupMenuEnd = 7,
    ContextHelpStart = 12,
```

```

    ContextHelpEnd = 13,
    DragDropStart = 14,
    DragDropEnd = 15,
    DialogStart = 16,
    DialogEnd = 17,
    ScrollingStart = 18,
    ScrollingEnd = 19,
    MenuCommand = 24,
    ObjectCreated = 32768,
    ObjectDestroyed = 32769,
    ObjectShow = 32770,
    ObjectHide = 32771,
    ObjectReorder = 32772,
    Focus = 32773,
    Selection = 32774,
    SelectionAdd = 32775,
    SelectionRemove = 32776,
    SelectionWithin = 32777,
    StateChanged = 32778,
    LocationChanged = 32779,
    NameChanged = 32780,
    DescriptionChanged = 32781,
    ValueChanged = 32782,
    ParentChanged = 32783,
    HelpChanged = 32928,
    DefaultActionChanged = 32944,
    AcceleratorChanged = 32960
};
enum QAccessible::StateFlag {
    Modal = -2147483648,
    Normal = 0,
    Unavailable = 1,
    Selected = 2,
    Focused = 4,
    Pressed = 8,
    Checked = 16,
    Mixed = 32,
    ReadOnly = 64,
    HotTracked = 128,
    DefaultButton = 256,
    Expanded = 512,
    Collapsed = 1024,
    Busy = 2048,
    Marquee = 4096,
    Animated = 8192,
    Invisible = 16384,
    Offscreen = 32768,
    Sizeable = 65536,
    Moveable = 131072,
    Movable = 262144,
    SelfVoicing = 524288,
    Focusable = 1048576,
    Selectable = 2097152,
    Linked = 4194304,
    Traversed = 8388608,
    MultiSelectable = 16777216,
    ExtSelectable = 33554432,
    Protected = 67108864,
    HasPopup = 134217728
};
class QFlags < QAccessible::StateFlag >;
typedef class QFlags < QAccessible::StateFlag > QAccessible::State;
enum QAccessible::Role {
    NoRole = 0,
    TitleBar = 1,
    MenuBar = 2,

```

```

ScrollBar = 3,
Grip = 4,
Sound = 5,
Cursor = 6,
Caret = 7,
AlertMessage = 8,
Window = 9,
Client = 10,
PopupMenu = 11,
MenuItem = 12,
ToolTip = 13,
Application = 14,
Document = 15,
Pane = 16,
Chart = 17,
Dialog = 18,
Border = 19,
Grouping = 20,
Separator = 21,
ToolBar = 22,
StatusBar = 23,
Table = 24,
ColumnHeader = 25,
RowHeader = 26,
Column = 27,
Row = 28,
Cell = 29,
Link = 30,
HelpBalloon = 31,
Assistant = 32,
List = 33,
ListItem = 34,
Tree = 35,
TreeItem = 36,
PageTab = 37,
PropertyPage = 38,
Indicator = 39,
Graphic = 40,
StaticText = 41,
EditableText = 42,
PushButton = 43,
CheckBox = 44,
RadioButton = 45,
ComboBox = 46,
ProgressBar = 48,
Dial = 49,
HotkeyField = 50,
Slider = 51,
SpinBox = 52,
Canvas = 53,
Animation = 54,
Equation = 55,
ButtonDropDown = 56,
ButtonMenu = 57,
ButtonDropGrid = 58,
Whitespace = 59,
PageTabList = 60,
Clock = 61,
Splitter = 62,
LayeredPane = 63,
UserRole = 65535
};
enum QAccessible::Text {
    Name = 0,
    Description = 1,
    Value = 2,

```

```

        Help = 3,
        Accelerator = 4,
        UserText = 65535
    };
    enum QAccessible::RelationFlag {
        Unrelated = 0,
        Self = 1,
        Ancestor = 2,
        Child = 4,
        Descendent = 8,
        Sibling = 16,
        HierarchyMask = 255,
        Up = 256,
        Down = 512,
        Left = 1024,
        Right = 2048,
        Covers = 4096,
        Covered = 8192,
        GeometryMask = 65280,
        FocusChild = 65536,
        Label = 131072,
        Labelled = 262144,
        Controller = 524288,
        Controlled = 1048576,
        LogicalMask = 16711680
    };
    class QFlags < QAccessible::RelationFlag >;
    typedef class QFlags < QAccessible::RelationFlag >
    QAccessible::Relation;
    enum QAccessible::Action {
        LastStandardAction = -11,
        AddToSelection = -11,
        ExtendSelection = -10,
        RemoveSelection = -9,
        ClearSelection = -8,
        Select = -7,
        Cancel = -6,
        Accept = -5,
        Decrease = -4,
        Increase = -3,
        SetFocus = -2,
        FirstStandardAction = -1,
        Press = -1,
        DefaultAction = 0
    };
    typedef void QAccessible::UpdateHandler;
    typedef void QAccessible::RootObjectHandler;
    class QAccessibleInterface;
    class QAccessibleEvent;

```

18.6.12 QtGui/qaccessiblebridge.h

```

class QAccessibleBridge;
struct QAccessibleBridgeFactoryInterface;
class QAccessibleBridgePlugin;

```

18.6.13 QtGui/qaccessibleobject.h

```

class QAccessibleObject;
class QAccessibleApplication;

```

18.6.14 QtGui/qaccessibleplugin.h

```
struct QAccessibleFactoryInterface;
class QAccessiblePlugin;
```

18.6.15 QtGui/qaccessiblewidget.h

```
class QAccessibleWidget;
```

18.6.16 QtGui/qaction.h

```
class QAction;
enum QAction::ActionEvent {
    Trigger = 0,
    Hover = 1
};
```

18.6.17 QtGui/qactiongroup.h

```
class QActionGroup;
```

18.6.18 QtGui/qapplication.h

```
class QApplication;
enum QApplication::Type {
    Tty = 0,
    GuiClient = 1,
    GuiServer = 2
};
enum QApplication::ColorSpec {
    NormalColor = 0,
    CustomColor = 1,
    ManyColor = 2
};
typedef int QApplication::ColorMode;
```

18.6.19 QtGui/qbitmap.h

```
class QBitmap;
```

18.6.20 QtGui/qboxlayout.h

```
class QBoxLayout;
enum QBoxLayout::Direction {
    LeftToRight = 0,
    RightToLeft = 1,
    TopToBottom = 2,
    Down = 2,
    BottomToTop = 3,
    Up = 3
};
class QHBoxLayout;
class QVBoxLayout;
```

18.6.21 QtGui/qbrush.h

```

class QBrush;
struct QBrushData;
class QGradient;
enum QGradient::Type {
    LinearGradient = 0,
    RadialGradient = 1,
    ConicalGradient = 2,
    NoGradient = 3
};
enum QGradient::Spread {
    PadSpread = 0,
    ReflectSpread = 1,
    RepeatSpread = 2
};
class QLinearGradient;
class QRadialGradient;
class QConicalGradient;

```

18.6.22 QtGui/qbuttongroup.h

```

class QButtonGroup;

```

18.6.23 QtGui/qcdestyle.h

```

class QCDEStyle;

```

18.6.24 QtGui/qcheckbox.h

```

class QCheckBox;
enum QCheckBox::ToggleState {
    Off = 0,
    NoChange = 1,
    On = 2
};

```

18.6.25 QtGui/qclipboard.h

```

class QClipboard;
enum QClipboard::Mode {
    Clipboard = 0,
    Selection = 1
};

```

18.6.26 QtGui/qcolor.h

```

class QColor;
enum QColor::Spec {
    Invalid = 0,
    Rgb = 1,
    Hsv = 2,
    Cmyk = 3
};

```

18.6.27 QtGui/qcolordialog.h

```
class QColorDialog;
```

18.6.28 QtGui/qcolormap.h

```
class QColormap;
enum QColormap::Mode {
    Direct = 0,
    Indexed = 1,
    Gray = 2
};
```

18.6.29 QtGui/qcombobox.h

```
class QComboBox;
enum QComboBox::InsertPolicy {
    NoInsert = 0,
    NoInsertion = 0,
    AtTop = 1,
    InsertAtTop = 1,
    AtCurrent = 2,
    InsertAtCurrent = 2,
    InsertAtBottom = 3,
    AtBottom = 3,
    InsertAfterCurrent = 4,
    AfterCurrent = 4,
    InsertBeforeCurrent = 5,
    BeforeCurrent = 5
};
typedef enum QComboBox::InsertPolicy {
    NoInsert = 0,
    NoInsertion = 0,
    AtTop = 1,
    InsertAtTop = 1,
    AtCurrent = 2,
    InsertAtCurrent = 2,
    InsertAtBottom = 3,
    AtBottom = 3,
    InsertAfterCurrent = 4,
    AfterCurrent = 4,
    InsertBeforeCurrent = 5,
    BeforeCurrent = 5
} QComboBox::Policy;
enum QComboBox::SizeAdjustPolicy {
    AdjustToContents = 0,
    AdjustToContentsOnFirstShow = 1,
    AdjustToMinimumContentsLength = 2
};
```

18.6.30 QtGui/qcommonstyle.h

```
class QCommonStyle;
```

18.6.31 QtGui/qcursor.h

```
class QCursor;
typedef enum Qt::CursorShape QCursorShape;
```

18.6.32 QtGui/qdatetimeedit.h

```

class QDateTimeEdit;
enum QDateTimeEdit::Section {
    NoSection = 0,
    AmPmSection = 1,
    MSecSection = 2,
    SecondSection = 4,
    MinuteSection = 8,
    HourSection = 16,
    TimeSections_Mask = 31,
    DaySection = 256,
    MonthSection = 512,
    YearSection = 1024,
    DateSections_Mask = 1792
};
class QFlags < QDateTimeEdit::Section >;
typedef class QFlags < QDateTimeEdit::Section >
QDateTimeEdit::Sections;
class QTimeEdit;
class QDateEdit;

```

18.6.33 QtGui/qdesktopwidget.h

```

class QDesktopWidget;

```

18.6.34 QtGui/qdial.h

```

typedef enum QtValidLicenseForGuiModule QtGuiModule;
class QDial;

```

18.6.35 QtGui/qdialog.h

```

class QDialog;
enum QDialog::DialogCode {
    Rejected = 0,
    Accepted = 1
};

```

18.6.36 QtGui/qdirmodel.h

```

class QFileIconProvider;
enum QFileIconProvider::IconType {
    Computer = 0,
    Desktop = 1,
    Trashcan = 2,
    Network = 3,
    Drive = 4,
    Folder = 5,
    File = 6
};
class QDirModel;
enum QDirModel::Roles {
    FileIconRole = 1,
    FilePathRole = 33,
    FileNameRole = 34
};

```


18.6.37 QtGui/qdockwidget.h

```

class QDockWidget;
enum QDockWidget::DockWidgetFeature {
    NoDockWidgetFeatures = 0,
    DockWidgetClosable = 1,
    DockWidgetMovable = 2,
    DockWidgetFloatable = 4,
    DockWidgetFeatureMask = 7,
    AllDockWidgetFeatures = 7,
    Reserved = 255
};
class QFlags < QDockWidget::DockWidgetFeature >;
typedef class QFlags < QDockWidget::DockWidgetFeature >
    QDockWidget::DockWidgetFeatures;

```

18.6.38 QtGui/qdrag.h

```

class QDrag;

```

18.6.39 QtGui/qerrormessage.h

```

class QErrorMessage;

```

18.6.40 QtGui/qevent.h

```

class QInputEvent;
class QMouseEvent;
class QHoverEvent;
class QWheelEvent;
class QTabletEvent;
enum QTabletEvent::TabletDevice {
    NoDevice = 0,
    Puck = 1,
    Stylus = 2,
    Airbrush = 3,
    FourDMouse = 4,
    XFreeEraser = 5,
    RotationStylus = 6
};
enum QTabletEvent::PointerType {
    UnknownPointer = 0,
    Pen = 1,
    Cursor = 2,
    Eraser = 3
};
class QKeyEvent;
class QFocusEvent;
enum QFocusEvent::Reason {
    Mouse = 0,
    Tab = 1,
    Backtab = 2,
    ActiveWindow = 3,
    Popup = 4,
    Shortcut = 5,
    MenuBar = 6,
    Other = 7
};
class QPaintEvent;
class QMoveEvent;

```

```

class QResizeEvent;
class QCloseEvent;
class QIconDragEvent;
class QShowEvent;
class QHideEvent;
class QContextMenuEvent;
enum QContextMenuEvent::Reason {
    Mouse = 0,
    Keyboard = 1,
    Other = 2
};
class QInputMethodEvent;
enum QInputMethodEvent::AttributeType {
    TextFormat = 0,
    Cursor = 1,
    Language = 2,
    Ruby = 3
};
class QInputMethodEvent::Attribute;
class QDropEvent;
enum QDropEvent::Action {
    Copy = 0,
    Link = 1,
    Move = 2,
    Private = 3,
    UserAction = 3
};
class QDragMoveEvent;
class QDragEnterEvent;
class QDragResponseEvent;
class QDragLeaveEvent;
class QHelpEvent;
class QStatusTipEvent;
class QWhatsThisClickedEvent;
class QActionEvent;
class QFileOpenEvent;
class QToolBarChangeEvent;
class QShortcutEvent;
class QClipboardEvent;
class QWindowStateChangeEvent;
class QMenuBarUpdatedEvent;

```

18.6.41 QtGui/qfiledialog.h

```

class QFileDialog;
enum QFileDialog::ViewMode {
    Detail = 0,
    List = 1
};
enum QFileDialog::FileMode {
    AnyFile = 0,
    ExistingFile = 1,
    Directory = 2,
    ExistingFiles = 3,
    DirectoryOnly = 4
};
enum QFileDialog::AcceptMode {
    AcceptOpen = 0,
    AcceptSave = 1
};
enum QFileDialog::DialogLabel {
    LookIn = 0,
    FileName = 1,
    FileType = 2,
    Accept = 3,

```

```

    Reject = 4
};
enum QFileDialog::Option {
    ShowDirsOnly = 1,
    DontResolveSymlinks = 2,
    DontConfirmOverwrite = 4,
    DontUseSheet = 8,
    DontUseNativeDialog = 16
};
class QFlags < QFileDialog::Option >;
typedef class QFlags < QFileDialog::Option > QFileDialog::Options;
typedef enum QFileDialog::FileMode {
    AnyFile = 0,
    ExistingFile = 1,
    Directory = 2,
    ExistingFiles = 3,
    DirectoryOnly = 4
} QFileDialog::Mode;

```

18.6.42 QtGui/qfocusframe.h

```
class QFocusFrame;
```

18.6.43 QtGui/qfont.h

```

class QFont;
enum QFont::StyleHint {
    Helvetica = 0,
    SansSerif = 0,
    Times = 1,
    Serif = 1,
    TypeWriter = 2,
    Courier = 2,
    OldEnglish = 3,
    Decorative = 3,
    System = 4,
    AnyStyle = 5
};
enum QFont::StyleStrategy {
    PreferDefault = 1,
    PreferBitmap = 2,
    PreferDevice = 4,
    PreferOutline = 8,
    ForceOutline = 16,
    PreferMatch = 32,
    PreferQuality = 64,
    PreferAntialias = 128,
    NoAntialias = 256,
    OpenGLCompatible = 512
};
enum QFont::Weight {
    Light = 25,
    Normal = 50,
    DemiBold = 63,
    Bold = 75,
    Black = 87
};
enum QFont::Style {
    StyleNormal = 0,
    StyleItalic = 1,
    StyleOblique = 2
};
enum QFont::Stretch {

```

```

        UltraCondensed = 50,
        ExtraCondensed = 62,
        Condensed = 75,
        SemiCondensed = 87,
        Unstretched = 100,
        SemiExpanded = 112,
        Expanded = 125,
        ExtraExpanded = 150,
        UltraExpanded = 200
    };

```

18.6.44 QtGui/qfontdatabase.h

```

class QFontDatabase;
enum QFontDatabase::WritingSystem {
    Any = 0,
    Latin = 1,
    Greek = 2,
    Cyrillic = 3,
    Armenian = 4,
    Hebrew = 5,
    Arabic = 6,
    Syriac = 7,
    Thaana = 8,
    Devanagari = 9,
    Bengali = 10,
    Gurmukhi = 11,
    Gujarati = 12,
    Oriya = 13,
    Tamil = 14,
    Telugu = 15,
    Kannada = 16,
    Malayalam = 17,
    Sinhala = 18,
    Thai = 19,
    Lao = 20,
    Tibetan = 21,
    Myanmar = 22,
    Georgian = 23,
    Khmer = 24,
    SimplifiedChinese = 25,
    TraditionalChinese = 26,
    Japanese = 27,
    Korean = 28,
    Vietnamese = 29,
    Other = 30,
    WritingSystemsCount = 31
};

```

18.6.45 QtGui/qfontdialog.h

```
class QFontDialog;
```

18.6.46 QtGui/qfontinfo.h

```
class QFontInfo;
```

18.6.47 QtGui/qfontmetrics.h

```
class QFontMetrics;
```

```
class QFontMetricsF;
```

18.6.48 QtGui/qframe.h

```
class QFrame;
enum QFrame::Shape {
    NoFrame = 0,
    Box = 1,
    Panel = 2,
    WinPanel = 3,
    HLine = 4,
    VLine = 5,
    TabWidgetPanel = 6,
    LineEditPanel = 6,
    ToolBarPanel = 6,
    MenuBarPanel = 6,
    PopupPanel = 6,
    StyledPanel = 6,
    GroupBoxPanel = 6
};
enum QFrame::Shadow {
    Plain = 16,
    Raised = 32,
    Sunken = 48
};
```

18.6.49 QtGui/qgridlayout.h

```
class QGridLayout;
```

18.6.50 QtGui/qgroupbox.h

```
class QGroupBox;
```

18.6.51 QtGui/qheaderview.h

```
class QHeaderView;
enum QHeaderView::ResizeMode {
    Interactive = 0,
    Stretch = 1,
    Custom = 2
};
```

18.6.52 QtGui/qicon.h

```
class QIcon;
enum QIcon::Mode {
    Normal = 0,
    Disabled = 1,
    Active = 2
};
enum QIcon::State {
    On = 0,
    Off = 1
};
enum QIcon::Size {
    Small = 0,
    Automatic = 0,
    Large = 1
};
```

```
};
typedef class QIconQIconSet;
```

18.6.53 QtGui/qiconengine.h

```
class QIconEngine;
```

18.6.54 QtGui/qiconengineplugin.h

```
struct QIconEngineFactoryInterface;
class QIconEnginePlugin;
```

18.6.55 QtGui/qimage.h

```
class QImageTextKeyLang;
class QImage;
enum QImage::InvertMode {
    InvertRgb = 0,
    InvertRgba = 1
};
enum QImage::Format {
    Format_Invalid = 0,
    Format_Mono = 1,
    Format_MonoLSB = 2,
    Format_Indexed8 = 3,
    Format_RGB32 = 4,
    Format_ARGB32 = 5,
    Format_ARGB32_Premultiplied = 6
};
enum QImage::Endian {
    BigEndian = 0,
    LittleEndian = 1,
    IgnoreEndian = 2
};
```

18.6.56 QtGui/qimageiohandler.h

```
class QImageIOHandler;
enum QImageIOHandler::ImageOption {
    Size = 0,
    ClipRect = 1,
    Description = 2,
    ScaledClipRect = 3,
    ScaledSize = 4,
    CompressionRatio = 5,
    Gamma = 6,
    Quality = 7,
    Name = 8,
    SubType = 9,
    IncrementalReading = 10,
    Endianness = 11,
    Animation = 12,
    BackgroundColor = 13
};
struct QImageIOHandlerFactoryInterface;
class QImageIOPlugin;
enum QImageIOPlugin::Capability {
    CanRead = 1,
    CanWrite = 2,
    CanReadIncremental = 4
};
```

```
};
class QFlags < QImageIOPlugin::Capability >;
typedef class QFlags < QImageIOPlugin::Capability >
    QImageIOPlugin::Capabilities;
```

18.6.57 QtGui/qimagereader.h

```
class QImageReader;
enum QImageReader::ImageReaderError {
    UnknownError = 0,
    FileNotFoundError = 1,
    DeviceError = 2,
    UnsupportedFormatError = 3,
    InvalidDataError = 4
};
```

18.6.58 QtGui/qimagewriter.h

```
class QImageWriter;
enum QImageWriter::ImageWriterError {
    UnknownError = 0,
    DeviceError = 1,
    UnsupportedFormatError = 2
};
```

18.6.59 QtGui/qinputcontext.h

```
class QInputContext;
enum QInputContext::StandardFormat {
    PreetitFormat = 0,
    SelectionFormat = 1
};
```

18.6.60 QtGui/qinputcontextfactory.h

```
class QInputContextFactory;
```

18.6.61 QtGui/qinputcontextplugin.h

```
struct QInputContextFactoryInterface;
class QInputContextPlugin;
```

18.6.62 QtGui/qinputdialog.h

```
class QInputDialog;
enum QInputDialog::Type {
    LineEdit = 0,
    SpinBox = 1,
    DoubleSpinBox = 2,
    ComboBox = 3,
    EditableComboBox = 4
};
```

18.6.63 QtGui/qitemdelegate.h

```
class QItemDelegate;
```

18.6.64 QtGui/qitemeditorfactory.h

```
class QItemEditorCreatorBase;
class QItemEditorFactory;
```

18.6.65 QtGui/qitemselectionmodel.h

```
class QItemSelectionRange;
class QItemSelectionModel;
enum QItemSelectionModel::SelectionFlag {
    NoUpdate = 0,
    Clear = 1,
    Select = 2,
    ClearAndSelect = 3,
    Deselect = 4,
    Toggle = 8,
    Current = 16,
    SelectCurrent = 18,
    ToggleCurrent = 24,
    Rows = 32,
    Columns = 64
};
class QFlags < QItemSelectionModel::SelectionFlag >;
typedef class QFlags < QItemSelectionModel::SelectionFlag >
    QItemSelectionModel::SelectionFlags;
class QItemSelection;
```

18.6.66 QtGui/qkeysequence.h

```
class QKeySequence;
enum QKeySequence::SequenceMatch {
    NoMatch = 0,
    PartialMatch = 1,
    ExactMatch = 2,
    Identical = 2
};
enum QKeySequence::SequenceFormat {
    NativeText = 0,
    PortableText = 1
};
```

18.6.67 QtGui/qlabel.h

```
class QLabel;
```

18.6.68 QtGui/qlayout.h

```
class QLayoutIterator;
class QLayout;
enum QLayout::SizeConstraint {
    SetDefaultConstraint = 0,
    Auto = 0,
    SetNoConstraint = 1,
    FreeResize = 1,
    SetMinimumSize = 2,
    Minimum = 2,
    Fixed = 3,
    SetFixedSize = 3,
    SetMaximumSize = 4,
```



```
    SetMinAndMaxSize = 5
};
```

18.6.69 QtGui/qlayoutitem.h

```
class QLayoutItem;
class QSpacerItem;
class QWidgetItem;
```

18.6.70 QtGui/qlcdnumber.h

```
class QLCDNumber;
enum QLCDNumber::Mode {
    Hex = 0,
    HEX = 0,
    Dec = 1,
    DEC = 1,
    Oct = 2,
    OCT = 2,
    Bin = 3,
    BIN = 3
};
enum QLCDNumber::SegmentStyle {
    Outline = 0,
    Filled = 1,
    Flat = 2
};
```

18.6.71 QtGui/qlineedit.h

```
class QLineEdit;
enum QLineEdit::EchoMode {
    Normal = 0,
    NoEcho = 1,
    Password = 2
};
```

18.6.72 QtGui/qlistview.h

```
class QListView;
enum QListView::Movement {
    Static = 0,
    Free = 1,
    Snap = 2
};
enum QListView::Flow {
    LeftToRight = 0,
    TopToBottom = 1
};
enum QListView::ResizeMode {
    Fixed = 0,
    Adjust = 1
};
enum QListView::LayoutMode {
    SinglePass = 0,
    Batched = 1
};
enum QListView::ViewMode {
    ListMode = 0,
    IconMode = 1
};
```

```
};
```

18.6.73 QtGui/qlistwidget.h

```
class QListWidgetItem;  
class QListWidget;
```

18.6.74 QtGui/qmainwindow.h

```
class QMainWindow;
```

18.6.75 QtGui/qmatrix.h

```
class QMatrix;
```

18.6.76 QtGui/qmenu.h

```
class QMenu;
```

18.6.77 QtGui/qmenubar.h

```
class QMenuBar;  
enum QMenuBar::Separator {  
    Never = 0,  
    InWindowsStyle = 1  
};
```

18.6.78 QtGui/qmenudata.h

```
class QMenuItem;
```

18.6.79 QtGui/qmessagebox.h

```
class QMessageBox;  
enum QMessageBox::Icon {  
    NoIcon = 0,  
    Information = 1,  
    Warning = 2,  
    Critical = 3,  
    Question = 4  
};  
enum QMessageBox::Button {  
    NoButton = 0,  
    Ok = 1,  
    Cancel = 2,  
    Yes = 3,  
    No = 4,  
    Abort = 5,  
    Retry = 6,  
    Ignore = 7,  
    YesAll = 8,  
    NoAll = 9,  
    ButtonMask = 255,  
    Default = 256,  
    Escape = 512,  
    FlagMask = 768
```

```
};
```

18.6.80 QtGui/qmime.h

```
class QMimeSource;
```

18.6.81 QtGui/qmotifstyle.h

```
class QMotifStyle;
```

18.6.82 QtGui/qmovie.h

```
class QMovie;
enum QMovie::MovieState {
    NotRunning = 0,
    Paused = 1,
    Running = 2
};
enum QMovie::CacheMode {
    CacheNone = 0,
    CacheAll = 1
};
```

18.6.83 QtGui/qpagesetupdialog.h

```
class QPageSetupDialog;
```

18.6.84 QtGui/qpaintdevice.h

```
class QPaintDevice;
enum QPaintDevice::PaintDeviceMetric {
    PdmWidth = 1,
    PdmHeight = 2,
    PdmWidthMM = 3,
    PdmHeightMM = 4,
    PdmNumColors = 5,
    PdmDepth = 6,
    PdmDpiX = 7,
    PdmDpiY = 8,
    PdmPhysicalDpiX = 9,
    PdmPhysicalDpiY = 10
};
```

18.6.85 QtGui/qpaintengine.h

```
class QTextItem;
enum QTextItem::RenderFlag {
    Dummy = -1,
    RightToLeft = 1,
    Overline = 16,
    Underline = 32,
    StrikeOut = 64
};
class QFlags < QTextItem::RenderFlag >;
typedef class QFlags < QTextItem::RenderFlag >
    QTextItem::RenderFlags;
class QPaintEngine;
enum QPaintEngine::PaintEngineFeature {
```

```

    AllFeatures = -1,
    PrimitiveTransform = 1,
    PatternTransform = 2,
    PixmapTransform = 4,
    PatternBrush = 8,
    LinearGradientFill = 16,
    RadialGradientFill = 32,
    ConicalGradientFill = 64,
    AlphaBlend = 128,
    PorterDuff = 256,
    PainterPaths = 512,
    Antialiasing = 1024,
    BrushStroke = 2048,
    PaintOutsidePaintEvent = 536870912
};
class QFlags < QPaintEngine::PaintEngineFeature >;
typedef class QFlags < QPaintEngine::PaintEngineFeature >
    QPaintEngine::PaintEngineFeatures;
enum QPaintEngine::DirtyFlag {
    DirtyPen = 1,
    DirtyBrush = 2,
    DirtyBrushOrigin = 4,
    DirtyFont = 8,
    DirtyBackground = 16,
    DirtyBackgroundMode = 32,
    DirtyTransform = 64,
    DirtyClipRegion = 128,
    DirtyClipPath = 256,
    DirtyHints = 512,
    DirtyCompositionMode = 1024,
    DirtyClipEnabled = 2048,
    AllDirty = 65535
};
class QFlags < QPaintEngine::DirtyFlag >;
typedef class QFlags < QPaintEngine::DirtyFlag >
    QPaintEngine::DirtyFlags;
enum QPaintEngine::PolygonDrawMode {
    OddEvenMode = 0,
    WindingMode = 1,
    ConvexMode = 2,
    PolylineMode = 3
};
enum QPaintEngine::Type {
    X11 = 0,
    Windows = 1,
    QuickDraw = 2,
    CoreGraphics = 3,
    MacPrinter = 4,
    QWindowSystem = 5,
    PostScript = 6,
    OpenGL = 7,
    Picture = 8,
    SVG = 9,
    Raster = 10,
    User = 50,
    MaxUser = 100
};
class QPaintEngineState;

```

18.6.86 QtGui/qpainter.h

```

class QPainter;
enum QPainter::RenderHint {
    Antialiasing = 1,
    TextAntialiasing = 2,

```

```

    SmoothPixmapTransform = 4
};
class QFlags < QPainter::RenderHint >;
typedef class QFlags < QPainter::RenderHint > QPainter::RenderHints;
enum QPainter::CompositionMode {
    CompositionMode_SourceOver = 0,
    CompositionMode_DestinationOver = 1,
    CompositionMode_Clear = 2,
    CompositionMode_Source = 3,
    CompositionMode_Destination = 4,
    CompositionMode_SourceIn = 5,
    CompositionMode_DestinationIn = 6,
    CompositionMode_SourceOut = 7,
    CompositionMode_DestinationOut = 8,
    CompositionMode_SourceAtop = 9,
    CompositionMode_DestinationAtop = 10,
    CompositionMode_Xor = 11
};

```

18.6.87 QtGui/qpainterpath.h

```

class QPainterPath;
enum QPainterPath::ElementType {
    MoveToElement = 0,
    LineToElement = 1,
    CurveToElement = 2,
    CurveToDataElement = 3
};
class QPainterPath::Element;
class QPainterPathPrivate;
class QPainterPathStroker;

```

18.6.88 QtGui/qpalette.h

```

class QPalette;
enum QPalette::ColorGroup {
    Active = 0,
    Normal = 0,
    Disabled = 1,
    Inactive = 2,
    NColorGroups = 3,
    Current = 4,
    All = 5
};
enum QPalette::ColorRole {
    WindowText = 0,
    Foreground = 0,
    Button = 1,
    Light = 2,
    Midlight = 3,
    Dark = 4,
    Mid = 5,
    Text = 6,
    BrightText = 7,
    ButtonText = 8,
    Base = 9,
    Background = 10,
    Window = 10,
    Shadow = 11,
    Highlight = 12,
    HighlightedText = 13,
    Link = 14,
    LinkVisited = 15,

```

```

        AlternateBase = 16,
        NColorRoles = 17,
        NoRole = 17
    };
    class QColorGroup;

```

18.6.89 QtGui/qpen.h

```
class QPen;
```

18.6.90 QtGui/qpicture.h

```

class QPicture;
typedef void picture_io_handler;
class QPictureIO;

```

18.6.91 QtGui/qpictureformatplugin.h

```

struct QPictureFormatInterface;
class QPictureFormatPlugin;

```

18.6.92 QtGui/qpixmap.h

```

class QPixmap;
enum QPixmap::ColorMode {
    Auto = 0,
    Color = 1,
    Mono = 2
};
enum QPixmap::Type {
    PixmapType = 0,
    BitmapType = 1
};

```

18.6.93 QtGui/qpixmapcache.h

```
class QPixmapCache;
```

18.6.94 QtGui/qplastiquestyle.h

```
class QPlastiqueStyle;
```

18.6.95 QtGui/qpolygon.h

```

class QPolygon;
class QPolygonF;

```

18.6.96 QtGui/qprintdialog.h

```
class QPrintDialog;
```

18.6.97 QtGui/qprintengine.h

```

class QPrintEngine;
enum QPrintEngine::PrintEnginePropertyKey {
    PPK_CollateCopies = 0,
    PPK_ColorMode = 1,
    PPK_Creator = 2,
    PPK_DocumentName = 3,
    PPK_FullPage = 4,
    PPK_NumberOfCopies = 5,
    PPK_Orientation = 6,
    PPK_OutputFileName = 7,
    PPK_PageOrder = 8,
    PPK_PageRect = 9,
    PPK_PageSize = 10,
    PPK_PaperRect = 11,
    PPK_PaperSource = 12,
    PPK_PrinterName = 13,
    PPK_PrinterProgram = 14,
    PPK_Resolution = 15,
    PPK_SelectionOption = 16,
    PPK_SupportedResolutions = 17,
    PPK_WindowsPageSize = 18,
    PPK_FontEmbedding = 19,
    PPK_SuppressSystemPrintStatus = 20,
    PPK_CustomBase = 65280
};

```

18.6.98 QtGui/qprinter.h

```

class QPrinter;
enum QPrinter::PrinterMode {
    ScreenResolution = 0,
    PrinterResolution = 1,
    HighResolution = 2
};
enum QPrinter::Orientation {
    Portrait = 0,
    Landscape = 1
};
enum QPrinter::PageSize {
    A4 = 0,
    B5 = 1,
    Letter = 2,
    Legal = 3,
    Executive = 4,
    A0 = 5,
    A1 = 6,
    A2 = 7,
    A3 = 8,
    A5 = 9,
    A6 = 10,
    A7 = 11,
    A8 = 12,
    A9 = 13,
    B0 = 14,
    B1 = 15,
    B10 = 16,
    B2 = 17,
    B3 = 18,
    B4 = 19,
    B6 = 20,
    B7 = 21,
    B8 = 22,
    B9 = 23,
    C5E = 24,
    Comm10E = 25,
};

```

```

        DLE = 26,
        Folio = 27,
        Ledger = 28,
        Tabloid = 29,
        Custom = 30,
        NPageSize = 30
    };
    enum QPrinter::PageOrder {
        FirstPageFirst = 0,
        LastPageFirst = 1
    };
    enum QPrinter::ColorMode {
        GrayScale = 0,
        Color = 1
    };
    enum QPrinter::PaperSource {
        OnlyOne = 0,
        Lower = 1,
        Middle = 2,
        Manual = 3,
        Envelope = 4,
        EnvelopeManual = 5,
        Auto = 6,
        Tractor = 7,
        SmallFormat = 8,
        LargeFormat = 9,
        LargeCapacity = 10,
        Cassette = 11,
        FormSource = 12
    };
    enum QPrinter::PrinterState {
        Idle = 0,
        Active = 1,
        Aborted = 2,
        Error = 3
    };
    enum QPrinter::OutputFormat {
        NativeFormat = 0,
        PdfFormat = 1
    };
    enum QPrinter::PrintRange {
        AllPages = 0,
        Selection = 1,
        PageRange = 2
    };
    enum QPrinter::PrinterOption {
        PrintToFile = 0,
        PrintSelection = 1,
        PrintPageRange = 2
    };
};

```

18.6.99 QtGui/qprogressbar.h

```

class QProgressBar;
enum QProgressBar::Direction {
    TopToBottom = 0,
    BottomToTop = 1
};

```

18.6.100 QtGui/qprogressdialog.h

```

class QProgressDialog;

```


18.6.101 QtGui/qproxymodel.h

```
class QProxyModel;
```

18.6.102 QtGui/qpushbutton.h

```
class QPushButton;
```

18.6.103 QtGui/qradiobutton.h

```
class QRadioButton;
```

18.6.104 QtGui/qregion.h

```
class QRegion;
enum QRegion::RegionType {
    Rectangle = 0,
    Ellipse = 1
};
struct QRegion::QRegionData;
```

18.6.105 QtGui/qrgb.h

```
typedef unsigned int QRgb;
```

18.6.106 QtGui/qrubberband.h

```
class QRubberBand;
enum QRubberBand::Shape {
    Line = 0,
    Rectangle = 1
};
```

18.6.107 QtGui/qscrollarea.h

```
class QScrollArea;
```

18.6.108 QtGui/qscrollbar.h

```
class QScrollBar;
```

18.6.109 QtGui/qsessionmanager.h

```
class QSessionManager;
enum QSessionManager::RestartHint {
    RestartIfRunning = 0,
    RestartAnyway = 1,
    RestartImmediately = 2,
    RestartNever = 3
};
```

18.6.110 QtGui/qshortcut.h

```
class QShortcut;
```

18.6.111 QtGui/qsizegrip.h

```
class QSizeGrip;
```

18.6.112 QtGui/qsizepolicy.h

```
class QSizePolicy;
enum QSizePolicy::SizePolicyMasks {
    HSize = 4,
    HMask = 15,
    VMask = 240
};
enum QSizePolicy::PolicyFlag {
    GrowFlag = 1,
    ExpandFlag = 2,
    ShrinkFlag = 4,
    IgnoreFlag = 8
};
enum QSizePolicy::Policy {
    Fixed = 0,
    Minimum = 1,
    MinimumExpanding = 3,
    Maximum = 4,
    Preferred = 5,
    Expanding = 7,
    Ignored = 13
};
typedef enum QSizePolicy::Policy {
    Fixed = 0,
    Minimum = 1,
    MinimumExpanding = 3,
    Maximum = 4,
    Preferred = 5,
    Expanding = 7,
    Ignored = 13
} QSizePolicy::SizeType;
typedef Qt::Orientations QSizePolicy::ExpandData;
```

18.6.113 QtGui/qslider.h

```
class QSlider;
enum QSlider::TickPosition {
    NoTicks = 0,
    NoMarks = 0,
    TicksLeft = 1,
    Left = 1,
    Above = 1,
    TicksAbove = 1,
    TicksRight = 2,
    TicksBelow = 2,
    Below = 2,
    Right = 2,
    TicksBothSides = 3,
    Both = 3
};
```

18.6.114 QtGui/qsortfilterproxymodel.h

```
class QSortFilterProxyModel;
```

18.6.115 QtGui/qsound.h

```
class QSound;
```

18.6.116 QtGui/qspinbox.h

```
class QSpinBox;
class QDoubleSpinBox;
```

18.6.117 QtGui/qsplashscreen.h

```
class QSplashScreen;
```

18.6.118 QtGui/qsplitter.h

```
class QSplitter;
enum QSplitter::ResizeMode {
    Stretch = 0,
    KeepSize = 1,
    FollowSizeHint = 2,
    Auto = 3
};
class QSplitterHandle;
```

18.6.119 QtGui/qstackedlayout.h

```
class QStackedLayout;
```

18.6.120 QtGui/qstackedwidget.h

```
class QStackedWidget;
```

18.6.121 QtGui/qstandarditemmodel.h

```
class QStandardItemModel;
```

18.6.122 QtGui/qstatusbar.h

```
class QStatusBar;
```

18.6.123 QtGui/qstringlistmodel.h

```
class QStringListModel;
```

18.6.124 QtGui/qstyle.h

```

class QStyle;
enum QStyle::StateFlag {
    State_None = 0,
    State_Default = 0,
    State_Enabled = 1,
    State_Raised = 2,
    State_Sunken = 4,
    State_Off = 8,
    State_NoChange = 16,
    State_On = 32,
    State_DownArrow = 64,
    State_Horizontal = 128,
    State_HasFocus = 256,
    State_Top = 512,
    State_Bottom = 1024,
    State_FocusAtBorder = 2048,
    State_AutoRaise = 4096,
    State_MouseOver = 8192,
    State_UpArrow = 16384,
    State_Selected = 32768,
    State_Active = 65536,
    State_Open = 262144,
    State_Children = 524288,
    State_Item = 1048576,
    State_Sibling = 2097152,
    State_Editing = 4194304,
    State_KeyboardFocusChange = 8388608,
    State_ReadOnly = 33554432
};
class QFlags < QStyle::StateFlag >;
typedef class QFlags < QStyle::StateFlag > QStyle::State;
typedef QStyle::State QStyle::SFlags;
enum QStyle::PrimitiveElement {
    PE_Q3CheckListController = 0,
    PE_Q3CheckListExclusiveIndicator = 1,
    PE_Q3CheckListIndicator = 2,
    PE_Q3DockWindowSeparator = 3,
    PE_Q3Separator = 4,
    PE_Frame = 5,
    PE_FrameDefaultButton = 6,
    PE_FrameDockWidget = 7,
    PE_FrameFocusRect = 8,
    PE_FrameGroupBox = 9,
    PE_FrameLineEdit = 10,
    PE_FrameMenu = 11,
    PE_FrameStatusBar = 12,
    PE_FrameTabWidget = 13,
    PE_FrameWindow = 14,
    PE_FrameButtonBevel = 15,
    PE_FrameButtonTool = 16,
    PE_FrameTabBarBase = 17,
    PE_PanelButtonCommand = 18,
    PE_PanelButtonBevel = 19,
    PE_PanelButtonTool = 20,
    PE_PanelMenuBar = 21,
    PE_PanelToolBar = 22,
    PE_PanelLineEdit = 23,
    PE_IndicatorArrowDown = 24,
    PE_IndicatorArrowLeft = 25,
    PE_IndicatorArrowRight = 26,
    PE_IndicatorArrowUp = 27,
    PE_IndicatorBranch = 28,
    PE_IndicatorButtonDropDown = 29,
    PE_IndicatorViewItemCheck = 30,
    PE_IndicatorCheckBox = 31,
    PE_IndicatorDockWidgetResizeHandle = 32,

```

```

PE_IndicatorHeaderArrow = 33,
PE_IndicatorMenuCheckMark = 34,
PE_IndicatorProgressChunk = 35,
PE_IndicatorRadioButton = 36,
PE_IndicatorSpinDown = 37,
PE_IndicatorSpinMinus = 38,
PE_IndicatorSpinPlus = 39,
PE_IndicatorSpinUp = 40,
PE_IndicatorToolBarHandle = 41,
PE_IndicatorToolBarSeparator = 42,
PE_PanelTipLabel = 43,
PE_IndicatorTabTear = 44,
PE_CustomBase = 251658240
};

enum QStyle::ControlElement {
    CE_CustomBase = -268435456,
    CE_PushButton = 0,
    CE_PushButtonBevel = 1,
    CE_PushButtonLabel = 2,
    CE_CheckBox = 3,
    CE_CheckBoxLabel = 4,
    CE_RadioButton = 5,
    CE_RadioButtonLabel = 6,
    CE_TabBarTab = 7,
    CE_TabBarTabShape = 8,
    CE_TabBarTabLabel = 9,
    CE_ProgressBar = 10,
    CE_ProgressBarGroove = 11,
    CE_ProgressBarContents = 12,
    CE_ProgressBarLabel = 13,
    CE_MenuItem = 14,
    CE_MenuScroller = 15,
    CE_MenuVMargin = 16,
    CE_MenuHMargin = 17,
    CE_MenuTearoff = 18,
    CE_MenuEmptyArea = 19,
    CE_MenuBarItem = 20,
    CE_MenuBarEmptyArea = 21,
    CE_ToolButtonLabel = 22,
    CE_Header = 23,
    CE_HeaderSection = 24,
    CE_HeaderLabel = 25,
    CE_Q3DockWindowEmptyArea = 26,
    CE_ToolBoxTab = 27,
    CE_SizeGrip = 28,
    CE_Splitter = 29,
    CE_RubberBand = 30,
    CE_DockWidgetTitle = 31,
    CE_ScrollBarAddLine = 32,
    CE_ScrollBarSubLine = 33,
    CE_ScrollBarAddPage = 34,
    CE_ScrollBarSubPage = 35,
    CE_ScrollBarSlider = 36,
    CE_ScrollBarFirst = 37,
    CE_ScrollBarLast = 38,
    CE_FocusFrame = 39,
    CE_ComboBoxLabel = 40,
    CE_ToolBar = 41
};

enum QStyle::SubElement {
    SE_CustomBase = -268435456,
    SE_PushButtonContents = 0,
    SE_PushButtonFocusRect = 1,
    SE_CheckBoxIndicator = 2,
    SE_CheckBoxContents = 3,
    SE_CheckBoxFocusRect = 4,

```

```

SE_CheckBoxClickRect = 5,
SE_RadioButtonIndicator = 6,
SE_RadioButtonContents = 7,
SE_RadioButtonFocusRect = 8,
SE_RadioButtonClickRect = 9,
SE_ComboBoxFocusRect = 10,
SE_SliderFocusRect = 11,
SE_Q3DockWindowHandleRect = 12,
SE_ProgressBarGroove = 13,
SE_ProgressBarContents = 14,
SE_ProgressBarLabel = 15,
SE_DialogButtonAccept = 16,
SE_DialogButtonReject = 17,
SE_DialogButtonApply = 18,
SE_DialogButtonHelp = 19,
SE_DialogButtonAll = 20,
SE_DialogButtonAbort = 21,
SE_DialogButtonIgnore = 22,
SE_DialogButtonRetry = 23,
SE_DialogButtonCustom = 24,
SE_ToolBoxTabContents = 25,
SE_HeaderLabel = 26,
SE_HeaderArrow = 27,
SE_TabWidgetTabBar = 28,
SE_TabWidgetTabPane = 29,
SE_TabWidgetTabContents = 30,
SE_TabWidgetLeftCorner = 31,
SE_TabWidgetRightCorner = 32,
SE_ViewItemCheckIndicator = 33,
SE_TabBarTearIndicator = 34,
SE_TreeViewDisclosureItem = 35
};
enum QStyle::ComplexControl {
    CC_CustomBase = -268435456,
    CC_SpinBox = 0,
    CC_ComboBox = 1,
    CC_ScrollBar = 2,
    CC_Slider = 3,
    CC_ToolButton = 4,
    CC_TitleBar = 5,
    CC_Q3ListView = 6,
    CC_Dial = 7,
    CC_GroupBox = 8
};
enum QStyle::SubControl {
    SC_All = -1,
    SC_None = 0,
    SC_SliderGroove = 1,
    SC_ToolButton = 1,
    SC_ComboBoxFrame = 1,
    SC_TitleBarSysMenu = 1,
    SC_SpinBoxUp = 1,
    SC_ScrollBarAddLine = 1,
    SC_DialGroove = 1,
    SC_Q3ListView = 1,
    SC_GroupBoxCheckBox = 1,
    SC_Q3ListViewBranch = 2,
    SC_TitleBarMinButton = 2,
    SC_SpinBoxDown = 2,
    SC_DialHandle = 2,
    SC_SliderHandle = 2,
    SC_GroupBoxLabel = 2,
    SC_ToolButtonMenu = 2,
    SC_ScrollBarSubLine = 2,
    SC_ComboBoxEditField = 2,
    SC_GroupBoxContents = 4,

```

```

SC_ScrollBarAddPage = 4,
SC_Q3ListViewExpand = 4,
SC_DialTickmarks = 4,
SC_TitleBarMaxButton = 4,
SC_SpinBoxFrame = 4,
SC_ComboBoxArrow = 4,
SC_SliderTickmarks = 4,
SC_GroupBoxFrame = 8,
SC_SpinBoxEditField = 8,
SC_ComboBoxListBoxPopup = 8,
SC_ScrollBarSubPage = 8,
SC_TitleBarCloseButton = 8,
SC_ScrollBarFirst = 16,
SC_TitleBarNormalButton = 16,
SC_ScrollBarLast = 32,
SC_TitleBarShadeButton = 32,
SC_ScrollBarSlider = 64,
SC_TitleBarUnshadeButton = 64,
SC_ScrollBarGroove = 128,
SC_TitleBarContextHelpButton = 128,
SC_TitleBarLabel = 256
};
class QFlags < QStyle::SubControl >;
typedef class QFlags < QStyle::SubControl > QStyle::SubControls;
typedef QStyle::SubControls QStyle::SCFlags;
enum QStyle::PixelMetric {
    PM_CustomBase = -268435456,
    PM_ButtonMargin = 0,
    PM_ButtonDefaultIndicator = 1,
    PM_MenuButtonIndicator = 2,
    PM_ButtonShiftHorizontal = 3,
    PM_ButtonShiftVertical = 4,
    PM_DefaultFrameWidth = 5,
    PM_SpinBoxFrameWidth = 6,
    PM_ComboBoxFrameWidth = 7,
    PM_MaximumDragDistance = 8,
    PM_ScrollBarExtent = 9,
    PM_ScrollBarSliderMin = 10,
    PM_SliderThickness = 11,
    PM_SliderControlThickness = 12,
    PM_SliderLength = 13,
    PM_SliderTickmarkOffset = 14,
    PM_SliderSpaceAvailable = 15,
    PM_DockWidgetSeparatorExtent = 16,
    PM_DockWidgetHandleExtent = 17,
    PM_DockWidgetFrameWidth = 18,
    PM_TabBarTabOverlap = 19,
    PM_TabBarTabHSpace = 20,
    PM_TabBarTabVSpace = 21,
    PM_TabBarBaseHeight = 22,
    PM_TabBarBaseOverlap = 23,
    PM_ProgressBarChunkWidth = 24,
    PM_SplitterWidth = 25,
    PM_TitleBarHeight = 26,
    PM_MenuScrollerHeight = 27,
    PM_MenuHMargin = 28,
    PM_MenuVMargin = 29,
    PM_MenuPanelWidth = 30,
    PM_MenuTearoffHeight = 31,
    PM_MenuDesktopFrameWidth = 32,
    PM_MenuBarPanelWidth = 33,
    PM_MenuBarItemSpacing = 34,
    PM_MenuBarVMargin = 35,
    PM_MenuBarHMargin = 36,
    PM_IndicatorWidth = 37,
    PM_IndicatorHeight = 38,

```

```

PM_ExclusiveIndicatorWidth = 39,
PM_ExclusiveIndicatorHeight = 40,
PM_CheckListButtonSize = 41,
PM_CheckListControllerSize = 42,
PM_DialogButtonsSeparator = 43,
PM_DialogButtonsButtonWidth = 44,
PM_DialogButtonsButtonHeight = 45,
PM_MDIFrameWidth = 46,
PM_MDIMinimizedWidth = 47,
PM_HeaderMargin = 48,
PM_HeaderMarkSize = 49,
PM_HeaderGripMargin = 50,
PM_TabBarTabShiftHorizontal = 51,
PM_TabBarTabShiftVertical = 52,
PM_TabBarScrollButtonWidth = 53,
PM_ToolBarFrameWidth = 54,
PM_ToolBarHandleExtent = 55,
PM_ToolBarItemSpacing = 56,
PM_ToolBarItemMargin = 57,
PM_ToolBarSeparatorExtent = 58,
PM_ToolBarExtensionExtent = 59,
PM_SpinBoxSliderHeight = 60,
PM_DefaultTopLevelMargin = 61,
PM_DefaultChildMargin = 62,
PM_DefaultLayoutSpacing = 63,
PM_ToolBarIconSize = 64,
PM_ListViewIconSize = 65,
PM_IconViewIconSize = 66,
PM_SmallIconSize = 67,
PM_LargeIconSize = 68,
PM_FocusFrameVMargin = 69,
PM_FocusFrameHMargin = 70,
PM_ToolTipLabelFrameWidth = 71,
PM_CheckBoxLabelSpacing = 72,
PM_TabBarIconSize = 73,
PM_SizeGripSize = 74,
PM_DockWidgetTitleMargin = 75
};

enum QStyle::ContentsType {
    CT_CustomBase = -268435456,
    CT_PushButton = 0,
    CT_CheckBox = 1,
    CT_RadioButton = 2,
    CT_ToolButton = 3,
    CT_ComboBox = 4,
    CT_Splitter = 5,
    CT_Q3DockWindow = 6,
    CT_ProgressBar = 7,
    CT_MenuItem = 8,
    CT_MenuBarItem = 9,
    CT_MenuBar = 10,
    CT_Menu = 11,
    CT_TabBarTab = 12,
    CT_Slider = 13,
    CT_ScrollBar = 14,
    CT_Q3Header = 15,
    CT_LineEdit = 16,
    CT_SpinBox = 17,
    CT_SizeGrip = 18,
    CT_TabWidget = 19,
    CT_DialogButtons = 20,
    CT_HeaderSection = 21,
    CT_GroupBox = 22
};

enum QStyle::StyleHint {
    SH_CustomBase = -268435456,

```


SH_EtchDisabledText = 0,
 SH_DitherDisabledText = 1,
 SH_ScrollBar_MiddleClickAbsolutePosition = 2,
 SH_ScrollBar_ScrollWhenPointerLeavesControl = 3,
 SH_TabBar_SelectMouseType = 4,
 SH_TabBar_Alignment = 5,
 SH_Header_ArrowAlignment = 6,
 SH_Slider_SnapToValue = 7,
 SH_Slider_SloppyKeyEvents = 8,
 SH_ProgressDialog_CenterCancelButton = 9,
 SH_ProgressDialog_TextLabelAlignment = 10,
 SH_PrintDialog_RightAlignButtons = 11,
 SH_MainWindow_SpaceBelowMenuBar = 12,
 SH_FontDialog_SelectAssociatedText = 13,
 SH_Menu_AllowActiveAndDisabled = 14,
 SH_Menu_SpaceActivatesItem = 15,
 SH_Menu_SubMenuPopupDelay = 16,
 SH_ScrollView_FrameOnlyAroundContents = 17,
 SH_MenuBar_AltKeyNavigation = 18,
 SH_ComboBox_ListMouseTracking = 19,
 SH_Menu_MouseTracking = 20,
 SH_MenuBar_MouseTracking = 21,
 SH_ItemView_ChangeHighlightOnFocus = 22,
 SH_Widget_ShareActivation = 23,
 SH_Workspace_FillSpaceOnMaximize = 24,
 SH_ComboBox_Popup = 25,
 SH_TitleBar_NoBorder = 26,
 SH_ScrollBar_StopMouseOverSlider = 27,
 SH_BlinkCursorWhenTextSelected = 28,
 SH_RichText_FullWidthSelection = 29,
 SH_Menu_Scrollable = 30,
 SH_GroupBox_TextLabelVerticalAlignment = 31,
 SH_GroupBox_TextLabelColor = 32,
 SH_Menu_SloppySubMenus = 33,
 SH_Table_GridLineColor = 34,
 SH_LineEdit_PasswordCharacter = 35,
 SH_DialogButtons_DefaultButton = 36,
 SH_ToolBox_SelectedPageTitleBold = 37,
 SH_TabBar_PreferNoArrows = 38,
 SH_ScrollBar_LeftClickAbsolutePosition = 39,
 SH_Q3ListViewExpand_SelectMouseType = 40,
 SH_UnderlineShortcut = 41,
 SH_UnderlineAccelerator = 41,
 SH_SpinBox_AnimateButton = 42,
 SH_SpinBox_KeyPressAutoRepeatRate = 43,
 SH_SpinBox_ClickAutoRepeatRate = 44,
 SH_Menu_FillScreenWithScroll = 45,
 SH_ToolTipLabel_Opacity = 46,
 SH_DrawMenuBarSeparator = 47,
 SH_TitleBar_ModifyNotification = 48,
 SH_Button_FocusPolicy = 49,
 SH_MenuBar_DismissOnSecondClick = 50,
 SH_MessageBox_UseBorderForButtonSpacing = 51,
 SH_TitleBar_AutoRaise = 52,
 SH_ToolButton_PopupDelay = 53,
 SH_FocusFrame_Mask = 54,
 SH_RubberBand_Mask = 55,
 SH_WindowFrame_Mask = 56,
 SH_SpinControls_DisableOnBounds = 57,
 SH_Dial_BackgroundRole = 58,
 SH_ComboBox_LayoutDirection = 59,
 SH_ItemView_EllipsisLocation = 60,
 SH_ItemView_ShowDecorationSelected = 61,
 SH_ItemView_ActivateItemOnSingleClick = 62,
 SH_ScrollBar_ContextMenu = 63,
 SH_ScrollBar_RollBetweenButtons = 64,

```

    SH_GUIStyle = 256,
    SH_ScrollBar_BackgroundMode = 257
};
enum QStyle::StandardPixmap {
    SP_CustomBase = -268435456,
    SP_TitleBarMenuButton = 0,
    SP_TitleBarMinButton = 1,
    SP_TitleBarMaxButton = 2,
    SP_TitleBarCloseButton = 3,
    SP_TitleBarNormalButton = 4,
    SP_TitleBarShadeButton = 5,
    SP_TitleBarUnshadeButton = 6,
    SP_TitleBarContextHelpButton = 7,
    SP_DockWidgetCloseButton = 8,
    SP_MessageBoxInformation = 9,
    SP_MessageBoxWarning = 10,
    SP_MessageBoxCritical = 11,
    SP_MessageBoxQuestion = 12,
    SP_DesktopIcon = 13,
    SP_TrashIcon = 14,
    SP_ComputerIcon = 15,
    SP_DriveFDIcon = 16,
    SP_DriveHDIcon = 17,
    SP_DriveCDIcon = 18,
    SP_DriveDVDIcon = 19,
    SP_DriveNetIcon = 20,
    SP_DirOpenIcon = 21,
    SP_DirClosedIcon = 22,
    SP_DirLinkIcon = 23,
    SP_FileIcon = 24,
    SP_FileLinkIcon = 25,
    SP_ToolBarHorizontalExtensionButton = 26,
    SP_ToolBarVerticalExtensionButton = 27,
    SP_FileDialogStart = 28,
    SP_FileDialogEnd = 29,
    SP_FileDialogToParent = 30,
    SP_FileDialogNewFolder = 31,
    SP_FileDialogDetailedView = 32,
    SP_FileDialogInfoView = 33,
    SP_FileDialogContentsView = 34,
    SP_FileDialogListView = 35,
    SP_FileDialogBack = 36
};

```

18.6.125 QtGui/qstylefactory.h

```
class QStyleFactory;
```

18.6.126 QtGui/qstyleoption.h

```

class QStyleOption;
enum QStyleOption::OptionType {
    SO_Default = 0,
    SO_FocusRect = 1,
    SO_Button = 2,
    SO_Tab = 3,
    SO_MenuItem = 4,
    SO_Frame = 5,
    SO_ProgressBar = 6,
    SO_ToolBox = 7,
    SO_Header = 8,
    SO_Q3DockWindow = 9,
    SO_DockWidget = 10,
};

```

```

SO_Q3ListViewItem = 11,
SO_ViewItem = 12,
SO_TabWidgetFrame = 13,
SO_TabBarBase = 14,
SO_RubberBand = 15,
SO_ToolBar = 16,
SO_CustomBase = 3840,
SO_Complex = 983040,
SO_Slider = 983041,
SO_SpinBox = 983042,
SO_ToolButton = 983043,
SO_ComboBox = 983044,
SO_Q3ListView = 983045,
SO_TitleBar = 983046,
SO_GroupBox = 983047,
SO_ComplexCustomBase = 251658240
};
class QStyleOptionFocusRect;
class QStyleOptionFrame;
class QStyleOptionFrameV2;
enum QStyleOptionFrameV2::FrameFeature {
    None = 0,
    Flat = 1
};
class QFlags < QStyleOptionFrameV2::FrameFeature >;
typedef class QFlags < QStyleOptionFrameV2::FrameFeature >
    QStyleOptionFrameV2::FrameFeatures;
class QStyleOptionTabWidgetFrame;
class QStyleOptionTabBarBase;
class QStyleOptionHeader;
enum QStyleOptionHeader::SectionPosition {
    Beginning = 0,
    Middle = 1,
    End = 2,
    OnlyOneSection = 3
};
enum QStyleOptionHeader::SelectedPosition {
    NotAdjacent = 0,
    NextIsSelected = 1,
    PreviousIsSelected = 2,
    NextAndPreviousAreSelected = 3
};
enum QStyleOptionHeader::SortIndicator {
    None = 0,
    SortUp = 1,
    SortDown = 2
};
class QStyleOptionButton;
enum QStyleOptionButton::ButtonFeature {
    None = 0,
    Flat = 1,
    HasMenu = 2,
    DefaultButton = 4,
    AutoDefaultButton = 8
};
class QFlags < QStyleOptionButton::ButtonFeature >;
typedef class QFlags < QStyleOptionButton::ButtonFeature >
    QStyleOptionButton::ButtonFeatures;
class QStyleOptionTab;
enum QStyleOptionTab::TabPosition {
    Beginning = 0,
    Middle = 1,
    End = 2,
    OnlyOneTab = 3
};
enum QStyleOptionTab::SelectedPosition {

```

```

        NotAdjacent = 0,
        NextIsSelected = 1,
        PreviousIsSelected = 2
    };
    enum QStyleOptionTab::CornerWidget {
        NoCornerWidgets = 0,
        LeftCornerWidget = 1,
        RightCornerWidget = 2
    };
    class QFlags < QStyleOptionTab::CornerWidget >;
    typedef class QFlags < QStyleOptionTab::CornerWidget >
        QStyleOptionTab::CornerWidgets;
    class QStyleOptionTabV2;
    class QStyleOptionToolBar;
    enum QStyleOptionToolBar::ToolBarPosition {
        Beginning = 0,
        Middle = 1,
        End = 2,
        OnlyOne = 3
    };
    enum QStyleOptionToolBar::ToolBarFeature {
        None = 0,
        Movable = 1
    };
    class QFlags < QStyleOptionToolBar::ToolBarFeature >;
    typedef class QFlags < QStyleOptionToolBar::ToolBarFeature >
        QStyleOptionToolBar::ToolBarFeatures;
    class QStyleOptionProgressBar;
    class QStyleOptionProgressBarV2;
    class QStyleOptionMenuItem;
    enum QStyleOptionMenuItem::MenuItemType {
        Normal = 0,
        DefaultItem = 1,
        Separator = 2,
        SubMenu = 3,
        Scroller = 4,
        TearOff = 5,
        Margin = 6,
        EmptyArea = 7
    };
    enum QStyleOptionMenuItem::CheckType {
        NotCheckable = 0,
        Exclusive = 1,
        NonExclusive = 2
    };
    class QStyleOptionQ3ListViewItem;
    enum QStyleOptionQ3ListViewItem::Q3ListViewItemFeature {
        None = 0,
        Expandable = 1,
        MultiLine = 2,
        Visible = 4,
        ParentControl = 8
    };
    class QFlags < QStyleOptionQ3ListViewItem::Q3ListViewItemFeature >;
    typedef class QFlags <
        QStyleOptionQ3ListViewItem::Q3ListViewItemFeature >
        QStyleOptionQ3ListViewItem::Q3ListViewItemFeatures;
    class QStyleOptionQ3DockWindow;
    class QStyleOptionDockWidget;
    class QStyleOptionViewItem;
    enum QStyleOptionViewItem::Position {
        Left = 0,
        Right = 1,
        Top = 2,
        Bottom = 3
    };
};

```

```

class QStyleOptionToolBox;
class QStyleOptionRubberBand;
class QStyleOptionComplex;
class QStyleOptionSlider;
class QStyleOptionSpinBox;
class QStyleOptionQ3ListView;
class QStyleOptionToolButton;
enum QStyleOptionToolButton::ToolButtonFeature {
    None = 0,
    Arrow = 1,
    Menu = 4,
    PopupDelay = 8
};
class QFlags < QStyleOptionToolButton::ToolButtonFeature >;
typedef class QFlags < QStyleOptionToolButton::ToolButtonFeature >
    QStyleOptionToolButton::ToolButtonFeatures;
class QStyleOptionComboBox;
class QStyleOptionTitleBar;
class QStyleOptionGroupBox;
class QStyleHintReturn;
enum QStyleHintReturn::HintReturnType {
    SH_Default = 61440,
    SH_Mask = 61441
};
class QStyleHintReturnMask;

```

18.6.127 QtGui/qstylepainter.h

```
class QStylePainter;
```

18.6.128 QtGui/qstyleplugin.h

```

struct QStyleFactoryInterface;
class QStylePlugin;

```

18.6.129 QtGui/qsyntaxhighlighter.h

```
class QSyntaxHighlighter;
```

18.6.130 QtGui/qtabbar.h

```

class QTabBar;
enum QTabBar::Shape {
    RoundedNorth = 0,
    RoundedAbove = 0,
    RoundedSouth = 1,
    RoundedBelow = 1,
    RoundedWest = 2,
    RoundedEast = 3,
    TriangularNorth = 4,
    TriangularAbove = 4,
    TriangularBelow = 5,
    TriangularSouth = 5,
    TriangularWest = 6,
    TriangularEast = 7
};

```

18.6.131 QtGui/qtableview.h

```
class QTableView;
```

18.6.132 QtGui/qtablewidget.h

```
class QTableWidgetSelectionRange;
class QTableWidgetItem;
class QTableWidget;
```

18.6.133 QtGui/qtabwidget.h

```
class QTabWidget;
enum QTabWidget::TabPosition {
    North = 0,
    Top = 0,
    South = 1,
    Bottom = 1,
    West = 2,
    East = 3
};
enum QTabWidget::TabShape {
    Rounded = 0,
    Triangular = 1
};
```

18.6.134 QtGui/qtextbrowser.h

```
class QTextBrowser;
```

18.6.135 QtGui/qtextcursor.h

```
class QTextCursor;
enum QTextCursor::MoveMode {
    MoveAnchor = 0,
    KeepAnchor = 1
};
enum QTextCursor::MoveOperation {
    NoMove = 0,
    Start = 1,
    Up = 2,
    StartOfLine = 3,
    StartOfBlock = 4,
    StartOfWord = 5,
    PreviousBlock = 6,
    PreviousCharacter = 7,
    PreviousWord = 8,
    Left = 9,
    WordLeft = 10,
    End = 11,
    Down = 12,
    EndOfLine = 13,
    EndOfWord = 14,
    EndOfBlock = 15,
    NextBlock = 16,
    NextCharacter = 17,
    NextWord = 18,
    Right = 19,
    WordRight = 20
};
enum QTextCursor::SelectionType {
    WordUnderCursor = 0,
```

```

    LineUnderCursor = 1,
    BlockUnderCursor = 2
};

```

18.6.136 QtGui/qtextdocument.h

```

enum Qt::HitTestAccuracy {
    ExactHit = 0,
    FuzzyHit = 1
};
enum Qt::WhiteSpaceMode {
    WhiteSpaceModeUndefined = -1,
    WhiteSpaceNormal = 0,
    WhiteSpacePre = 1,
    WhiteSpaceNoWrap = 2
};
class QAbstractUndoItem;
class QTextDocument;
enum QTextDocument::MetaInformation {
    DocumentTitle = 0
};
enum QTextDocument::FindFlag {
    FindBackward = 1,
    FindCaseSensitively = 2,
    FindWholeWords = 4
};
class QFlags < QTextDocument::FindFlag >;
typedef class QFlags < QTextDocument::FindFlag >
    QTextDocument::FindFlags;
enum QTextDocument::ResourceType {
    HtmlResource = 1,
    ImageResource = 2,
    UserResource = 100
};

```

18.6.137 QtGui/qtextdocumentfragment.h

```

class QTextDocumentFragment;

```

18.6.138 QtGui/qtextedit.h

```

class QTextEdit;
enum QTextEdit::LineWrapMode {
    NoWrap = 0,
    WidgetWidth = 1,
    FixedPixelWidth = 2,
    FixedColumnWidth = 3
};
enum QTextEdit::AutoFormattingFlag {
    AutoAll = -1,
    AutoNone = 0,
    AutoBulletList = 1
};
class QFlags < QTextEdit::AutoFormattingFlag >;
typedef class QFlags < QTextEdit::AutoFormattingFlag >
    QTextEdit::AutoFormatting;
enum QTextEdit::CursorAction {
    MoveBackward = 0,
    MoveForward = 1,
    MoveWordBackward = 2,
    MoveWordForward = 3,
    MoveUp = 4,

```

```

        MoveDown = 5,
        MoveLineStart = 6,
        MoveLineEnd = 7,
        MoveHome = 8,
        MoveEnd = 9,
        MovePageUp = 10,
        MovePgUp = 10,
        MovePageDown = 11,
        MovePgDown = 11
    };
    enum QTextEdit::KeyboardAction {
        ActionBackspace = 0,
        ActionDelete = 1,
        ActionReturn = 2,
        ActionKill = 3,
        ActionWordBackspace = 4,
        ActionWordDelete = 5
    };

```

18.6.139 QtGui/qtextformat.h

```

class QTextLength;
enum QTextLength::Type {
    VariableLength = 0,
    FixedLength = 1,
    PercentageLength = 2
};
class QTextFormat;
enum QTextFormat::FormatType {
    InvalidFormat = -1,
    BlockFormat = 1,
    CharFormat = 2,
    ListFormat = 3,
    TableFormat = 4,
    FrameFormat = 5,
    UserFormat = 100
};
enum QTextFormat::Property {
    ObjectIndex = 0,
    CssFloat = 2048,
    LayoutDirection = 2049,
    OutlinePen = 2064,
    BackgroundBrush = 2080,
    ForegroundBrush = 2081,
    BlockAlignment = 4112,
    BlockTopMargin = 4144,
    BlockBottomMargin = 4145,
    BlockLeftMargin = 4146,
    BlockRightMargin = 4147,
    TextIndent = 4148,
    BlockIndent = 4160,
    BlockNonBreakableLines = 4176,
    BlockTrailingHorizontalRulerWidth = 4192,
    FontFamily = 8192,
    FontPointSize = 8193,
    FontSizeAdjustment = 8194,
    FontSizeIncrement = 8194,
    FontWeight = 8195,
    FontItalic = 8196,
    FontUnderline = 8197,
    FontOverline = 8198,
    FontStrikeOut = 8199,
    FontFixedPitch = 8200,
    FontPixelSize = 8201,
    TextUnderlineColor = 8208,

```



```

TextVerticalAlignment = 8225,
TextOutline = 8226,
IsAnchor = 8240,
AnchorHref = 8241,
AnchorName = 8242,
ObjectType = 12032,
ListStyle = 12288,
ListIndent = 12289,
FrameBorder = 16384,
FrameMargin = 16385,
FramePadding = 16386,
FrameWidth = 16387,
FrameHeight = 16388,
TableColumns = 16640,
TableColumnWidthConstraints = 16641,
TableCellSpacing = 16642,
TableCellPadding = 16643,
TableCellRowSpan = 18448,
TableCellColumnSpan = 18449,
ImageName = 20480,
ImageWidth = 20496,
ImageHeight = 20497,
UserProperty = 1048576
};
enum QTextFormat::ObjectTypes {
    NoObject = 0,
    ImageObject = 1,
    TableObject = 2,
    UserObject = 4096
};
class QTextCharFormat;
enum QTextCharFormat::VerticalAlignment {
    AlignNormal = 0,
    AlignSuperScript = 1,
    AlignSubScript = 2
};
class QTextBlockFormat;
class QTextListFormat;
enum QTextListFormat::Style {
    ListUpperAlpha = 0,
    ListLowerAlpha = -5,
    ListDecimal = -4,
    ListSquare = -3,
    ListCircle = -2,
    ListDisc = -1,
    ListStyleUndefined = 0
};
class QTextImageFormat;
class QTextFrameFormat;
enum QTextFrameFormat::Position {
    InFlow = 0,
    FloatLeft = 1,
    FloatRight = 2
};
class QTextTableFormat;

```

18.6.140 QtGui/qtextlayout.h

```

class QTextInlineObject;
class QTextLayout;
struct QTextLayout::FormatRange;
enum QTextLayout::CursorMode {
    SkipCharacters = 0,
    SkipWords = 1
};

```

```

class QTextLine;
enum QTextLine::Edge {
    Leading = 0,
    Trailing = 1
};
enum QTextLine::CursorPosition {
    CursorBetweenCharacters = 0,
    CursorOnCharacter = 1
};

```

18.6.141 QtGui/qttextlist.h

```

class QTextList;

```

18.6.142 QtGui/qttextobject.h

```

class QTextObject;
class QTextBlockGroup;
class QTextFrameLayoutData;
class QTextFrame;
class QTextFrame::iterator;
typedef class QTextFrame::iterator QTextFrame::Iterator;
class QTextBlockUserData;
class QTextBlock;
class QTextBlock::iterator;
typedef class QTextBlock::iterator QTextBlock::Iterator;
class QTextFragment;

```

18.6.143 QtGui/qttextoption.h

```

class QTextOption;
enum QTextOption::WrapMode {
    NoWrap = 0,
    WordWrap = 1,
    ManualWrap = 2,
    WrapAnywhere = 3,
    WrapAtWordBoundaryOrAnywhere = 4
};
enum QTextOption::Flag {
    IncludeTrailingSpaces = -2147483648
};
class QFlags < QTextOption::Flag >;
typedef class QFlags < QTextOption::Flag > QTextOption::Flags;

```

18.6.144 QtGui/qttexttable.h

```

class QTextTableCell;
class QTextTable;

```

18.6.145 QtGui/qttoolbar.h

```

class QToolBar;

```

18.6.146 QtGui/qttoolbox.h

```

class QToolBox;

```

18.6.147 QtGui/qttoolbarbutton.h

```

class QToolButton;
enum QToolButton::ToolButtonPopupMode {
    DelayedPopup = 0,
    MenuButtonPopup = 1,
    InstantPopup = 2
};
enum QToolButton::TextPosition {
    BesideIcon = 0,
    Right = 0,
    BelowIcon = 1,
    Under = 1
};

```

18.6.148 QtGui/qttooltip.h

```

class QToolTip;

```

18.6.149 QtGui/qttreeview.h

```

class QTreeView;

```

18.6.150 QtGui/qtreewidget.h

```

class QTreeWidgetItem;
class QTreeWidget;

```

18.6.151 QtGui/qtreewidgetitemiterator.h

```

class QTreeWidgetItemIterator;
enum QTreeWidgetItemIterator::IteratorFlag {
    All = 0,
    Hidden = 1,
    NotHidden = 2,
    Selected = 4,
    Unselected = 8,
    Selectable = 16,
    NotSelectable = 32,
    DragEnabled = 64,
    DragDisabled = 128,
    DropEnabled = 256,
    DropDisabled = 512,
    HasChildren = 1024,
    NoChildren = 2048,
    Checked = 4096,
    NotChecked = 8192,
    Enabled = 16384,
    Disabled = 32768,
    Editable = 65536,
    NotEditable = 131072,
    UserFlag = 16777216
};
class QFlags < QTreeWidgetItemIterator::IteratorFlag >;
typedef class QFlags < QTreeWidgetItemIterator::IteratorFlag >
    QTreeWidgetItemIterator::IteratorFlags;

```

18.6.152 QtGui/qvalidator.h

```

class QValidator;
enum QValidator::State {
    Invalid = 0,
    Intermediate = 1,
    Valid = 1,
    Acceptable = 2
};
class QIntValidator;
class QDoubleValidator;
class QRegExpValidator;

```

18.6.153 QtGui/qvfbhdr.h

```

struct QVFbHeader;
struct QVFbKeyData;

```

18.6.154 QtGui/qwhatsthis.h

```

class QWhatsThis;

```

18.6.155 QtGui/qwidget.h

```

class QWidgetData;
class QWidget;
enum QWidget::BackgroundOrigin {
    WidgetOrigin = 0,
    ParentOrigin = 1,
    WindowOrigin = 2,
    AncestorOrigin = 3
};

```

18.6.156 QtGui/qwindowdefs.h

```

typedef class QList < QWidget * > QWidgetList;
typedef unsigned long int WId;

```

18.6.157 QtGui/qwindowsstyle.h

```

class QWindowsStyle;

```

18.6.158 QtGui/qwmatrix.h

```

typedef class QMatrix QWMatrix;

```

18.6.159 QtGui/qworkspace.h

```

class QWorkspace;
enum QWorkspace::WindowOrder {
    CreationOrder = 0,
    StackingOrder = 1
};

```

18.7 Interface Definitions for libQtGui

The interfaces defined on the following pages are included in libQtGui and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 18.5 shall behave as described in the referenced base document.

_ZN11QGridLayout21setDefaultPositioningEiN2Qt11OrientationE

Name

`QGridLayout::setDefaultPositioning` — set the positioning mode

Synopsis

```
#include <QtGui/qgridlayout.h>
void QGridLayout::setDefaultPositioning (int n, Qt::Orientation orient);
```

Description

The `QGridLayout::setDefaultPositioning()` function shall set the positioning mode used by `QGridLayout::addItem()`.

If *orient* is `Qt::Horizontal`, the layout is expanded to *n* columns, and items will be added columns-first. Otherwise it is expanded to *n* rows and items will be added rows-first.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN11QHeaderView13updateSectionEi

Name

`QHeaderView::updateSection` — update HeaderView section

Synopsis

```
#include <QtGui/qheaderview.h>
void QHeaderView::updateSection (int logicalIndex);
```

Description

The `QHeaderView::updateSection()` function shall update the section specified by the given *logicalIndex*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN12QPaintEngine13setSystemClipERK7QRegion**Name**

QPaintEngine::setSystemClip — set the system clip for the engine

Synopsis

```
#include <QtGui/qpaintengine.h>
void QPaintEngine::setSystemClip (const QRegion &region);
```

Description

The `QPaintEngine::setSystemClip()` function shall set the system clip for the engine. The system clip defines the basis area that the engine has to draw in. All clips that are set must be an intersection with the system clip.

The system clip can be reset to no clip by setting an empty region.

The function takes effect only if the engine is not active.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN12QPaintEngine13setSystemRectERK5QRect**Name**

QPaintEngine::setSystemRect — set the target rect for drawing within the backing store

Synopsis

```
#include <QtGui/qpaintengine.h>
void QPaintEngine::setSystemRect (const QRect &rect);
```

Description

The `QPaintEngine::setSystemRect()` function shall set the target rect for drawing within the backing store.

The function takes effect only if the engine is not active.

This function should only be used by the backing store, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN12QPaintEngine14setPaintDeviceEP12QPaintDevice**Name**

`QPaintEngine::setPaintDevice` — set the paintdevice that the engine operates on

Synopsis

```
#include <QtGui/qpaintengine.h>
void QPaintEngine::setPaintDevice (QPaintDevice *device);
```

Description

The `QPaintEngine::setPaintDevice()` function shall set the paint device that this engine operates on to *device*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

QPixmapCache::cacheLimit**Name**

`QPixmapCache::cacheLimit` — UUID greater-than comparison operator

Synopsis

```
public class QPixmapCache {
    bool operator >(const QPixmapCache& other);
}
```

Description

`QPixmapCache::cacheLimit` has the same specification as in QtGui 4.2.0, except for differences noted below.

Default Value

Default value of the cache limit is unspecified.

_ZN13QGraphicsItem10addToIndexEv**Name**

`QGraphicsItem::addToIndex` — add the item from the scene's index

Synopsis

```
#include <QtGui/qgraphicsitem.h>
void QGraphicsItem::addToIndex (void);
```

Description

The `QGraphicsItem::addToIndex()` function shall add the calling object to the scene's index.

This function can be called in conjunction with the `removeFromIndex()` function to ensure the index bookkeeping is correct when the item's position, transformation or shape changes.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN13QGraphicsItem12setExtensionENS_9ExtensionERK8QVariant**Name**

`QGraphicsItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsItem::setExtension(Extension extension, const
QVariant & variant);
```

Description

The `QGraphicsItem::setExtension()` function doesn't do anything. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN13QGraphicsItem15removeFromIndexEv**Name**

`QGraphicsItem::removeFromIndex` — remove the item from the scene's index

Synopsis

```
#include <QtGui/qgraphicsitem.h>
void QGraphicsItem::removeFromIndex (void);
```

Description

The `QGraphicsItem::removeFromIndex()` function shall remove the calling object from the scene's index.

This function can be called in conjunction with the `addToIndex()` function to ensure the index bookkeeping is correct when the item's position, transformation or shape changes.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN13QInputContext14setFocusWidgetEP7QWidget**Name**

`QInputContext::setFocusWidget` — set the input focus for the current context

Synopsis

```
#include <QtGui/qinputcontext.h>
void QInputContext::setFocusWidget (QWidget * w);
```

Description

The `QInputContext::setFocusWidget()` shall set the widget that has an input focus for this input context to `w`.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

See Also

`focusWidget()`

_ZN13QTextDocument14appendUndoItemEP17QAbstractUndoItem**Name**

`QTextDocument::appendUndoItem` — append an item to the undo stack

Synopsis

```
#include <QtGui/qtextdocument.h>
void QTextDocument::appendUndoItem(QAbstractUndoItem * item);
```

Description

The `QTextDocument::appendUndoItem()` function shall append a custom undo *item* to the undo stack.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QAbstractItemView13doItemsLayoutEv**Name**

`QAbstractItemView::doItemsLayout` — lay out the items in the view

Synopsis

```
#include <QtGui/qabstractitemview.h>
virtual void QAbstractItemView::doItemsLayout(void);
```

Description

The `QAbstractItemView::doItemsLayout()` function shall lay out the items in the view.

Inheritors of the `QAbstractItemView` class are intended to overload this function. The default implementation just calls `updateGeometries()` function and update the viewport.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QAbstractItemView16updateEditorDataEv**Name**

`QAbstractItemView::updateEditorData` — update the data shown in the open editor widgets in the view

Synopsis

```
#include <QtGui/qabstractitemview.h>
virtual void QAbstractItemView::updateEditorData(void);
```

Description

The `QAbstractItemView::updateEditorData()` function shall update the data shown in the open editor widgets in the view.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QAbstractItemView22updateEditorGeometriesEv**Name**

`QAbstractItemView::updateEditorGeometries` — update geometry of open editor widgets in the view

Synopsis

```
#include <QtGui/qabstractitemview.h>
virtual void QAbstractItemView::updateEditorGeometries(void);
```

Description

The `QAbstractItemView::updateEditorGeometries()` function shall update the geometry of the open editor widgets in the view.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QAbstractItemView23verticalScrollbarActionEi**Name**

`QAbstractItemView::verticalScrollbarAction` — dummy hook

Synopsis

```
#include <QtGui/qabstractitemview.h>
virtual void QAbstractItemView::verticalScrollbarAction(int action);
```

Description

The `QAbstractItemView::verticalScrollbarAction()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QAbstractItemView25horizontalScrollbarActionEi

Name

`QAbstractItemView::horizontalScrollbarAction` — dummy hook

Synopsis

```
#include <QtGui/qabstractitemview.h>
virtual void QAbstractItemView::horizontalScrollbarAction(int action);
```

Description

The `QAbstractItemView::horizontalScrollbarAction()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QGraphicsLineItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsLineItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsLineItem::setExtension(Extension extension, const
QVariant & variant);
```

Description

The `QGraphicsLineItem::setExtension()` function does nothing. It is provided as a hook to avoid problems with adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QGraphicsPathItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsPathItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsPathItem::setExtension(Extension extension, const
QVariant & variant);
```

Description

The `QGraphicsPathItem::setExtension()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QGraphicsRectItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsRectItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsRectItem::setExtension(Extension extension, const
QVariant & variant);
```

Description

The `QGraphicsRectItem::setExtension()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN17QGraphicsTextItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsTextItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsTextItem::setExtension(Extension extension, const
QVariant & variant);
```

Description

The `QGraphicsTextItem::setExtension()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN18QDragResponseEventC1Eb, _ZN18QDragResponseEventC2Eb

Name

`QDragResponseEvent::QDragResponseEvent` — `QDragResponseEvent` constructor

Synopsis

```
#include <QtGui/qevent.h>
QDragResponseEvent::QDragResponseEvent(bool accepted);
```

Description

The `QDragResponseEvent::QDragResponseEvent()` function shall construct a drag response event containing the *accepted* value, indicating whether the drag and drop operation was accepted by the recipient.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN19QGraphicsPixmapItem12setExtensionEN13QGraphicsItem 9ExtensionERK8QVariant

Name

`QGraphicsPixmapItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsPixmapItem::setExtension(Extension extension,
const QVariant & variant);
```

Description

The `QGraphicsPixmapItem::setExtension()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN19QGraphicsSceneEvent9setWidgetEP7QWidget**Name**

`QGraphicsSceneEvent::setWidget` — Sets the widget related to the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneEvent::setWidget (QWidget *widget);
```

Description

The `QGraphicsSceneEvent::setWidget()` function shall set the widget related to the event to *source*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

**_ZN19QGraphicsSceneEventC1EN6QEvent4TypeE,
_ZN19QGraphicsSceneEventC2EN6QEvent4TypeE****Name**

`QGraphicsSceneEvent::QGraphicsSceneEvent` — `QGraphicsSceneEvent` constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneEvent::QGraphicsSceneEvent (QEvent::Type type);
```

Description

Constructs a new `QGraphicsSceneEvent` of the specified type *type*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

**_ZN19QToolBarChangeEventC1Eb,
_ZN19QToolBarChangeEventC2Eb****Name**

`QToolBarChangeEvent::QToolBarChangeEvent` — `QToolBarChangeEvent` constructor

Synopsis

```
#include <QtGui/qevent.h>
QToolBarChangeEvent::QToolBarChangeEvent (bool state);
```

Description

The `QToolBarChangeEvent::QToolBarChangeEvent()` function shall construct a `QToolBarChangeEvent` object given the current button state in `state`.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN20QGraphicsEllipseItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsEllipseItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsEllipseItem::setExtension (Extension extension,
const QVariant & variant);
```

Description

The `QGraphicsEllipseItem::setExtension()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN20QGraphicsPolygonItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsPolygonItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsPolygonItem::setExtension (Extension extension,
const QVariant & variant);
```

Description

The `QGraphicsPolygonItem::setExtension()` does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN23QGraphicsSceneHelpEvent11setScenePosERK7QPointF**Name**

`QGraphicsSceneHelpEvent::setScenePos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneHelpEvent::setScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneHelpEvent::setScenePos()` function shall set the position of the cursor associated with the event (i.e. the position where the event occurred) to the given point *pos* relative to the scene.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN23QGraphicsSceneHelpEvent12setScreenPosERK6QPoint**Name**

`QGraphicsSceneHelpEvent::setScreenPos` — set the event mouse position relative to the screen

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneHelpEvent::setScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneHelpEvent::setScreenPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position where the event occurred) to the given point *pos* relative to the screen.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN23QGraphicsSceneHelpEventC1EN6QEvent4TypeE, _ZN23QGraphicsSceneHelpEventC2EN6QEvent4TypeE

Name

`QGraphicsSceneHelpEvent::QGraphicsSceneHelpEvent` —
QGraphicsSceneHelpEvent constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneHelpEvent::QGraphicsSceneHelpEvent (QEvent::Type type);
```

Description

Constructs a new `QGraphicsSceneHelpEvent` of the specified type *type*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN23QGraphicsSimpleTextItem12setExtensionEN13QGraphicsItem9ExtensionERK8QVariant

Name

`QGraphicsSimpleTextItem::setExtension` — set extension for the item

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual void QGraphicsSimpleTextItem::setExtension(Extension extension,
const QVariant & variant);
```

Description

The `QGraphicsSimpleTextItem::setExtension()` function does nothing. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QAbstractPageSetupDialog7printerEv

Name

`QAbstractPageSetupDialog::printer` — get the printer that this page setup dialog is operating on

Synopsis

```
#include <QtGui/qabstractpagesetupdialog.h>
QPrinter * QAbstractPageSetupDialog::printer(void);
```

Description

The `QAbstractPageSetupDialog::printer()` function shall return the printer that this page setup dialog is operating on.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QAbstractPageSetupDialogC1EP8QPrinterP7QWidget, _ZN24QAbstractPageSetupDialogC2EP8QPrinterP7QWidget

Name

`QAbstractPageSetupDialog::QAbstractPageSetupDialog(QPrinter * printer, QWidget * parent)` — `QAbstractPageSetupDialog` constructor

Synopsis

```
#include <QtGui/qabstractpagesetupdialog.h>
QAbstractPageSetupDialog::QAbstractPageSetupDialog(QPrinter * printer,
QWidget * parent);
```

Description

The `QAbstractPageSetupDialog::QAbstractPageSetupDialog()` function shall construct the page setup dialog for the printer *printer* with *parent* as parent widget.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneHoverEvent11setScenePosERK7QPointF

Name

`QGraphicsSceneHoverEvent::setScenePos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneHoverEvent::setScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneHoverEvent::setScenePos()` function shall set the position of the cursor associated with the event (i.e. the position where the event occurred) to the given point *pos* relative to the scene.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneHoverEvent12setScreenPosERK6QPoint**Name**

`QGraphicsSceneHoverEvent::setScreenPos` — set the event mouse position relative to the screen

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneHoverEvent::setScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneHoverEvent::setScreenPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position where the event occurred) to the given point *pos* relative to the screen.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneHoverEvent6setPosERK7QPointF**Name**

`QGraphicsSceneHoverEvent::setPos` — set the event mouse position relative to the widget that generated the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneHoverEvent::setPos (const QPointF &pos);
```

Description

The `QGraphicsSceneHoverEvent::setPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos*. The position should be relative to the widget that generated the event.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneHoverEventC1EN6QEvent4TypeE, _ZN24QGraphicsSceneHoverEventC2EN6QEvent4TypeE

Name

`QGraphicsSceneHoverEvent::QGraphicsSceneHoverEvent` —
`QGraphicsSceneHoverEvent` constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneHoverEvent::QGraphicsSceneHoverEvent (QEvent::Type type);
```

Description

Constructs a new `QGraphicsSceneHoverEvent` of the specified type *type*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent10setButtonsE6QFlagsIN2Qt11MouseButtonEE

Name

`QGraphicsSceneMouseEvent::setButtons` — set the mouse buttons that were pressed when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setButtons (Qt::MouseButtonss buttons);
```

Description

The `QGraphicsSceneMouseEvent::setButtons()` function shall set the mouse buttons associated with the event (i.e. buttons that were pressed when the event was created) to *buttons*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent10setLastPosERK7QPointF**Name**

`QGraphicsSceneMouseEvent::setLastPos` — set the last recorded mouse cursor position in item coordinates

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setLastPos (const QPointF &pos);
```

Description

The `QGraphicsSceneMouseEvent::setLastPos()` function shall set the last recorded mouse cursor position associated with the event to the given point *pos* in item coordinates.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent11setScenePosERK7QPointF**Name**

`QGraphicsSceneMouseEvent::setScenePos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneMouseEvent::setScenePos()` function shall set the position of the cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the scene.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent12setModifiersE6QFlagsIN2Qt16KeyboardModifierEE

Name

`QGraphicsSceneMouseEvent::setModifiers` — set the keyboard modifiers that were active when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setModifiers (Qt::KeyboardModifiers
modifiers);
```

Description

The `QGraphicsSceneMouseEvent::setModifiers()` function shall set the keyboard modifiers associated with the event (i.e. modifiers that were active when the event was created) to *modifiers*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent12setScreenPosERK6QPoint

Name

`QGraphicsSceneMouseEvent::setScreenPos` — set the event mouse position relative to the screen

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneMouseEvent::setScreenPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the screen.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent15setLastScenePosERK7QPointF

Name

`QGraphicsSceneMouseEvent::setLastScenePos` — set the last recorded mouse cursor position in scene coordinates

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setLastScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneMouseEvent::setLastScenePos()` function shall set the last recorded mouse cursor position associated with the event to the given point *pos* in scene coordinates.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent16setButtonDownPosEN2Qt11MouseButtonERK7QPointF

Name

`QGraphicsSceneMouseEvent::setButtonDownPos` — set the mouse position where the specified button was clicked in item coordinates

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setButtonDownPos (Qt::MouseButton button,
const QPointF &pos);
```

Description

The `QGraphicsSceneMouseEvent::setButtonDownPos()` function shall set the mouse cursor position where the specified button *button* was clicked to the given point *pos* in item coordinates.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent16setLastScreenPosERK6QPoint

Name

`QGraphicsSceneMouseEvent::setLastScreenPos` — set the last recorded mouse cursor position in screen coordinates

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setLastScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneMouseEvent::setLastScreenPos()` function shall set the last recorded mouse cursor position associated with the event to the given point *pos* in screen coordinates.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent21setButtonDownScenePosEN2Qt11MouseButtonERK7QPointF

Name

`QGraphicsSceneMouseEvent::setButtonDownScenePos` — set the mouse position where the specified button was clicked in scene coordinates

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setButtonDownScenePos (Qt::MouseButton button, const QPointF &pos);
```

Description

The `QGraphicsSceneMouseEvent::setButtonDownScenePos()` function shall set the mouse cursor position where the specified button *button* was clicked to the given point *pos* in scene coordinates.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent22setButtonDownScreenPos EN2Qt11MouseButtonERK6QPoint

Name

`QGraphicsSceneMouseEvent::setButtonDownScreenPos` — set the mouse position where the specified button was clicked in screen coordinates

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setButtonDownScreenPos (Qt::MouseButton
button, const QPoint &pos);
```

Description

The `QGraphicsSceneMouseEvent::setButtonDownScreenPos()` function shall set the mouse cursor position where the specified button *button* was clicked to the given point *pos* in screen coordinates.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent6setPosERK7QPointF

Name

`QGraphicsSceneMouseEvent::setPos` — set the event mouse position relative to the widget that generated the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setPos (const QPointF &pos);
```

Description

The `QGraphicsSceneMouseEvent::setPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos*. The position should be relative to the widget that generated the event.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEvent9setButtonEN2Qt11MouseButtonE

Name

`QGraphicsSceneMouseEvent::setButton` — set the mouse button associated with event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneMouseEvent::setButton (Qt::MouseButton button);
```

Description

The `QGraphicsSceneMouseEvent::setButton()` function shall set the mouse button associated with the event (i.e. the button that caused the event) to *button*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneMouseEventC1EN6QEvent4TypeE, _ZN24QGraphicsSceneMouseEventC2EN6QEvent4TypeE

Name

`QGraphicsSceneMouseEvent::QGraphicsSceneMouseEvent` —
`QGraphicsSceneMouseEvent` constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneMouseEvent::QGraphicsSceneMouseEvent (QEvent::Type type);
```

Description

Constructs a new `QGraphicsSceneMouseEvent` of the specified type *type*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEvent10setButtonsE6QFlagsIN2Qt11MouseButtonEE

Name

`QGraphicsSceneWheelEvent::setButtons` — set the mouse buttons that were pressed when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneWheelEvent::setButtons (Qt::MouseButtonss buttons);
```

Description

The `QGraphicsSceneWheelEvent::setButtons()` function shall set the mouse buttons associated with the event (i.e. buttons that were pressed when the event was created) to *buttons*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEvent11setScenePosERK7QPointF

Name

`QGraphicsSceneWheelEvent::setScenePos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneWheelEvent::setScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneWheelEvent::setScenePos()` function shall set the position of the cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the scene.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEvent12setModifiersE6QFlagsIN2Qt16KeyboardModifierEE

Name

`QGraphicsSceneWheelEvent::setModifiers` — set the keyboard modifiers that were active when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneWheelEvent::setModifiers (Qt::KeyboardModifiers
modifiers);
```

Description

The `QGraphicsSceneWheelEvent::setModifiers()` function shall set the keyboard modifiers associated with the event (i.e. modifiers that were active when the event occurred) to *modifiers*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEvent12setScreenPosERK6QPoint

Name

`QGraphicsSceneWheelEvent::setScreenPos` — set the event mouse position relative to the screen

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneWheelEvent::setScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneWheelEvent::setScreenPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the screen.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEvent6setPosERK7QPointF**Name**

`QGraphicsSceneWheelEvent::setPos` — set the event mouse position relative to the widget that generated the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneWheelEvent::setPos (const QPointF &pos);
```

Description

The `QGraphicsSceneWheelEvent::setPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos*. The position should be relative to the widget that generated the event.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEvent8setDeltaEi**Name**

`QGraphicsSceneWheelEvent::setDelta` — set the distance that the wheel is rotated

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneWheelEvent::setDelta (int delta);
```

Description

The `QGraphicsSceneWheelEvent::setDelta()` function shall set the distance that the wheel is rotated to *delta*, in eighths (1/8s) of a degree.

A positive value indicates that the wheel was rotated forwards away from the user; a negative value indicates that the wheel was rotated backwards toward the user.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN24QGraphicsSceneWheelEventC1EN6QEvent4TypeE, _ZN24QGraphicsSceneWheelEventC2EN6QEvent4TypeE

Name

`QGraphicsSceneWheelEvent::QGraphicsSceneWheelEvent` —
`QGraphicsSceneWheelEvent` constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneWheelEvent::QGraphicsSceneWheelEvent (QEvent::Type type);
```

Description

Constructs a new `QGraphicsSceneWheelEvent` of the specified type *type* which should be always `QEvent::GraphicsSceneWheel`.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent10setButtonsE6QFlagsIN2Qt11MouseButtonEE

Name

`QGraphicsSceneDragDropEvent::setButtons` — set the mouse buttons that were pressed when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setButtons (Qt::MouseButtonss
buttons);
```

Description

The `QGraphicsSceneDragDropButtons()` function shall set the mouse buttons associated with the event (i.e. buttons that were pressed when the event was created) to *buttons*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent11setMimeDataEPK9QMimeData

Name

`QGraphicsSceneDragDropEvent::setMimeData` — set the MIME data for the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setMimeData (const QMimeData *data);
```

Description

The `QGraphicsSceneDragDropEvent::setMimeData()` function shall set the MIME data for the event *data*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent11setScenePosERK7QPointF

Name

`QGraphicsSceneDragDropEvent::setScenePos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneDragDropEvent::setScenePos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the scene.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent12setModifiersE6QFlagsIN2Qt16KeyboardModifierEE

Name

`QGraphicsSceneDragDropEvent::setModifiers` — set the keyboard modifiers that were active when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setModifiers (Qt::KeyboardModifiers
modifiers);
```

Description

The `QGraphicsSceneDragDropEvent::setModifiers()` function shall set the keyboard modifiers associated with the event (i.e. modifiers that were active when the event was created) to *modifiers*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent12setScreenPosERK6QPoint

Name

`QGraphicsSceneDragDropEvent::setScreenPos` — set the event mouse position relative to the screen

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneDragDropEvent::setScreenPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the screen.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent17setProposedActionEN2Qt10DropActionE

Name

`QGraphicsSceneDragDropEvent::setProposedAction` — set the proposed action for the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void    QGraphicsSceneDragDropEvent::setProposedAction    (Qt::DropAction
action);
```

Description

The `QGraphicsSceneDragDropEvent::setProposedAction()` function shall set the proposed action to *action*.

The *action* should be one of the possible actions set using the `QGraphicsSceneDragDropEvent::setPossibleActions()` function.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent18setPossibleActionsE6QFlagsIN2Qt10DropActionEE

Name

`QGraphicsSceneDragDropEvent::setPossibleActions` — set the possible drop actions that the drag can result in

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void    QGraphicsSceneDragDropEvent::setPossibleActions    (Qt::DropActions
actions);
```

Description

The `QGraphicsSceneDragDropEvent::setPossibleActions()` function shall set the possible drop actions that the drag can result in to *actions*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent6setPosERK7QPointF**Name**

`QGraphicsSceneDragDropEvent::setPos` — set the event mouse position relative to the widget that generated the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setPos (const QPointF &pos);
```

Description

The `QGraphicsSceneDragDropEvent::setPos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos*. The position should be relative to the widget that generated the event.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEvent9setSourceEP7QWidget**Name**

`QGraphicsSceneDragDropEvent::setSource` — set the source widget for the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneDragDropEvent::setSource (QWidget *source);
```

Description

The `QGraphicsSceneDragDropEvent::setSource()` function shall set the source widget (i.e. the widget that created the drop event) to *source*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN27QGraphicsSceneDragDropEventC1EN6QEvent4TypeE, _ZN27QGraphicsSceneDragDropEventC2EN6QEvent4TypeE

Name

`QGraphicsSceneDragDropEvent::QGraphicsSceneDragDropEvent` — `QGraphicsSceneDragDropEvent` constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneDragDropEvent::QGraphicsSceneDragDropEvent (QEvent::Type
type = None);
```

Description

Constructs a new `QGraphicsSceneDragDropEvent` of the specified type. The type should have one of the following values:

- `QEvent::GraphicsSceneDragEnter`
- `QEvent::GraphicsSceneDragLeave`
- `QEvent::GraphicsSceneDragMove`
- `QEvent::GraphicsSceneDrop`

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN2Qt12codecForHtmlERK10QByteArray

Name

`Qt::codecForHtml` — detect encoding of the provided snippet of html

Synopsis

```
#include <QtGui/qtextdocument.h>
QTextCodec *Qt::codecForHtml (const QByteArray &ba);
```

Description

The `Qt::codecForHtml()` function has the same behavior as the `QTextCodec::codecForHtml()` function described in QtCore 4.2.0.

_ZN30QGraphicsSceneContextMenuEvent11setScenePosERK7QPointF

Name

`QGraphicsSceneContextMenuEvent::setScenePos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneContextMenuEvent::setScenePos (const QPointF &pos);
```

Description

The `QGraphicsSceneContextMenuEvent::setScenePos()` function shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the scene.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN30QGraphicsSceneContextMenuEvent12setModifiersE6QFlagsIN2Qt16KeyboardModifierEE

Name

`QGraphicsSceneContextMenuEvent::setModifiers` — set the keyboard modifiers that were active when the event was created

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneContextMenuEvent::setModifiers
(Qt::KeyboardModifiers modifiers);
```

Description

The `QGraphicsSceneContextMenuEvent::setModifiers()` function shall set the keyboard modifiers associated with the event (i.e. modifiers that were active when the event occurred) to *modifiers*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN30QGraphicsSceneContextMenuEvent12setScreenPosERK6QPoint

Name

`QGraphicsSceneContextMenuEvent::setScreenPos` — set the event mouse position relative to the scene

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneContextMenuEvent::setScreenPos (const QPoint &pos);
```

Description

The `QGraphicsSceneContextMenuEvent::setScreenPos()` shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos* relative to the screen.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN30QGraphicsSceneContextMenuEvent6setPosERK7QPointF

Name

`QGraphicsSceneContextMenuEvent::setPos` — set the event mouse position relative to the widget that generated the event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneContextMenuEvent::setPos (const QPointF &pos);
```

Description

The `QGraphicsSceneContextMenuEvent::setPos()` shall set the position of the mouse cursor associated with the event (i.e. the position at the moment when the event occurred) to the given point *pos*. The position should be relative to the widget that generated the event.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN30QGraphicsSceneContextMenuEvent9setReasonENS_6ReasonE**Name**

`QGraphicsSceneContextMenuEvent::setReason` — set the reason for the context menu event

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
void QGraphicsSceneContextMenuEvent::setReason (Reason reason);
```

Description

The `QGraphicsSceneContextMenuEvent::setReason()` function shall set the reason for the context menu to *reason*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN30QGraphicsSceneContextMenuEventC1EN6QEvent4TypeE, _ZN30QGraphicsSceneContextMenuEventC2EN6QEvent4TypeE**Name**

`QGraphicsSceneContextMenuEvent::QGraphicsSceneContextMenuEvent` — `QGraphicsSceneContextMenuEvent` constructor

Synopsis

```
#include <QtGui/qgraphicssceneevent.h>
QGraphicsSceneContextMenuEvent::QGraphicsSceneContextMenuEvent
(QEvent::Type type = None);
```

Description

Constructs the graphics scene context menu event of the specified type.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN5QFont7cleanupEv**Name**

`QFont::cleanup` — cleanup the font system

Synopsis

```
#include <QtGui/qfont.h>
void QFont::cleanup(void);
```

Description

The `QFont::cleanup()` function shall cleanup the font system. This function should be called automatically when the application is finished.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN6QColor10invalidateEv

Name

`QColor::invalidate` — mark the color as invalid

Synopsis

```
#include <QtGui/qcolor.h>
void QColor::invalidate (void);
```

Description

The `QColor::invalidate()` function shall mark the color as invalid and set all components to zero. Alpha channel should be set to fully opaque (for compatibility with Qt3).

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN7QLayout11widgetEventEP6QEvent

Name

`QLayout::widgetEvent` — perform child widget layout when the parent widget is resized

Synopsis

```
#include <QtGui/qlayout.h>
void QLayout::widgetEvent (QEvent *e);
```

Description

The `QLayout::widgetEvent()` function shall perform child widget layout when the parent widget is resized. It should also handle removal of widgets.

The event `e` should have one of the following values:

- `QEvent::Resize`
- `QEvent::ChildRemoved`
- `QEvent::LayoutRequest`

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN7QPixmap19x11SetDefaultScreenEi**Name**

`QPixmap::x11SetDefaultScreen` — sets the default screen of the `QPixmap` object

Synopsis

```
#include <QtGui/qpixmap.h>
static int QPixmap::x11SetDefaultScreen (int screen);
```

Description

The `QPixmap::x11SetDefaultScreen()` function shall set the internal 'defaultScreen' property of the `QPixmap` object to *screen*.

The function shall return the previous defaultScreen value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN7QWidget11createWinIdEv**Name**

`QWidget::createWinId` — ensure that the widget is known to the windowing system

Synopsis

```
#include <QtGui/qwidget.h>
void QWidget::createWinId(void);
```

Description

The `QWidget::createWinId()` function shall ensure that the widget has a window system identifier (that is, known to the windowing system) and create such an identifier if it doesn't exist.

Value of the internal identifier can be checked by the inline `internalWinId()` member of the `QWidget` class.

_ZN7QWidget19overrideWindowStateE6QFlagsIN2Qt11WindowStateEE**Name**

`QWidget::overrideWindowState` — set the window state on child widgets

Synopsis

```
#include <QtGui/qwidget.h>
void QWidget::overrideWindowState(Qt::WindowStates state);
```

Description

The `QWidget::overrideWindowState()` function shall set the window state on child widgets similar to the `setWindowState()`.

The difference is that the window state changed event has the `isOverride()` flag set.

This function is provided mainly to keep old applications that use `Q3Workspace` working.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN8QMenuBar15setCornerWidgetEP7QWidgetN2Qt6CornerE

Name

`QMenuBar::setCornerWidget` — set the position of the widget

Synopsis

```
#include <QtGui/qmenubar.h>
void QMenuBar::setCornerWidget(QWidget * w, Qt::Corner corner =
Qt::TopRightCorner);
```

Description

The `QMenuBar::setCornerWidget()` function shall set widget `w` to be shown directly on the left of the first or the right of the last menu item, depending on `corner`.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN9QListView13doItemsLayoutEv

Name

`QListView::doItemsLayout` — layout items

Synopsis

```
#include <QtGui/qlistview.h>
void QListView::doItemsLayout(void);
```

Description

The `QListView::doItemsLayout()` function shall layout the items according to the flow and wrapping properties.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZN9QListView14resizeContentsEii**Name**

`QListView::resizeContents` — resize the internal contents of the view

Synopsis

```
#include <QtGui/qlistview.h>
void QListView::resizeContents(int width, int height);
```

Description

The `QListView::resizeContents()` function shall resize the internal contents to *width* and *height* and set the scroll bar ranges accordingly.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK11QProxyModel13setProxyModelERK11QModelIndex**Name**

`QProxyModel::setProxyModel` — change the model pointer to point to the proxy model

Synopsis

```
#include <QtGui/qproxymodel.h>
QModelIndex QProxyModel::setProxyModel(const QModelIndex &
source_index);
```

Description

The `QProxyModel::setProxyModel()` function shall change the model pointer in the given *source_index* to point to the proxy model.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK11QProxyModel14connectToModelEPK18QAbstractItemModel**Name**

`QProxyModel::connectToModel` — connect to all the signals emitted by the model

Synopsis

```
#include <QtGui/qproxymodel.h>
void QProxyModel::connectToModel(const QAbstractItemModel * model);
```

Description

The `QProxyModel::connectToModel()` function shall connect to all the signals emitted by given *model*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK11QProxyModel14setSourceModelERK11QModelIndex

Name

`QProxyModel::setSourceModel` — INSERT PURPOSE HERE

Synopsis

```
#include <QtGui/qproxymodel.h>
QModelIndex QProxyModel::setSourceModel(const QModelIndex &
proxy_index);
```

Description

The `QProxyModel::setSourceModel()` function shall change the model pointer in the given *proxy_index* to point to the source model.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK11QProxyModel19disconnectFromModelEPK18QAbstractItemModel

Name

`QProxyModel::disconnectFromModel` — disconnect from all signals emitted by the model

Synopsis

```
#include <QtGui/qproxymodel.h>
void QProxyModel::disconnectFromModel(const QAbstractItemModel *
model);
```

Description

The `QProxyModel::disconnectFromModel()` function shall disconnect from all the signals emitted by the given *model*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK12QPaintEngine10systemClipEv**Name**

QPaintEngine::systemClip — get the system clip

Synopsis

```
#include <QtGui/qpaintengine.h>
QRegion QPaintEngine::systemClip (void);
```

Description

The `QPaintEngine::systemClip()` function shall return the system clip. The system clip is read only while the painter is active. An empty region indicates that system clip is not in use.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK12QPaintEngine10systemRectEv**Name**

QPaintEngine::systemRect — retrieve the rect for drawing within the backing store

Synopsis

```
#include <QtGui/qpaintengine.h>
QRect QPaintEngine::systemRect (void);
```

Description

The `QPaintEngine::systemRect()` function shall retrieve the rectangle for drawing within the backing store.

This function should only be used by the backing store, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK12QPaintEngine16coordinateOffsetEv**Name**

`QPaintEngine::coordinateOffset` — get the offset from the painters origo to the engines origo

Synopsis

```
#include <QtGui/qpaintengine.h>
virtual QPoint QPaintEngine::coordinateOffset (void);
```

Description

The `QPaintEngine::coordinateOffset()` function shall return the offset from the painters origo to the engines origo. This value is used by `QPainter` for engines who have internal double buffering.

This function only makes sense when the engine is active.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK13QGraphicsItem17supportsExtensionENS_9ExtensionE**Name**

`QGraphicsItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsItem::supportsExtension(Extension extension);
```

Description

The `QGraphicsItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions.

This function does nothing and should always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK13QGraphicsItem9extensionERK8QVariant**Name**

`QGraphicsItem::extension` — auxiliary virtual function

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsItem::extension (const QVariant &variant);
```

Description

The `QGraphicsItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

_ZNK13QInputContext11focusWidgetEv**Name**

`QInputContext::focusWidget` — get the widget that has an input focus for this input context

Synopsis

```
#include <QtGui/qinputcontext.h>
QWidget * QInputContext::focusWidget (void);
```

Description

The `QInputContext::focusWidget()` function shall return the widget that has an input focus for this input context.

The return value may differ from `holderWidget()` if the input context is shared between several text widgets.

Ordinary input methods should not call this function directly to keep platform independence and flexible configuration possibility.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

See Also

`setFocusWidget()`, `holderWidget()`

_ZNK13QItemDelegate10decorationERK20QStyleOptionViewItemRK8QVariant**Name**

`QItemDelegate::decoration` — get the pixmap used to decorate the root of the item view

Synopsis

```
#include <QtGui/qitemdelegate.h>
QPixmap QItemDelegate::decoration(const QStyleOptionViewItem & option,
const QVariant & variant);
```

Description

The `QItemDelegate::decoration()` function shall return the pixmap used to decorate the root of the item view.

The style *option* controls the appearance of the root; the *variant* refers to the data associated with an item.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK13QItemDelegate8selectedERK7QPixmapRK8QPaletteb

Name

`QItemDelegate::selected` — get the selected version of the given pixmap using the given palette

Synopsis

```
#include <QtGui/qitemdelegate.h>
QPixmap * QItemDelegate::selected(const QPixmap & pixmap, const
QPalette & palette, bool enabled);
```

Description

The `QItemDelegate::selected()` function shall return the selected version of the given *pixmap* using the given *palette*.

The *enabled* argument decides whether the normal or disabled highlight color of the palette is used.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK13QTextDocument7frameAtEi

Name

`QTextDocument::frameAt` — get the frame that contains the given cursor position

Synopsis


```
#include <QtGui/qtextdocument.h>
QTextFrame * QTextDocument::frameAt(int pos);
```

Description

The `QTextDocument::frameAt()` function shall return the frame that contains the text cursor position `pos`.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK14QTextTableCell12lastPositionEv

Name

`QTextTableCell::lastPosition` — get the last position in the cell's document

Synopsis

```
#include <QtGui/qtexttable.h>
int QTextTableCell::lastPosition(void);
```

Description

The `QTextTableCell::lastPosition()` function shall return the last valid position in the document occupied by this cell.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK14QTextTableCell13firstPositionEv

Name

`QTextTableCell::firstPosition` — get the first position in the cell's document

Synopsis

```
#include <QtGui/qtexttable.h>
int QTextTableCell::firstPosition(void);
```

Description

The `QTextTableCell::firstPosition()` function shall return the first valid position in the document occupied by this cell.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsLineItem17supportsExtensionEN13QGraphicsLineItem9ExtensionE**Name**

`QGraphicsLineItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsLineItem::supportsExtension(Extension extension);
```

Description

The `QGraphicsLineItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This function shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsLineItem9extensionERK8QVariant**Name**

`QGraphicsLineItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsLineItem::extension(const QVariant &variant);
```

Description

The `QGraphicsLineItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsPathItem17supportsExtensionEN13QGraphicsPathItem9ExtensionE**Name**

`QGraphicsPathItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsPathItem::supportsExtension(Extension extension);
```

Description

The `QGraphicsPathItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This functions shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsPathItem9extensionERK8QVariant

Name

`QGraphicsPathItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsPathItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsPathItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsRectItem17supportsExtensionEN13QGraphicsItem9ExtensionE

Name

`QGraphicsRectItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsRectItem::supportsExtension(Extension extension);
```

Description

The `QGraphicsRectItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This functions shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsRectItem9extensionERK8QVariant**Name**

`QGraphicsRectItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsRectItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsRectItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsTextItem17supportsExtensionEN13QGraphicsItem9ExtensionE**Name**

`QGraphicsTextItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsTextItem::supportsExtension(Extension extension);
```

Description

The `QGraphicsTextItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This function shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK17QGraphicsTextItem9extensionERK8QVariant**Name**

`QGraphicsTextItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsTextItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsTextItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK19QGraphicsPixmapItem17supportsExtensionEN13QGraphicsItem9ExtensionE

Name

`QGraphicsPixmapItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsPixmapItem::supportsExtension(Extension
extension);
```

Description

The `QGraphicsPixmapItem::supportsExtension()` is provided as a hook to avoid problems related to adding virtual functions. This function shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK19QGraphicsPixmapItem9extensionERK8QVariant

Name

`QGraphicsPixmapItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsPixmapItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsPixmapItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK20QGraphicsEllipseItem17supportsExtensionEN13QGraphicsItem9ExtensionE**Name**

`QGraphicsEllipseItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsEllipseItem::supportsExtension(Extension
extension);
```

Description

The `QGraphicsEllipseItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This function shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK20QGraphicsEllipseItem9extensionERK8QVariant**Name**

`QGraphicsEllipseItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsEllipseItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsEllipseItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK20QGraphicsPolygonItem17supportsExtensionEN13QGraphicsItem9ExtensionE**Name**

`QGraphicsPolygonItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsPolygonItem::supportsExtension(Extension
extension);
```

Description

The `QGraphicsPolygonItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This function shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK20QGraphicsPolygonItem9extensionERK8QVariant

Name

`QGraphicsPolygonItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsPolygonItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsPolygonItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK23QGraphicsSimpleTextItem17supportsExtensionEN13QGraphicsItem9ExtensionE

Name

`QGraphicsSimpleTextItem::supportsExtension` — check if the item supports the extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual bool QGraphicsSimpleTextItem::supportsExtension(Extension
extension);
```

Description

The `QGraphicsSimpleTextItem::supportsExtension()` function is provided as a hook to avoid problems related to adding virtual functions. This function shall always return false.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK23QGraphicsSimpleTextItem9extensionERK8QVariant**Name**

`QGraphicsSimpleTextItem::extension` — get the item's extension

Synopsis

```
#include <QtGui/qgraphicsitem.h>
virtual QVariant QGraphicsSimpleTextItem::extension(const QVariant &
variant);
```

Description

The `QGraphicsSimpleTextItem::extension()` function shall simply return its argument. It is provided as a hook to avoid problems related to adding virtual functions.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK6QImage7devTypeEv**Name**

`QImage::devType` — get identifier designating that the object is a `QImage`

Synopsis

```
#include <QtGui/qimage.h>
int QImage::devType(void);
```

Description

The `QImage::devType()` shall always return the `QInternal::Image` value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK7QLayout13totalSizeHintEv**Name**

`QLayout::totalSizeHint` — get the preferred size for the layout

Synopsis

```
#include <QtGui/qlayout.h>
QSize QLayout::totalSizeHint (void);
```

Description

The `QLayout::totalSizeHint()` function shall return the preferred size of the layout. The result shall be the same as for the `sizeHint()`, but with `contentsMargins` and menu bar taken into account.

Note that all inheritors of the `QLayoutItem` class should provide their own implementation of the `sizeHint()` function.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK7QLayout16totalMaximumSizeEv**Name**

`QLayout::totalMaximumSize` — get the maximum size of the layout

Synopsis

```
#include <QtGui/qlayout.h>
QSize QLayout::totalMaximumSize (void);
```

Description

The `QLayout::totalMaximumSize()` function has the same behavior as the `QLayout::maximumSize()` function described in QtGui 4.2.0, but also takes `contentsMargins` and menu bar into account.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK7QLayout16totalMinimumSizeEv**Name**

`QLayout::totalMinimumSize` — get the minimum size of the layout

Synopsis

```
#include <QtGui/qlayout.h>
QSize QLayout::totalMinimumSize (void);
```

Description

The `QLayout::totalMinimumSize()` function has the same behavior as the `QLayout::minimumSize()` function described in QtGui 4.2.0, but also takes `contentsMargins` and `menu bar` into account.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK7QLayout19totalHeightForWidthEi**Name**

`QLayout::totalHeightForWidth` — get the preferred height for the layout

Synopsis

```
#include <QtGui/qlayout.h>
int QLayout::totalHeightForWidth (int w);
```

Description

The `QLayout::totalHeightForWidth()` function has the same behavior as the `QLayoutItem::heightForWidth()` function described in QtGui 4.2.0, but also takes `contentsMargins` and `menu bar` into account (`heightForWidth()` is inherited by `QLayout` from `QLayoutItem`).

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK7QPixmap7devTypeEv**Name**

`QPixmap::devType` — get identifier designating that the object is a QPixmap

Synopsis

```
#include <QtGui/qpixmap.h>
int QPixmap::devType(void);
```

Description

The `QPixmap::devType()` function shall always return the `QInternal::Pixmap` value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK7QWidget7devTypeEv

Name

`QWidget::devType` — get identifier designating that the object is a `QWidget`

Synopsis

```
#include <QtGui/qwidget.h>
int QWidget::devType(void);
```

Description

The `QWidget::devType()` function shall always return the `QInternal::Widget` value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK8QMenuBar12cornerWidgetEN2Qt6CornerE

Name

`QMenuBar::cornerWidget` — get the widget in the left of the first or the right of the last menu

Synopsis

```
#include <QtGui/qmenubar.h>
QWidget * QMenuBar::cornerWidget(Qt::Corner corner =
Qt::TopRightCorner);
```

Description

The `QMenuBar::cornerWidget()` function shall return the widget in the left of the first or the right of the last menu item, depending on *corner*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK8QMenuBar14actionGeometryEP7QAction**Name**

`QMenuBar::actionGeometry` — get geometry of an action

Synopsis

```
#include <QtGui/qmenubar.h>
QRect QMenuBar::actionGeometry(QAction * act);
```

Description

The `QMenuBar::actionGeometry()` function shall return the geometry of action *act*.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK8QMenuBar8actionAtERK6QPoint**Name**

`QMenuBar::actionAt` — get the item at the given point

Synopsis

```
#include <QtGui/qmenubar.h>
QAction * QMenuBar::actionAt(const QPoint & pt);
```

Description

The `QMenuBar::actionAt()` function shall return the item at the given point *pt*.

If there is no item there or if it is a separator item then the function shall return 0.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

ZNK8QPainter3mapEiiPiS0**Name**

`QPainter::map` — apply the painter's transformation on the given point

Synopsis

```
#include <QtGui/qpainter.h>
void QPainter::map (int x, int y, int *rx, int *ry);
```

Description

The `QPainter::map()` function shall set (*rx* and *ry*) to the point that results from applying the painter's current transformation on the point (*x*, *y*).

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK8QPicture7devTypeEv**Name**

`QPicture::devType` — get identifier designating that the object is a `QPicture`

Synopsis

```
#include <QtGui/qpicture.h>
int QPicture::devType(void);
```

Description

The `QPicture::devType()` function shall always return the `QInternal::Picture` value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK8QPrinter7devTypeEv**Name**

`QPrinter::devType` — get identifier designating that the object is a `QPrinter`

Synopsis

```
#include <QtGui/qprinter.h>
int QPrinter::devType(void);
```

Description

The `QPrinter::devType()` function shall always return the `QInternal::Printer` value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK8QToolBar14actionGeometryEP7QAction**Name**

`QToolBar::actionGeometry` — get geometry of the toolbar item associated with the given action

Synopsis

```
#include <QtGui/qtoolbar.h>
QRect QToolBar::actionGeometry(QAction * action);
```

Description

The `QToolBar::actionGeometry()` function shall return the geometry of the toolbar item associated with the given *action*.

If no matching item is found, then the function shall return an invalid `QRect`.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK9QFontInfo8overlineEv

Name

`QFontInfo::overline` — get the overline value of the matched window system font

Synopsis

```
#include <QtGui/qfontinfo.h>
bool QFontInfo::overline (void);
```

Description

The `QFontInfo::overline()` function shall return the overline value of the matched window system font.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK9QFontInfo9strikeOutEv

Name

`QFontInfo::strikeOut` — get the underline value of the matched window system font

Synopsis

```
#include <QtGui/qfontinfo.h>
bool QFontInfo::strikeOut (void);
```

Description

The `QFontInfo::strikeOut()` function shall return the strikeout value of the matched window system font.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK9QFontInfo9underlineEv**Name**

`QFontInfo::underline` — get the underline value of the matched window system font

Synopsis

```
#include <QtGui/qfontinfo.h>
bool QFontInfo::underline (void);
```

Description

The `QFontInfo::underline()` function shall return the underline value of the matched window system font.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

18.8 Interfaces for libQtXml

Table 18-521 defines the library name and shared object name for the libQtXml library

Table 18-521 libQtXml Definition

Library:	libQtXml
SONAME:	libQtXml.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI
[QtXml] QtXml 4.2.0

18.8.1 Qt4 XML**18.8.1.1 Class data for QXmlAttributes**

The virtual table for the `QXmlAttributes` class is described by Table 18-522

Table 18-522 Primary vtable for QXmlAttributes

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for <code>QXmlAttributes</code>
<code>vfunc[0]:</code>	NULL or <code>QXmlAttributes::~~QXmlAttributes()</code>
<code>vfunc[1]:</code>	NULL or <code>QXmlAttributes::~~QXmlAttributes()</code>

The Run Time Type Information for the `QXmlAttributes` class is described by Table 18-523

Table 18-523 typeinfo for QXmlAttributes

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlAttributes

18.8.1.2 Class data for QXmlInputSource

The virtual table for the QXmlInputSource class is described by Table 18-524

Table 18-524 Primary vtable for QXmlInputSource

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlInputSource
vfunc[0]:	QXmlInputSource::~~QXmlInputSource()
vfunc[1]:	QXmlInputSource::~~QXmlInputSource()
vfunc[2]:	QXmlInputSource::setData(QString const&)
vfunc[3]:	QXmlInputSource::setData(QByteArray const&)
vfunc[4]:	QXmlInputSource::fetchData()
vfunc[5]:	QXmlInputSource::data() const
vfunc[6]:	QXmlInputSource::next()
vfunc[7]:	QXmlInputSource::reset()
vfunc[8]:	QXmlInputSource::fromRawData(QByteArray const&, bool)

The Run Time Type Information for the QXmlInputSource class is described by Table 18-525

Table 18-525 typeinfo for QXmlInputSource

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlInputSource

18.8.1.3 Class data for QXmlReader

The virtual table for the QXmlReader class is described by Table 18-526

Table 18-526 Primary vtable for QXmlReader

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlReader

vfunc[0]:	NULL or QXmlReader::~QXmlReader()
vfunc[1]:	NULL or QXmlReader::~QXmlReader()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual
vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual
vfunc[8]:	__cxa_pure_virtual
vfunc[9]:	__cxa_pure_virtual
vfunc[10]:	__cxa_pure_virtual
vfunc[11]:	__cxa_pure_virtual
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	__cxa_pure_virtual
vfunc[19]:	__cxa_pure_virtual
vfunc[20]:	__cxa_pure_virtual
vfunc[21]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlReader class is described by Table 18-527

Table 18-527 typeinfo for QXmlReader

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlReader

18.8.1.4 Class data for QXmlSimpleReader

The virtual table for the QXmlSimpleReader class is described by Table 18-528

Table 18-528 Primary vtable for QXmlSimpleReader

Base Offset	0
Virtual Base Offset	0

RTTI	typeid for QXmlSimpleReader
vfunc[0]:	QXmlSimpleReader::~~QXmlSimpleReader()
vfunc[1]:	QXmlSimpleReader::~~QXmlSimpleReader()
vfunc[2]:	QXmlSimpleReader::feature(QString const&, bool*) const
vfunc[3]:	QXmlSimpleReader::setFeature(QString const&, bool)
vfunc[4]:	QXmlSimpleReader::hasFeature(QString const&) const
vfunc[5]:	QXmlSimpleReader::property(QString const&, bool*) const
vfunc[6]:	QXmlSimpleReader::setProperty(QString const&, void*)
vfunc[7]:	QXmlSimpleReader::hasProperty(QString const&) const
vfunc[8]:	QXmlSimpleReader::setEntityResolver(QXmlEntityResolver*)
vfunc[9]:	QXmlSimpleReader::entityResolver() const
vfunc[10]:	QXmlSimpleReader::setDTDHandler(QXmlDTDHandler*)
vfunc[11]:	QXmlSimpleReader::DTDHandler() const
vfunc[12]:	QXmlSimpleReader::setContentHandler(QXmlContentHandler*)
vfunc[13]:	QXmlSimpleReader::contentHandler() const
vfunc[14]:	QXmlSimpleReader::setErrorHandler(QXmlErrorHandler*)
vfunc[15]:	QXmlSimpleReader::errorHandler() const
vfunc[16]:	QXmlSimpleReader::setLexicalHandler(QXmlLexicalHandler*)
vfunc[17]:	QXmlSimpleReader::lexicalHandler() const
vfunc[18]:	QXmlSimpleReader::setDeclHandler(QXmlDeclHandler*)
vfunc[19]:	QXmlSimpleReader::declHandler() const

vfunc[20]:	QXmlSimpleReader::parse(QXmlInputSource const&)
vfunc[21]:	QXmlSimpleReader::parse(QXmlInputSource const*)
vfunc[22]:	QXmlSimpleReader::parse(QXmlInputSource const*, bool)
vfunc[23]:	QXmlSimpleReader::parseContinue()

The Run Time Type Information for the QXmlSimpleReader class is described by Table 18-529

Table 18-529 typeinfo for QXmlSimpleReader

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QXmlSimpleReader
basetype:	typeinfo for QXmlReader

18.8.1.5 Class data for QXmlLocator

The virtual table for the QXmlLocator class is described by Table 18-530

Table 18-530 Primary vtable for QXmlLocator

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlLocator
vfunc[0]:	QXmlLocator::~~QXmlLocator()
vfunc[1]:	QXmlLocator::~~QXmlLocator()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlLocator class is described by Table 18-531

Table 18-531 typeinfo for QXmlLocator

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlLocator

18.8.1.6 Class data for QXmlContentHandler

The virtual table for the QXmlContentHandler class is described by Table 18-532

Table 18-532 Primary vtable for QXmlContentHandler

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QXmlContentHandler
vfunc[0]:	NULL or QXmlContentHandler::~~QXmlContentHandler()
vfunc[1]:	NULL or QXmlContentHandler::~~QXmlContentHandler()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual
vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual
vfunc[8]:	__cxa_pure_virtual
vfunc[9]:	__cxa_pure_virtual
vfunc[10]:	__cxa_pure_virtual
vfunc[11]:	__cxa_pure_virtual
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlContentHandler class is described by Table 18-533

Table 18-533 typeinfo for QXmlContentHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlContentHandler

18.8.1.7 Class data for QXmlErrorHandler

The virtual table for the QXmlErrorHandler class is described by Table 18-534

Table 18-534 Primary vtable for QXmlErrorHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlErrorHandler
vfunc[0]:	NULL or QXmlErrorHandler::~~QXmlErrorHandler()

vfunc[1]:	NULL or QXmlErrorHandler::~~QXmlErrorHandler()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlErrorHandler class is described by Table 18-535

Table 18-535 typeinfo for QXmlErrorHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlErrorHandler

18.8.1.8 Class data for QXmlDTDHandler

The virtual table for the QXmlDTDHandler class is described by Table 18-536

Table 18-536 Primary vtable for QXmlDTDHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlDTDHandler
vfunc[0]:	NULL or QXmlDTDHandler::~~QXmlDTDHandler()
vfunc[1]:	NULL or QXmlDTDHandler::~~QXmlDTDHandler()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlDTDHandler class is described by Table 18-537

Table 18-537 typeinfo for QXmlDTDHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlDTDHandler

18.8.1.9 Class data for QXmlEntityResolver

The virtual table for the QXmlEntityResolver class is described by Table 18-538

Table 18-538 Primary vtable for QXmlEntityResolver

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlEntityResolver
vfunc[0]:	NULL or QXmlEntityResolver::~~QXmlEntityR esolver()
vfunc[1]:	NULL or QXmlEntityResolver::~~QXmlEntityR esolver()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlEntityResolver class is described by Table 18-539

Table 18-539 typeinfo for QXmlEntityResolver

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlEntityResolver

18.8.1.10 Class data for QXmlLexicalHandler

The virtual table for the QXmlLexicalHandler class is described by Table 18-540

Table 18-540 Primary vtable for QXmlLexicalHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlLexicalHandler
vfunc[0]:	NULL or QXmlLexicalHandler::~~QXmlLexical Handler()
vfunc[1]:	NULL or QXmlLexicalHandler::~~QXmlLexical Handler()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual

vfunc[6]:	__cxa_pure_virtual
vfunc[7]:	__cxa_pure_virtual
vfunc[8]:	__cxa_pure_virtual
vfunc[9]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlLexicalHandler class is described by Table 18-541

Table 18-541 typeinfo for QXmlLexicalHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlLexicalHandler

18.8.1.11 Class data for QXmlDeclHandler

The virtual table for the QXmlDeclHandler class is described by Table 18-542

Table 18-542 Primary vtable for QXmlDeclHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlDeclHandler
vfunc[0]:	NULL or QXmlDeclHandler::~~QXmlDeclHandler()
vfunc[1]:	NULL or QXmlDeclHandler::~~QXmlDeclHandler()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	__cxa_pure_virtual

The Run Time Type Information for the QXmlDeclHandler class is described by Table 18-543

Table 18-543 typeinfo for QXmlDeclHandler

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QXmlDeclHandler

18.8.1.12 Class data for QXmlDefaultHandler

The virtual table for the QXmlDefaultHandler class is described by Table 18-544

Table 18-544 Primary vtable for QXmlDefaultHandler

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QXmlDefaultHandler
vfunc[0]:	NULL or QXmlDefaultHandler::~~QXmlDefaultHandler()
vfunc[1]:	NULL or QXmlDefaultHandler::~~QXmlDefaultHandler()
vfunc[2]:	QXmlDefaultHandler::setDocumentLocator(QXmlLocator*)
vfunc[3]:	QXmlDefaultHandler::startDocument()
vfunc[4]:	QXmlDefaultHandler::endDocument()
vfunc[5]:	QXmlDefaultHandler::startPrefixMapping(QString const&, QString const&)
vfunc[6]:	QXmlDefaultHandler::endPrefixMapping(QString const&)
vfunc[7]:	QXmlDefaultHandler::startElement(QString const&, QString const&, QString const&, QXmlAttributes const&)
vfunc[8]:	QXmlDefaultHandler::endElement(QString const&, QString const&, QString const&)
vfunc[9]:	QXmlDefaultHandler::characters(QString const&)
vfunc[10]:	QXmlDefaultHandler::ignoreableWhitespace(QString const&)
vfunc[11]:	QXmlDefaultHandler::processingInstruction(QString const&, QString const&)
vfunc[12]:	QXmlDefaultHandler::skippedEntity(QString const&)
vfunc[13]:	QXmlDefaultHandler::errorString() const
vfunc[14]:	QXmlDefaultHandler::warning(QXmlParseException const&)
vfunc[15]:	QXmlDefaultHandler::error(QXmlParseException const&)

vfunc[16]:	QXmlDefaultHandler::fatalError(QXmlParseException const&)
vfunc[17]:	QXmlDefaultHandler::notationDecl(QString const&, QString const&, QString const&)
vfunc[18]:	QXmlDefaultHandler::unparsedEntityDecl(QString const&, QString const&, QString const&, QString const&)
vfunc[19]:	QXmlDefaultHandler::resolveEntity(QString const&, QString const&, QXmlInputSource*&)
vfunc[20]:	QXmlDefaultHandler::startDTD(QString const&, QString const&, QString const&)
vfunc[21]:	QXmlDefaultHandler::endDTD()
vfunc[22]:	QXmlDefaultHandler::startEntity(QString const&)
vfunc[23]:	QXmlDefaultHandler::endEntity(QString const&)
vfunc[24]:	QXmlDefaultHandler::startCDATA()
vfunc[25]:	QXmlDefaultHandler::endCDATA()
vfunc[26]:	QXmlDefaultHandler::comment(QString const&)
vfunc[27]:	QXmlDefaultHandler::attributeDecl(QString const&, QString const&, QString const&, QString const&, QString const&)
vfunc[28]:	QXmlDefaultHandler::internalEntityDecl(QString const&, QString const&)
vfunc[29]:	QXmlDefaultHandler::externalEntityDecl(QString const&, QString const&, QString const&)

The Run Time Type Information for the QXmlDefaultHandler class is described by Table 18-545

Table 18-545 typeinfo for QXmlDefaultHandler

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	
Name	typeinfo name for QXmlDefaultHandler	

flags:	0	2
basetype:	typeinfo for QXmlContentHandler	
basetype:	typeinfo for QXmlErrorHandler	1026
basetype:	typeinfo for QXmlErrorHandler	2050
basetype:	typeinfo for QXmlErrorHandler	1026
basetype:	typeinfo for QXmlErrorHandler	2050
basetype:	typeinfo for QXmlErrorHandler	1026

18.8.1.13 Interfaces for Qt4 XML

An LSB conforming implementation shall provide the generic functions for Qt4 XML specified in Table 18-546, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-546 libQtXml - Qt4 XML Function Interfaces

ZN10QDomEntityC1ERKS [QtXml]	_ZN10QDomEntityC1Ev [QtXml]
ZN10QDomEntityC2ERKS [QtXml]	_ZN10QDomEntityC2Ev [QtXml]
ZN10QDomEntityaSERKS [QtXml]	_ZN11QDomCommentC1ERKS_ [QtXml]
_ZN11QDomCommentC1Ev [QtXml]	_ZN11QDomCommentC2ERKS_ [QtXml]
_ZN11QDomCommentC2Ev [QtXml]	_ZN11QDomCommentaSERKS_ [QtXml]
_ZN11QDomElement10setTagName ERK7QString [QtXml]	_ZN11QDomElement12setAttributeE RK7QStringS2_ [QtXml]
_ZN11QDomElement12setAttributeE RK7QStringd [QtXml]	_ZN11QDomElement12setAttributeE RK7QStringf [QtXml]
_ZN11QDomElement12setAttributeE RK7QStringx [QtXml]	_ZN11QDomElement12setAttributeE RK7QStringy [QtXml]
_ZN11QDomElement13attributeNode eERK7QString [QtXml]	_ZN11QDomElement14setAttribute NSE7QStringRKS0_S2_ [QtXml]
_ZN11QDomElement14setAttribute NSE7QStringRKS0_d [QtXml]	_ZN11QDomElement14setAttribute NSE7QStringRKS0_x [QtXml]
_ZN11QDomElement14setAttribute NSE7QStringRKS0_y [QtXml]	_ZN11QDomElement15attributeNode eNSE7QStringS2_ [QtXml]

_ZN11QDomElement15removeAttributeERK7QString [QtXml]	_ZN11QDomElement16setAttributeNodeERK8QDomAttr [QtXml]
ZN11QDomElement17removeAttributeNSERK7QStringS2 [QtXml]	_ZN11QDomElement18setAttributeNodeNSERK8QDomAttr [QtXml]
_ZN11QDomElement19removeAttributeNodeERK8QDomAttr [QtXml]	_ZN11QDomElementC1ERKS_ [QtXml]
_ZN11QDomElementC1Ev [QtXml]	_ZN11QDomElementC2ERKS_ [QtXml]
_ZN11QDomElementC2Ev [QtXml]	_ZN11QDomElementaSERKS_ [QtXml]
_ZN11QXmlLocatorC1Ev [QtXml]	_ZN11QXmlLocatorC2Ev [QtXml]
_ZN11QXmlLocatorD0Ev [QtXml]	_ZN11QXmlLocatorD1Ev [QtXml]
_ZN11QXmlLocatorD2Ev [QtXml]	_ZN12QDomDocument10importNodeERK8QDomNodeb [QtXml]
ZN12QDomDocument10setContentEP15QXmlInputSourceP10QXmlReaderP7QStringPiS6 [QtXml]	_ZN12QDomDocument10setContentEP9QIODeviceP7QStringPiS4_ [QtXml]
ZN12QDomDocument10setContentEP9QIODevicebP7QStringPiS4 [QtXml]	_ZN12QDomDocument10setContentERK10QByteArrayP7QStringPiS5_ [QtXml]
ZN12QDomDocument10setContentERK10QByteArraybP7QStringPiS5 [QtXml]	_ZN12QDomDocument10setContentERK7QStringPS0_PiS4_ [QtXml]
_ZN12QDomDocument10setContentERK7QStringbPS0_PiS4_ [QtXml]	_ZN12QDomDocument11elementByIdERK7QString [QtXml]
_ZN12QDomDocument13createElementERK7QString [QtXml]	_ZN12QDomDocument13createElementERK7QString [QtXml]
_ZN12QDomDocument14createTextNodeERK7QString [QtXml]	_ZN12QDomDocument15createAttributeERK7QString [QtXml]
ZN12QDomDocument15createElementNSERK7QStringS2 [QtXml]	_ZN12QDomDocument17createElementNSERK7QStringS2_ [QtXml]
_ZN12QDomDocument18createCDATASectionERK7QString [QtXml]	_ZN12QDomDocument19elementsByTagNameNSERK7QStringS2_ [QtXml]
_ZN12QDomDocument21createEntityReferenceERK7QString [QtXml]	_ZN12QDomDocument22createDocumentFragmentEv [QtXml]
ZN12QDomDocument27createProcessingInstructionERK7QStringS2 [QtXml]	_ZN12QDomDocumentC1ERK16QDomDocumentType [QtXml]
_ZN12QDomDocumentC1ERK7QString [QtXml]	_ZN12QDomDocumentC1ERKS_ [QtXml]
_ZN12QDomDocumentC1Ev [QtXml]	_ZN12QDomDocumentC2ERK16QDomDocumentType [QtXml]

_ZN12QDomDocumentC2ERK7QString [QtXml]	_ZN12QDomDocumentC2ERKS_ [QtXml]
_ZN12QDomDocumentC2Ev [QtXml]	_ZN12QDomDocumentD1Ev [QtXml]
_ZN12QDomDocumentD2Ev [QtXml]	_ZN12QDomDocumentaSERKS_ [QtXml]
ZN12QDomNodeListC1ERKS [QtXml]	_ZN12QDomNodeListC1Ev [QtXml]
ZN12QDomNodeListC2ERKS [QtXml]	_ZN12QDomNodeListC2Ev [QtXml]
_ZN12QDomNodeListD1Ev [QtXml]	_ZN12QDomNodeListD2Ev [QtXml]
ZN12QDomNodeListaSERKS [QtXml]	_ZN12QDomNotationC1ERKS_ [QtXml]
_ZN12QDomNotationC1Ev [QtXml]	_ZN12QDomNotationC2ERKS_ [QtXml]
_ZN12QDomNotationC2Ev [QtXml]	_ZN12QDomNotationaSERKS_ [QtXml]
_ZN14QXmlAttributes5clearEv [QtXml]	_ZN14QXmlAttributes6appendERK7QStringS2_S2_S2_ [QtXml]
_ZN15QXmlInputSource11fromRawDataERK10QByteArrayb [QtXml]	_ZN15QXmlInputSource4nextEv [QtXml]
_ZN15QXmlInputSource5resetEv [QtXml]	_ZN15QXmlInputSource7setDataERK10QByteArray [QtXml]
_ZN15QXmlInputSource7setDataERK7QString [QtXml]	_ZN15QXmlInputSource9fetchDataEv [QtXml]
_ZN15QXmlInputSourceC1EP9QIODevice [QtXml]	_ZN15QXmlInputSourceC1ER11QTextStream [QtXml]
_ZN15QXmlInputSourceC1ER5QFile [QtXml]	_ZN15QXmlInputSourceC1Ev [QtXml]
_ZN15QXmlInputSourceC2EP9QIODevice [QtXml]	_ZN15QXmlInputSourceC2ER11QTextStream [QtXml]
_ZN15QXmlInputSourceC2ER5QFile [QtXml]	_ZN15QXmlInputSourceC2Ev [QtXml]
_ZN15QXmlInputSourceD0Ev [QtXml]	_ZN15QXmlInputSourceD1Ev [QtXml]
_ZN15QXmlInputSourceD2Ev [QtXml]	_ZN16QDomCDATASectionC1ERKS_ [QtXml]
_ZN16QDomCDATASectionC1Ev [QtXml]	_ZN16QDomCDATASectionC2ERKS_ [QtXml]
_ZN16QDomCDATASectionC2Ev [QtXml]	_ZN16QDomCDATASectionaSERKS_ [QtXml]

ZN16QDomDocumentTypeC1ERKS [QtXml]	_ZN16QDomDocumentTypeC1Ev [QtXml]
ZN16QDomDocumentTypeC2ERKS [QtXml]	_ZN16QDomDocumentTypeC2Ev [QtXml]
ZN16QDomDocumentTypeaSERKS [QtXml]	_ZN16QDomNamedNodeMap12setNamedItemERK8QDomNode [QtXml]
_ZN16QDomNamedNodeMap14setNamedItemNSERK8QDomNode [QtXml]	_ZN16QDomNamedNodeMap15removeNamedItemERK7QString [QtXml]
ZN16QDomNamedNodeMap17removeNamedItemNSERK7QStringS2 [QtXml]	_ZN16QDomNamedNodeMapC1ERKS_ [QtXml]
_ZN16QDomNamedNodeMapC1Ev [QtXml]	_ZN16QDomNamedNodeMapC2ERKS_ [QtXml]
_ZN16QDomNamedNodeMapC2Ev [QtXml]	_ZN16QDomNamedNodeMapD1Ev [QtXml]
_ZN16QDomNamedNodeMapD2Ev [QtXml]	_ZN16QDomNamedNodeMapaSERKS_ [QtXml]
_ZN16QXmlSimpleReader10setFeatureERK7QStringb [QtXml]	_ZN16QXmlSimpleReader11setPropertyERK7QStringPv [QtXml]
_ZN16QXmlSimpleReader13parseContinueEv [QtXml]	_ZN16QXmlSimpleReader13setDTDHandlerEP14QXmlDTDHandler [QtXml]
_ZN16QXmlSimpleReader14setDeclHandlerEP15QXmlDeclHandler [QtXml]	_ZN16QXmlSimpleReader15setErrorHandlerEP16QXmlErrorHandler [QtXml]
_ZN16QXmlSimpleReader17setContentHandlerEP18QXmlContentHandler [QtXml]	_ZN16QXmlSimpleReader17setEntityResolverEP18QXmlEntityResolver [QtXml]
_ZN16QXmlSimpleReader17setLexicalHandlerEP18QXmlLexicalHandler [QtXml]	_ZN16QXmlSimpleReader5parseEPK15QXmlInputSource [QtXml]
_ZN16QXmlSimpleReader5parseEPK15QXmlInputSourceb [QtXml]	_ZN16QXmlSimpleReader5parseERK15QXmlInputSource [QtXml]
_ZN16QXmlSimpleReaderC1Ev [QtXml]	_ZN16QXmlSimpleReaderC2Ev [QtXml]
_ZN16QXmlSimpleReaderD0Ev [QtXml]	_ZN16QXmlSimpleReaderD1Ev [QtXml]
_ZN16QXmlSimpleReaderD2Ev [QtXml]	_ZN17QDomCharacterData10appendDataERK7QString [QtXml]
_ZN17QDomCharacterData10deleteDataEmm [QtXml]	_ZN17QDomCharacterData10insertDataEmRK7QString [QtXml]

_ZN17QDomCharacterData11replaceDataEmmRK7QString [QtXml]	_ZN17QDomCharacterData13substringDataEmm [QtXml]
_ZN17QDomCharacterData7setDataERK7QString [QtXml]	_ZN17QDomCharacterDataC1ERKS_ [QtXml]
_ZN17QDomCharacterDataC1Ev [QtXml]	_ZN17QDomCharacterDataC2ERKS_ [QtXml]
_ZN17QDomCharacterDataC2Ev [QtXml]	_ZN17QDomCharacterDataaSERKS_ [QtXml]
_ZN18QDomImplementation14createDocumentERK7QStringS2_RK16QDomDocumentType [QtXml]	_ZN18QDomImplementation17invalidDataPolicyEv [QtXml]
_ZN18QDomImplementation18createDocumentTypeERK7QStringS2_S2_ [QtXml]	_ZN18QDomImplementation20setInvalidDataPolicyENS_17InvalidDataPolicyE [QtXml]
_ZN18QDomImplementation6isNullEv [QtXml]	_ZN18QDomImplementationC1ERKS_ [QtXml]
_ZN18QDomImplementationC1Ev [QtXml]	_ZN18QDomImplementationC2ERKS_ [QtXml]
_ZN18QDomImplementationC2Ev [QtXml]	_ZN18QDomImplementationD1Ev [QtXml]
_ZN18QDomImplementationD2Ev [QtXml]	_ZN18QDomImplementationaSERKS_ [QtXml]
_ZN18QXmlDefaultHandler10charactersERK7QString [QtXml]	_ZN18QXmlDefaultHandler10endElementERK7QStringS2_S2_ [QtXml]
_ZN18QXmlDefaultHandler10fatalErrorERK18QXmlParseException [QtXml]	_ZN18QXmlDefaultHandler10startCDATAEv [QtXml]
_ZN18QXmlDefaultHandler11endDocumentEv [QtXml]	_ZN18QXmlDefaultHandler11startEntityERK7QString [QtXml]
_ZN18QXmlDefaultHandler12notationDeclERK7QStringS2_S2_ [QtXml]	_ZN18QXmlDefaultHandler12startElementERK7QStringS2_S2_RK14QXmlAttributes [QtXml]
_ZN18QXmlDefaultHandler13attributeDeclERK7QStringS2_S2_S2_S2_ [QtXml]	_ZN18QXmlDefaultHandler13resolveEntityERK7QStringS2_RP15QXmlInputSource [QtXml]
_ZN18QXmlDefaultHandler13skippeDEntityERK7QString [QtXml]	_ZN18QXmlDefaultHandler13startDocumentEv [QtXml]
_ZN18QXmlDefaultHandler16endPrefixMappingERK7QString [QtXml]	_ZN18QXmlDefaultHandler18externalEntityDeclERK7QStringS2_S2_ [QtXml]
ZN18QXmlDefaultHandler18internalEntityDeclERK7QStringS2 [QtXml]	_ZN18QXmlDefaultHandler18setDocumentLocatorEP11QXmlLocator [QtXml]

ZN18QXmlDefaultHandler18startPrefixMappingERK7QStringS2[QtXml]	_ZN18QXmlDefaultHandler18unparsedEntityDeclERK7QStringS2_S2_S2_[QtXml]
_ZN18QXmlDefaultHandler19ignorableWhitespaceERK7QString [QtXml]	_ZN18QXmlDefaultHandler21processingInstructionERK7QStringS2_[QtXml]
_ZN18QXmlDefaultHandler5errorERK18QXmlParseException [QtXml]	_ZN18QXmlDefaultHandler6endDTDDev [QtXml]
_ZN18QXmlDefaultHandler7commentERK7QString [QtXml]	_ZN18QXmlDefaultHandler7warningERK18QXmlParseException [QtXml]
_ZN18QXmlDefaultHandler8endCDATAEv [QtXml]	_ZN18QXmlDefaultHandler8startDTDERK7QStringS2_S2_ [QtXml]
_ZN18QXmlDefaultHandler9endEntityERK7QString [QtXml]	_ZN18QXmlParseExceptionC1ERK7QStringiS2_S2_ [QtXml]
_ZN18QXmlParseExceptionC2ERK7QStringiS2_S2_ [QtXml]	_ZN18QXmlParseExceptionD1Ev [QtXml]
_ZN18QXmlParseExceptionD2Ev [QtXml]	_ZN19QDomEntityReferenceC1ERKS_ [QtXml]
_ZN19QDomEntityReferenceC1Ev [QtXml]	_ZN19QDomEntityReferenceC2ERKS_ [QtXml]
_ZN19QDomEntityReferenceC2Ev [QtXml]	_ZN19QDomEntityReferenceC3ERKS_ [QtXml]
ZN20QDomDocumentFragmentC1ERKS [QtXml]	_ZN20QDomDocumentFragmentC1Ev [QtXml]
ZN20QDomDocumentFragmentC2ERKS [QtXml]	_ZN20QDomDocumentFragmentC2Ev [QtXml]
ZN20QDomDocumentFragmentC3ERKS [QtXml]	_ZN20QXmlNamespaceSupport10popContextEv [QtXml]
_ZN20QXmlNamespaceSupport11pushContextEv [QtXml]	_ZN20QXmlNamespaceSupport5resetEv [QtXml]
ZN20QXmlNamespaceSupport9setPrefixERK7QStringS2 [QtXml]	_ZN20QXmlNamespaceSupportC1Ev [QtXml]
_ZN20QXmlNamespaceSupportC2Ev [QtXml]	_ZN20QXmlNamespaceSupportD1Ev [QtXml]
_ZN20QXmlNamespaceSupportD2Ev [QtXml]	_ZN25QDomProcessingInstruction7setDataERK7QString [QtXml]
ZN25QDomProcessingInstructionC1ERKS [QtXml]	_ZN25QDomProcessingInstructionC1Ev [QtXml]
ZN25QDomProcessingInstructionC2ERKS [QtXml]	_ZN25QDomProcessingInstructionC2Ev [QtXml]
ZN25QDomProcessingInstructionC3ERKS [QtXml]	_ZN28QDomAttribute8setValueERK7QString [QtXml]

ZN8QDomAttrC1ERKS [QtXml]	_ZN8QDomAttrC1Ev [QtXml]
ZN8QDomAttrC2ERKS [QtXml]	_ZN8QDomAttrC2Ev [QtXml]
ZN8QDomAttrASERKS [QtXml]	_ZN8QDomNode11appendChildERKS_ [QtXml]
_ZN8QDomNode11insertAfterERKS_S1_ [QtXml]	_ZN8QDomNode11removeChildERKS_ [QtXml]
_ZN8QDomNode12insertBeforeERKS_S1_ [QtXml]	_ZN8QDomNode12replaceChildERKS_S1_ [QtXml]
_ZN8QDomNode12setNodeValueERK7QString [QtXml]	_ZN8QDomNode5clearEv [QtXml]
_ZN8QDomNode9normalizeEv [QtXml]	_ZN8QDomNode9setPrefixERK7QString [QtXml]
ZN8QDomNodeC1ERKS [QtXml]	_ZN8QDomNodeC1Ev [QtXml]
ZN8QDomNodeC2ERKS [QtXml]	_ZN8QDomNodeC2Ev [QtXml]
_ZN8QDomNodeD1Ev [QtXml]	_ZN8QDomNodeD2Ev [QtXml]
ZN8QDomNodeaSERKS [QtXml]	_ZN8QDomText9splitTextEi [QtXml]
ZN8QDomTextC1ERKS [QtXml]	_ZN8QDomTextC1Ev [QtXml]
ZN8QDomTextC2ERKS [QtXml]	_ZN8QDomTextC2Ev [QtXml]
ZN8QDomTextaSERKS [QtXml]	_ZNK10QDomEntity12notationNameEv [QtXml]
_ZNK10QDomEntity8publicIdEv [QtXml]	_ZNK10QDomEntity8systemIdEv [QtXml]
_ZNK11QDomElement10attributesEv [QtXml]	_ZNK11QDomElement11attributeNSE7QStringRKS0_S2_ [QtXml]
_ZNK11QDomElement12hasAttributeERK7QString [QtXml]	_ZNK11QDomElement14hasAttributeNSERK7QStringS2_ [QtXml]
_ZNK11QDomElement17elementsByTagNameERK7QString [QtXml]	_ZNK11QDomElement19elementsByTagNameNSERK7QStringS2_ [QtXml]
_ZNK11QDomElement4textEv [QtXml]	_ZNK11QDomElement7tagNameEv [QtXml]
ZNK11QDomElement9attributeERK7QStringS2 [QtXml]	_ZNK12QDomDocument11toByteArrayEi [QtXml]
_ZNK12QDomDocument14implementationEv [QtXml]	_ZNK12QDomDocument15documentElementEv [QtXml]
_ZNK12QDomDocument17elementsByTagNameERK7QString [QtXml]	_ZNK12QDomDocument7doctypeEv [QtXml]
_ZNK12QDomDocument8toStringEi [QtXml]	_ZNK12QDomNodeList4itemEi [QtXml]
_ZNK12QDomNodeList6lengthEv [QtXml]	_ZNK12QDomNodeListeqERKS_ [QtXml]

Znk12QDomNodeListneERKS [QtXml]	_Znk12QDomNotation8publicIdEv [QtXml]
_Znk12QDomNotation8systemIdEv [QtXml]	_Znk14QXmlAttributes3uriEi [QtXml]
_Znk14QXmlAttributes4typeERK7Q String [QtXml]	_Znk14QXmlAttributes4typeERK7Q StringS2_ [QtXml]
_Znk14QXmlAttributes4typeEi [QtXml]	_Znk14QXmlAttributes5indexERK7 QString [QtXml]
Znk14QXmlAttributes5indexERK7 QStringS2 [QtXml]	_Znk14QXmlAttributes5qNameEi [QtXml]
_Znk14QXmlAttributes5valueERK7 QString [QtXml]	_Znk14QXmlAttributes5valueERK7 QStringS2_ [QtXml]
_Znk14QXmlAttributes5valueEi [QtXml]	_Znk14QXmlAttributes6lengthEv [QtXml]
_Znk14QXmlAttributes9localName Ei [QtXml]	_Znk15QXmlInputSource4dataEv [QtXml]
_Znk16QDomDocumentType14inte rnalSubsetEv [QtXml]	_Znk16QDomDocumentType4name Ev [QtXml]
_Znk16QDomDocumentType8entiti esEv [QtXml]	_Znk16QDomDocumentType8publi cIdEv [QtXml]
_Znk16QDomDocumentType8syste mIdEv [QtXml]	_Znk16QDomDocumentType9notat ionsEv [QtXml]
Znk16QDomNamedNodeMap11na medItemNSERK7QStringS2 [QtXml]	_Znk16QDomNamedNodeMap4ite mEi [QtXml]
_Znk16QDomNamedNodeMap6len gthEv [QtXml]	_Znk16QDomNamedNodeMap8con tainsERK7QString [QtXml]
_Znk16QDomNamedNodeMap9na medItemERK7QString [QtXml]	_Znk16QDomNamedNodeMapeqE RKS_ [QtXml]
Znk16QDomNamedNodeMapneE RKS [QtXml]	_Znk16QXmlSimpleReader10DTD HandlerEv [QtXml]
_Znk16QXmlSimpleReader10hasFea tureERK7QString [QtXml]	_Znk16QXmlSimpleReader11decl HandlerEv [QtXml]
_Znk16QXmlSimpleReader11hasPro pertyERK7QString [QtXml]	_Znk16QXmlSimpleReader12error HandlerEv [QtXml]
_Znk16QXmlSimpleReader14conten tHandlerEv [QtXml]	_Znk16QXmlSimpleReader14entity ResolverEv [QtXml]
_Znk16QXmlSimpleReader14lexical HandlerEv [QtXml]	_Znk16QXmlSimpleReader7feature ERK7QStringPb [QtXml]
_Znk16QXmlSimpleReader8propert yERK7QStringPb [QtXml]	_Znk17QDomCharacterData4dataE v [QtXml]

_ZNK17QDomCharacterData6lengthEv [QtXml]	_ZNK17QDomCharacterData8nodeTypeEv [QtXml]
ZNK18QDomImplementation10hasFeatureERK7QStringS2 [QtXml]	_ZNK18QDomImplementationeqERKS_ [QtXml]
ZNK18QDomImplementationneERKS [QtXml]	_ZNK18QXmlDefaultHandler11errorStringEv [QtXml]
_ZNK18QXmlParseException10lineNumberEv [QtXml]	_ZNK18QXmlParseException12columnNumberEv [QtXml]
_ZNK18QXmlParseException7messageEv [QtXml]	_ZNK18QXmlParseException8publicIdEv [QtXml]
_ZNK18QXmlParseException8systemIdEv [QtXml]	_ZNK20QXmlNamespaceSupport11processNameERK7QStringbRS0_S3_ [QtXml]
_ZNK20QXmlNamespaceSupport3uriERK7QString [QtXml]	_ZNK20QXmlNamespaceSupport6prefixERK7QString [QtXml]
_ZNK20QXmlNamespaceSupport8prefixesERK7QString [QtXml]	_ZNK20QXmlNamespaceSupport8prefixesEv [QtXml]
_ZNK20QXmlNamespaceSupport9splitNameERK7QStringRS0_S3_ [QtXml]	_ZNK25QDomProcessingInstruction4dataEv [QtXml]
_ZNK25QDomProcessingInstruction6targetEv [QtXml]	_ZNK8QDomAttr12ownerElementEv [QtXml]
_ZNK8QDomAttr4nameEv [QtXml]	_ZNK8QDomAttr5valueEv [QtXml]
_ZNK8QDomAttr9specifiedEv [QtXml]	_ZNK8QDomNode10attributesEv [QtXml]
_ZNK8QDomNode10childNodesEv [QtXml]	_ZNK8QDomNode10firstChildEv [QtXml]
_ZNK8QDomNode10isDocumentEv [QtXml]	_ZNK8QDomNode10isNotationEv [QtXml]
_ZNK8QDomNode10lineNumberEv [QtXml]	_ZNK8QDomNode10parentNodeEv [QtXml]
_ZNK8QDomNode10toDocumentEv [QtXml]	_ZNK8QDomNode10toNotationEv [QtXml]
ZNK8QDomNode11isSupportedERK7QStringS2 [QtXml]	_ZNK8QDomNode11nextSiblingEv [QtXml]
_ZNK8QDomNode12columnNumberEv [QtXml]	_ZNK8QDomNode12namespaceURI [QtXml]
_ZNK8QDomNode13hasAttributesEv [QtXml]	_ZNK8QDomNode13hasChildNodesEv [QtXml]
_ZNK8QDomNode13ownerDocumentEv [QtXml]	_ZNK8QDomNode14isCDATASectionEv [QtXml]

_ZNK8QDomNode14isDocumentTypeEv [QtXml]	_ZNK8QDomNode14toCDATASectionEv [QtXml]
_ZNK8QDomNode14toDocumentTypeEv [QtXml]	_ZNK8QDomNode15isCharacterDataEv [QtXml]
_ZNK8QDomNode15previousSiblingEv [QtXml]	_ZNK8QDomNode15toCharacterDataEv [QtXml]
_ZNK8QDomNode16lastChildElementERK7QString [QtXml]	_ZNK8QDomNode17firstChildElementERK7QString [QtXml]
_ZNK8QDomNode17isEntityReferenceEv [QtXml]	_ZNK8QDomNode17toEntityReferenceEv [QtXml]
_ZNK8QDomNode18isDocumentFragmentEv [QtXml]	_ZNK8QDomNode18nextSiblingElementERK7QString [QtXml]
_ZNK8QDomNode18toDocumentFragmentEv [QtXml]	_ZNK8QDomNode22previousSiblingElementERK7QString [QtXml]
_ZNK8QDomNode23isProcessingInstructionEv [QtXml]	_ZNK8QDomNode23toProcessingInstructionEv [QtXml]
_ZNK8QDomNode4saveER11QTextStreami [QtXml]	_ZNK8QDomNode6isAttrEv [QtXml]
_ZNK8QDomNode6isNullEv [QtXml]	_ZNK8QDomNode6isTextEv [QtXml]
_ZNK8QDomNode6prefixEv [QtXml]	_ZNK8QDomNode6toAttrEv [QtXml]
_ZNK8QDomNode6toTextEv [QtXml]	_ZNK8QDomNode8isEntityEv [QtXml]
_ZNK8QDomNode8nodeNameEv [QtXml]	_ZNK8QDomNode8nodeTypeEv [QtXml]
_ZNK8QDomNode8toEntityEv [QtXml]	_ZNK8QDomNode9cloneNodeEb [QtXml]
_ZNK8QDomNode9isCommentEv [QtXml]	_ZNK8QDomNode9isElementEv [QtXml]
_ZNK8QDomNode9lastChildEv [QtXml]	_ZNK8QDomNode9localNameEv [QtXml]
_ZNK8QDomNode9namedItemERK7QString [QtXml]	_ZNK8QDomNode9nodeValueEv [QtXml]
_ZNK8QDomNode9toCommentEv [QtXml]	_ZNK8QDomNode9toElementEv [QtXml]
ZNK8QDomNodeeqERKS [QtXml]	_ZNK8QDomNodeneERKS_ [QtXml]
_ZlsR11QTextStreamRK8QDomNode [QtXml]	

An LSB conforming implementation shall provide the generic data interfaces for Qt4 XML specified in Table 18-547, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-547 libQtXml - Qt4 XML Data Interfaces

_ZN15QXmlInputSource13EndOfDocumentE [QtXml]	_ZN15QXmlInputSource9EndOfDataE [QtXml]
_ZTI10QXmlReader [CXXABI-1.86]	_ZTI11QXmlLocator [CXXABI-1.86]
_ZTI14QXmlAttributes [CXXABI-1.86]	_ZTI14QXmlDTDHandler [CXXABI-1.86]
_ZTI15QXmlDeclHandler [CXXABI-1.86]	_ZTI15QXmlInputSource [CXXABI-1.86]
_ZTI16QXmlErrorHandler [CXXABI-1.86]	_ZTI16QXmlSimpleReader [CXXABI-1.86]
_ZTI18QXmlContentHandler [CXXABI-1.86]	_ZTI18QXmlDefaultHandler [CXXABI-1.86]
_ZTI18QXmlEntityResolver [CXXABI-1.86]	_ZTI18QXmlLexicalHandler [CXXABI-1.86]
_ZTV10QXmlReader [CXXABI-1.86]	_ZTV11QXmlLocator [CXXABI-1.86]
_ZTV14QXmlAttributes [CXXABI-1.86]	_ZTV14QXmlDTDHandler [CXXABI-1.86]
_ZTV15QXmlDeclHandler [CXXABI-1.86]	_ZTV15QXmlInputSource [CXXABI-1.86]
_ZTV16QXmlErrorHandler [CXXABI-1.86]	_ZTV16QXmlSimpleReader [CXXABI-1.86]
_ZTV18QXmlContentHandler [CXXABI-1.86]	_ZTV18QXmlDefaultHandler [CXXABI-1.86]
_ZTV18QXmlEntityResolver [CXXABI-1.86]	_ZTV18QXmlLexicalHandler [CXXABI-1.86]

18.9 Data Definitions for libQtXml

This section defines global identifiers and their values that are associated with interfaces contained in libQtXml. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

18.9.1 QtXml/qdom.h

```
class QDomImplementation;
```

```

enum QDomImplementation::InvalidDataPolicy {
    AcceptInvalidChars = 0,
    DropInvalidChars = 1,
    ReturnNullNode = 2
};
class QDomNode;
enum QDomNode::NodeType {
    ElementNode = 1,
    AttributeNode = 2,
    TextNode = 3,
    CDATASectionNode = 4,
    EntityReferenceNode = 5,
    EntityNode = 6,
    ProcessingInstructionNode = 7,
    CommentNode = 8,
    DocumentNode = 9,
    DocumentTypeNode = 10,
    DocumentFragmentNode = 11,
    NotationNode = 12,
    BaseNode = 21,
    CharacterDataNode = 22
};
class QDomNodeList;
class QDomDocumentType;
class QDomDocument;
class QDomNamedNodeMap;
class QDomDocumentFragment;
class QDomCharacterData;
class QDomAttr;
class QDomElement;
class QDomText;
class QDomComment;
class QDomCDATASection;
class QDomNotation;
class QDomEntity;
class QDomEntityReference;
class QDomProcessingInstruction;

```

18.9.2 QtXml/qxml.h

```

typedef enum QtValidLicenseForXmlModule QtXmlModule;
class QXmlNamespaceSupport;
class QXmlAttributes;
struct QXmlAttributes::Attribute;
class QXmlInputSource;
class QXmlParseException;
class QXmlReader;
class QXmlSimpleReader;
class QXmlLocator;
class QXmlContentHandler;
class QXmlErrorHandler;
class QXmlDTDHandler;
class QXmlEntityResolver;
class QXmlLexicalHandler;
class QXmlDeclHandler;
class QXmlDefaultHandler;

```

18.10 Interfaces for libQtOpenGL

Table 18-548 defines the library name and shared object name for the libQtOpenGL library

Table 18-548 libQtOpenGL Definition

Library:	libQtOpenGL
SONAME:	libQtOpenGL.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI

[LSB] This Specification

[QtOpenGL] QtOpenGL 4.2.0

[QtXml] QtXml 4.2.0

18.10.1 Qt4 OpenGL

18.10.1.1 Class data for QGLContext

The virtual table for the QGLContext class is described by Table 18-549

Table 18-549 Primary vtable for QGLContext

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGLContext
vfunc[0]:	QGLContext::~~QGLContext()
vfunc[1]:	QGLContext::~~QGLContext()
vfunc[2]:	QGLContext::create(QGLContext const*)
vfunc[3]:	QGLContext::makeCurrent()
vfunc[4]:	QGLContext::doneCurrent()
vfunc[5]:	QGLContext::swapBuffers() const
vfunc[6]:	QGLContext::chooseContext(QGLContext const*)
vfunc[7]:	QGLContext::tryVisual(QGLFormat const&, int)
vfunc[8]:	QGLContext::chooseVisual()

The Run Time Type Information for the QGLContext class is described by Table 18-550

Table 18-550 typeinfo for QGLContext

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QGLContext

18.10.1.2 Class data for QGLWidget

The virtual table for the QGLWidget class is described by Table 18-551

Table 18-551 Primary vtable for QGLWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGLWidget
vfunc[0]:	QGLWidget::metaObject() const
vfunc[1]:	QGLWidget::qt_metacast(char const*)
vfunc[2]:	QGLWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QGLWidget::~~QGLWidget()
vfunc[4]:	QGLWidget::~~QGLWidget()
vfunc[5]:	QGLWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QGLWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)
vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)

vfunc[24]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QGLWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QGLWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const
vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)

vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()
vfunc[54]:	QGLWidget::updateGL()
vfunc[55]:	QGLWidget::updateOverlayGL()
vfunc[56]:	QGLWidget::initializeGL()
vfunc[57]:	QGLWidget::resizeGL(int, int)
vfunc[58]:	QGLWidget::paintGL()
vfunc[59]:	QGLWidget::initializeOverlayGL()
vfunc[60]:	QGLWidget::resizeOverlayGL(int, int)
vfunc[61]:	QGLWidget::paintOverlayGL()
vfunc[62]:	QGLWidget::glInit()
vfunc[63]:	QGLWidget::glDraw()

The Run Time Type Information for the QGLWidget class is described by Table 18-552

Table 18-552 typeinfo for QGLWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QGLWidget
basetype:	typeinfo for QWidget

18.10.1.3 Class data for QGLPixelBuffer

The virtual table for the QGLPixelBuffer class is described by Table 18-553

Table 18-553 Primary vtable for QGLPixelBuffer

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGLPixelBuffer
vfunc[0]:	QGLPixelBuffer::~~QGLPixelBuffer()
vfunc[1]:	QGLPixelBuffer::~~QGLPixelBuffer()
vfunc[2]:	NULL or QGLPixelBuffer::devType() const

vfunc[3]:	QGLPixelBuffer::paintEngine() const
vfunc[4]:	QGLPixelBuffer::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QGLPixelBuffer class is described by Table 18-554

Table 18-554 typeinfo for QGLPixelBuffer

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QGLPixelBuffer
basetype:	typeinfo for QPaintDevice

18.10.1.4 Class data for QGLFramebufferObject

The virtual table for the QGLFramebufferObject class is described by Table 18-555

Table 18-555 Primary vtable for QGLFramebufferObject

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QGLFramebufferObject
vfunc[0]:	QGLFramebufferObject::~~QGLFramebufferObject()
vfunc[1]:	QGLFramebufferObject::~~QGLFramebufferObject()
vfunc[2]:	NULL or QGLFramebufferObject::devType() const
vfunc[3]:	QGLFramebufferObject::paintEngine() const
vfunc[4]:	QGLFramebufferObject::metric(QPaintDevice::PaintDeviceMetric) const

The Run Time Type Information for the QGLFramebufferObject class is described by Table 18-556

Table 18-556 typeinfo for QGLFramebufferObject

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QGLFramebufferObject
basetype:	typeinfo for QPaintDevice

18.10.1.5 Interfaces for Qt4 OpenGL

An LSB conforming implementation shall provide the generic functions for Qt4 OpenGL specified in Table 18-557, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-557 libQtOpenGL - Qt4 OpenGL Function Interfaces

_ZN10QGLContext11bindTextureERK6QImageji [QtOpenGL]	_ZN10QGLContext11bindTextureERK7QPixmapji [QtOpenGL]
_ZN10QGLContext11bindTextureERK7QString [QtOpenGL]	_ZN10QGLContext11doneCurrentEv [QtOpenGL]
_ZN10QGLContext11makeCurrentEv [QtOpenGL]	_ZN10QGLContext12chooseVisualEv [QtOpenGL]
ZN10QGLContext13chooseContextEPKS [QtOpenGL]	_ZN10QGLContext13deleteTextureEj [QtOpenGL]
_ZN10QGLContext14currentContextEv [QtOpenGL]	_ZN10QGLContext14setInitializedEb [QtOpenGL]
_ZN10QGLContext16setWindowCreatedEb [QtOpenGL]	_ZN10QGLContext17textureCacheLimitEv [QtOpenGL]
_ZN10QGLContext20setTextureCacheLimitEi [QtOpenGL]	_ZN10QGLContext24generateFontDisplayListsERK5QFonti [QtOpenGL]
_ZN10QGLContext5resetEv [QtOpenGL]	_ZN10QGLContext6createEPKS_ [QtOpenGL]
_ZN10QGLContext8setValidEb [LSB]	_ZN10QGLContext9setDeviceEP12QPaintDevice [QtOpenGL]
_ZN10QGLContext9setFormatERK9QGLFormat [QtOpenGL]	_ZN10QGLContext9tryVisualERK9QGLFormati [QtOpenGL]
_ZN10QGLContextC1ERK9QGLFormat [QtOpenGL]	_ZN10QGLContextC1ERK9QGLFormatP12QPaintDevice [QtOpenGL]
_ZN10QGLContextC2ERK9QGLFormat [QtOpenGL]	_ZN10QGLContextC2ERK9QGLFormatP12QPaintDevice [QtOpenGL]
_ZN10QGLContextD0Ev [QtOpenGL]	_ZN10QGLContextD1Ev [QtOpenGL]
_ZN10QGLContextD2Ev [QtOpenGL]	_ZN11QGLColormap10setEntriesEiP Kji [QtOpenGL]
_ZN11QGLColormap8setEntryEiRK6QColor [QtOpenGL]	_ZN11QGLColormap8setEntryEij [QtOpenGL]
ZN11QGLColormapC1ERKS [QtOpenGL]	_ZN11QGLColormapC1Ev [QtOpenGL]
ZN11QGLColormapC2ERKS [QtOpenGL]	_ZN11QGLColormapC2Ev [QtOpenGL]
_ZN11QGLColormapD1Ev [QtOpenGL]	_ZN11QGLColormapD2Ev [QtOpenGL]

ZN11QGLColormapSERKS [QtOpenGL]	_ZN14QGLPixelBuffer11bindTexture ERK6QImagej [QtOpenGL]
_ZN14QGLPixelBuffer11bindTexture ERK7QPixmapj [QtOpenGL]	_ZN14QGLPixelBuffer11bindTexture ERK7QString [QtOpenGL]
_ZN14QGLPixelBuffer11doneCurren tEv [QtOpenGL]	_ZN14QGLPixelBuffer11makeCurre ntEv [QtOpenGL]
_ZN14QGLPixelBuffer13deleteTextu reEj [QtOpenGL]	_ZN14QGLPixelBuffer17hasOpenGL PbuffersEv [QtOpenGL]
_ZN14QGLPixelBuffer20bindToDyn amicTextureEj [QtOpenGL]	_ZN14QGLPixelBuffer25releaseFrom DynamicTextureEv [QtOpenGL]
_ZN14QGLPixelBufferC1ERK5QSize RK9QGLFormatP9QGLWidget [QtOpenGL]	_ZN14QGLPixelBufferC1EiiRK9QGL FormatP9QGLWidget [QtXml]
_ZN14QGLPixelBufferC2ERK5QSize RK9QGLFormatP9QGLWidget [QtOpenGL]	_ZN14QGLPixelBufferC2EiiRK9QGL FormatP9QGLWidget [QtXml]
_ZN14QGLPixelBufferD0Ev [QtOpenGL]	_ZN14QGLPixelBufferD1Ev [QtOpenGL]
_ZN14QGLPixelBufferD2Ev [QtOpenGL]	_ZN20QGLFramebufferObject27has OpenGLFramebufferObjectsEv [QtXml]
_ZN20QGLFramebufferObject4bindE v [QtXml]	_ZN20QGLFramebufferObject7releas eEv [QtXml]
_ZN20QGLFramebufferObjectC1ER K5QSizej [QtXml]	_ZN20QGLFramebufferObjectC1Eiij [QtXml]
_ZN20QGLFramebufferObjectC2ER K5QSizej [QtXml]	_ZN20QGLFramebufferObjectC2Eiij [QtXml]
_ZN20QGLFramebufferObjectD0Ev [QtXml]	_ZN20QGLFramebufferObjectD1Ev [QtXml]
_ZN20QGLFramebufferObjectD2Ev [QtXml]	_ZN9QGLFormat10setOverlayEb [QtOpenGL]
_ZN9QGLFormat10setSamplesEi [QtOpenGL]	_ZN9QGLFormat10setStencilEb [QtOpenGL]
_ZN9QGLFormat13defaultFormatEv [QtOpenGL]	_ZN9QGLFormat15setDoubleBuffer Eb [QtOpenGL]
_ZN9QGLFormat15setSwapIntervalE i [QtXml]	_ZN9QGLFormat16setDefaultFormat ERKS_ [QtOpenGL]
_ZN9QGLFormat16setRedBufferSize Ei [QtXml]	_ZN9QGLFormat16setSampleBuffers Eb [QtOpenGL]
_ZN9QGLFormat17hasOpenGLOver laysEv [QtOpenGL]	_ZN9QGLFormat17setBlueBufferSize Ei [QtXml]
_ZN9QGLFormat18openGLVersionF lagsEv [QtXml]	_ZN9QGLFormat18setAccumBufferS izeEi [QtOpenGL]

_ZN9QGLFormat18setAlphaBufferSizeEi [QtOpenGL]	_ZN9QGLFormat18setDepthBufferSizeEi [QtOpenGL]
_ZN9QGLFormat18setDirectRenderingEb [QtOpenGL]	_ZN9QGLFormat18setGreenBufferSizeEi [QtXml]
_ZN9QGLFormat20defaultOverlayFormatEv [QtOpenGL]	_ZN9QGLFormat20setStencilBufferSizeEi [QtOpenGL]
ZN9QGLFormat23setDefaultOverlayFormatERKS [QtOpenGL]	_ZN9QGLFormat7setRgbaEb [QtOpenGL]
_ZN9QGLFormat8setAccumEb [QtOpenGL]	_ZN9QGLFormat8setAlphaEb [QtOpenGL]
_ZN9QGLFormat8setDepthEb [QtOpenGL]	_ZN9QGLFormat8setPlaneEi [QtOpenGL]
_ZN9QGLFormat9hasOpenGLEv [QtOpenGL]	_ZN9QGLFormat9setOptionE6QFlagsIN3QGL12FormatOptionEE [QtOpenGL]
_ZN9QGLFormat9setStereoEb [QtOpenGL]	_ZN9QGLFormatC1E6QFlagsIN3QGL12FormatOptionEEi [QtOpenGL]
ZN9QGLFormatC1ERKS [QtOpenGL]	_ZN9QGLFormatC1Ev [QtOpenGL]
_ZN9QGLFormatC2E6QFlagsIN3QGL12FormatOptionEEi [QtOpenGL]	_ZN9QGLFormatC2ERKS_ [QtOpenGL]
_ZN9QGLFormatC2Ev [QtOpenGL]	_ZN9QGLFormatD1Ev [QtOpenGL]
_ZN9QGLFormatD2Ev [QtOpenGL]	_ZN9QGLFormatASERKS_ [QtOpenGL]
_ZN9QGLWidget10paintEventEP11QPaintEvent [QtOpenGL]	_ZN9QGLWidget10renderTextEdddRK7QStringRK5QFonti [QtOpenGL]
_ZN9QGLWidget10renderTextEiiRK7QStringRK5QFonti [QtOpenGL]	_ZN9QGLWidget10setContextEP10QGLContextPKS0_b [QtOpenGL]
_ZN9QGLWidget11bindTextureERK6QImagei [QtOpenGL]	_ZN9QGLWidget11bindTextureERK7QPixmapji [QtOpenGL]
_ZN9QGLWidget11bindTextureERK7QString [QtOpenGL]	_ZN9QGLWidget11doneCurrentEv [QtOpenGL]
_ZN9QGLWidget11makeCurrentEv [QtOpenGL]	_ZN9QGLWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtOpenGL]
_ZN9QGLWidget11qt_metacastEPKc [QtOpenGL]	_ZN9QGLWidget11resizeEventEP12QResizeEvent [QtOpenGL]
_ZN9QGLWidget11setColormapERK11QGLColormap [QtOpenGL]	_ZN9QGLWidget11swapBuffersEv [QtOpenGL]
_ZN9QGLWidget12initializeGLEv [QtOpenGL]	_ZN9QGLWidget12renderPixmapEii b [QtOpenGL]

_ZN9QGLWidget13deleteTextureEj [QtOpenGL]	_ZN9QGLWidget14paintOverlayGLEv [QtOpenGL]
_ZN9QGLWidget15grabFramebufferEb [QtOpenGL]	_ZN9QGLWidget15resizeOverlayGLEii [QtOpenGL]
_ZN9QGLWidget15updateOverlayGLEv [QtOpenGL]	_ZN9QGLWidget16setMouseTrackingEb [QtOpenGL]
_ZN9QGLWidget17convertToGLFormatERK6QImage [QtOpenGL]	_ZN9QGLWidget17setAutoBufferSwapEb [QtOpenGL]
_ZN9QGLWidget18makeOverlayCurrentEv [QtOpenGL]	_ZN9QGLWidget19fontDisplayListBaseERK5QFonti [QtOpenGL]
_ZN9QGLWidget19initializeOverlayGLEv [QtOpenGL]	_ZN9QGLWidget5eventEP6QEvent [QtOpenGL]
_ZN9QGLWidget6glDrawEv [QtOpenGL]	_ZN9QGLWidget6glInitEv [QtOpenGL]
_ZN9QGLWidget7paintGLEv [QtOpenGL]	_ZN9QGLWidget8resizeGLEii [QtOpenGL]
_ZN9QGLWidget8updateGLEv [QtOpenGL]	_ZN9QGLWidget9setFormatERK9QGLFormat [QtOpenGL]
_ZN9QGLWidgetC1EP10QGLContextP7QWidgetPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]	_ZN9QGLWidgetC1EP10QGLContextP7QWidgetPKcPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]
_ZN9QGLWidgetC1EP7QWidgetPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]	_ZN9QGLWidgetC1EP7QWidgetPKcPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]
_ZN9QGLWidgetC1ERK9QGLFormatP7QWidgetPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]	_ZN9QGLWidgetC1ERK9QGLFormatP7QWidgetPKcPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]
_ZN9QGLWidgetC2EP10QGLContextP7QWidgetPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]	_ZN9QGLWidgetC2EP10QGLContextP7QWidgetPKcPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]
_ZN9QGLWidgetC2EP7QWidgetPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]	_ZN9QGLWidgetC2EP7QWidgetPKcPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]
_ZN9QGLWidgetC2ERK9QGLFormatP7QWidgetPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]	_ZN9QGLWidgetC2ERK9QGLFormatP7QWidgetPKcPKS_6QFlagsIN2Qt10WindowTypeEE [QtOpenGL]
_ZN9QGLWidgetD0Ev [QtOpenGL]	_ZN9QGLWidgetD1Ev [QtOpenGL]
_ZN9QGLWidgetD2Ev [QtOpenGL]	_ZNK10QGLContext10colorIndexERK6QColor [QtOpenGL]
_ZNK10QGLContext11initializedEv [QtOpenGL]	_ZNK10QGLContext11swapBuffersEv [QtOpenGL]
_ZNK10QGLContext13windowCreatedEv [QtOpenGL]	_ZNK10QGLContext14deviceIsPixmapEv [QtOpenGL]

_ZNK10QGLContext14getProcAddressERK7QString [QtOpenGL]	_ZNK10QGLContext15requestedFormatEv [QtOpenGL]
_ZNK10QGLContext23overlayTransparentColorEv [QtOpenGL]	_ZNK10QGLContext6deviceEv [QtOpenGL]
_ZNK10QGLContext6formatEv [QtOpenGL]	_ZNK10QGLContext7isValidEv [QtOpenGL]
_ZNK10QGLContext9isSharingEv [QtOpenGL]	_ZNK11QGLColormap10entryColorEi [QtOpenGL]
_ZNK11QGLColormap11findNearestEj [QtOpenGL]	_ZNK11QGLColormap4findEj [QtOpenGL]
_ZNK11QGLColormap4sizeEv [QtOpenGL]	_ZNK11QGLColormap7isEmptyEv [QtOpenGL]
_ZNK11QGLColormap8entryRgbEi [QtOpenGL]	_ZNK14QGLPixelBuffer11paintEngineEv [QtOpenGL]
_ZNK14QGLPixelBuffer20updateDynamicTextureEj [QtOpenGL]	_ZNK14QGLPixelBuffer22generateDynamicTextureEv [QtOpenGL]
_ZNK14QGLPixelBuffer4sizeEv [QtOpenGL]	_ZNK14QGLPixelBuffer6formatEv [QtOpenGL]
_ZNK14QGLPixelBuffer6handleEv [QtOpenGL]	_ZNK14QGLPixelBuffer6metricEN12QPaintDevice17PaintDeviceMetricE [QtOpenGL]
_ZNK14QGLPixelBuffer7isValidEv [QtOpenGL]	_ZNK14QGLPixelBuffer7toImageEv [QtOpenGL]
_ZNK20QGLFramebufferObject11paintEngineEv [QtXml]	_ZNK20QGLFramebufferObject4sizeEv [QtXml]
_ZNK20QGLFramebufferObject6handleEv [QtXml]	_ZNK20QGLFramebufferObject6metricEN12QPaintDevice17PaintDeviceMetricE [QtXml]
_ZNK20QGLFramebufferObject7isValidEv [QtXml]	_ZNK20QGLFramebufferObject7textureEv [QtXml]
_ZNK20QGLFramebufferObject7toImageEv [QtXml]	_ZNK9QGLFormat10testOptionE6QFlagsIN3QGL12FormatOptionEE [QtOpenGL]
_ZNK9QGLFormat12swapIntervalEv [QtXml]	_ZNK9QGLFormat13redBufferSizeEv [QtXml]
_ZNK9QGLFormat14blueBufferSizeEv [QtXml]	_ZNK9QGLFormat15accumBufferSizeEv [QtOpenGL]
_ZNK9QGLFormat15alphaBufferSizeEv [QtOpenGL]	_ZNK9QGLFormat15depthBufferSizeEv [QtOpenGL]
_ZNK9QGLFormat15greenBufferSizeEv [QtXml]	_ZNK9QGLFormat17stencilBufferSizeEv [QtOpenGL]
_ZNK9QGLFormat5planeEv [QtOpenGL]	_ZNK9QGLFormat7samplesEv [QtOpenGL]

_ZNK9QGLWidget10metaObjectEv [QtOpenGL]	_ZNK9QGLWidget11paintEngineEv [QtOpenGL]
_ZNK9QGLWidget12doubleBufferEv [QtOpenGL]	_ZNK9QGLWidget13qglClearColorERK6QColor [QtOpenGL]
_ZNK9QGLWidget14autoBufferSwapEv [QtOpenGL]	_ZNK9QGLWidget14overlayContextEv [QtOpenGL]
_ZNK9QGLWidget6formatEv [QtOpenGL]	_ZNK9QGLWidget7contextEv [QtOpenGL]
_ZNK9QGLWidget7isValidEv [QtOpenGL]	_ZNK9QGLWidget8colormapEv [QtOpenGL]
_ZNK9QGLWidget8qglColorERK6QColor [QtOpenGL]	_ZNK9QGLWidget9isSharingEv [QtOpenGL]
ZeqRK9QGLFormatS1 [QtOpenGL]	_ZneRK9QGLFormatS1_ [QtOpenGL]

An LSB conforming implementation shall provide the generic data interfaces for Qt4 OpenGL specified in Table 18-558, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-558 libQtOpenGL - Qt4 OpenGL Data Interfaces

_ZN10QGLContext10currentCtxE [QtOpenGL]	_ZN9QGLWidget16staticMetaObjectE [QtOpenGL]
_ZTI10QGLContext [CXXABI-1.86]	_ZTI14QGLPixelBuffer [CXXABI-1.86]
_ZTI20QGLFramebufferObject [CXXABI-1.86]	_ZTI9QGLWidget [CXXABI-1.86]
_ZTV10QGLContext [CXXABI-1.86]	_ZTV14QGLPixelBuffer [CXXABI-1.86]
_ZTV20QGLFramebufferObject [CXXABI-1.86]	_ZTV9QGLWidget [CXXABI-1.86]

18.11 Data Definitions for libQtOpenGL

This section defines global identifiers and their values that are associated with interfaces contained in libQtOpenGL. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The

C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

18.11.1 QtOpenGL/qgl.h

```
enum QGL::FormatOption {
    DoubleBuffer = 1,
    DepthBuffer = 2,
    Rgba = 4,
    AlphaChannel = 8,
    AccumBuffer = 16,
    StencilBuffer = 32,
    StereoBuffers = 64,
    DirectRendering = 128,
    HasOverlay = 256,
    SampleBuffers = 512,
    SingleBuffer = 65536,
    NoDepthBuffer = 131072,
    ColorIndex = 262144,
    NoAlphaChannel = 524288,
    NoAccumBuffer = 1048576,
    NoStencilBuffer = 2097152,
    NoStereoBuffers = 4194304,
    IndirectRendering = 8388608,
    NoOverlay = 16777216,
    NoSampleBuffers = 33554432
};
class QFlags < QGL::FormatOption >;
typedef class QFlags < QGL::FormatOption > QGL::FormatOptions;
class QGLFormat;
class QGLContext;
class QGLWidget;
```

18.11.2 QtOpenGL/qglcolormap.h

```
class QGLColormap;
struct QGLColormap::QGLColormapData;
```

18.11.3 QtOpenGL/qglpixelbuffer.h

```
typedef enum QtValidLicenseForOpenGLModule QtOpenGLModule;
class QGLPixelBuffer;
```

18.12 Interface Definitions for libQtOpenGL

The interfaces defined on the following pages are included in libQtOpenGL and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 18.10 shall behave as described in the referenced base document.

_ZN10QGLContext8setValidEb**Name**

`QGLContext::setValid` — set validity of the GL rendering context

Synopsis

```
#include <QtOpenGL/qgl.h>
void QGLContext::setValid (bool valid);
```

Description

The `QGLContext::setValid()` function shall set validity of the GL rendering context to *validity* (i.e. force the context to be either valid or not valid).

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

18.13 Interfaces for libQtSql

Table 18-559 defines the library name and shared object name for the libQtSql library

Table 18-559 libQtSql Definition

Library:	libQtSql
SONAME:	libQtSql.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI

[LSB] This Specification

[QtSql] QtSql 4.2.0

[QtXml] QtXml 4.2.0

18.13.1 Qt4 SQL**18.13.1.1 Class data for QSqlResult**

The virtual table for the `QSqlResult` class is described by Table 18-560

Table 18-560 Primary vtable for QSqlResult

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for <code>QSqlResult</code>
<code>vfunc[0]:</code>	<code>QSqlResult::~~QSqlResult()</code>
<code>vfunc[1]:</code>	<code>QSqlResult::~~QSqlResult()</code>
<code>vfunc[2]:</code>	<code>QSqlResult::handle() const</code>
<code>vfunc[3]:</code>	<code>QSqlResult::setAt(int)</code>
<code>vfunc[4]:</code>	<code>QSqlResult::setActive(bool)</code>

vfunc[5]:	QSqlResult::setLastError(QSqlError const&)
vfunc[6]:	QSqlResult::setQuery(QString const&)
vfunc[7]:	QSqlResult::setSelect(bool)
vfunc[8]:	QSqlResult::setForwardOnly(bool)
vfunc[9]:	QSqlResult::exec()
vfunc[10]:	QSqlResult::prepare(QString const&)
vfunc[11]:	QSqlResult::savePrepare(QString const&)
vfunc[12]:	QSqlResult::bindValue(int, QVariant const&, QFlags<QSql::ParamTypeFlag>)
vfunc[13]:	QSqlResult::bindValue(QString const&, QVariant const&, QFlags<QSql::ParamTypeFlag>)
vfunc[14]:	__cxa_pure_virtual
vfunc[15]:	__cxa_pure_virtual
vfunc[16]:	__cxa_pure_virtual
vfunc[17]:	__cxa_pure_virtual
vfunc[18]:	QSqlResult::fetchNext()
vfunc[19]:	QSqlResult::fetchPrevious()
vfunc[20]:	__cxa_pure_virtual
vfunc[21]:	__cxa_pure_virtual
vfunc[22]:	__cxa_pure_virtual
vfunc[23]:	__cxa_pure_virtual
vfunc[24]:	QSqlResult::record() const
vfunc[25]:	QSqlResult::lastInsertId() const
vfunc[26]:	QSqlResult::virtual_hook(int, void*)

The Run Time Type Information for the QSqlResult class is described by Table 18-561

Table 18-561 typeinfo for QSqlResult

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QSqlResult

18.13.1.2 Class data for QSqlDriverFactoryInterface

The virtual table for the QSqlDriverFactoryInterface class is described by Table 18-562

Table 18-562 Primary vtable for QSqlDriverFactoryInterface

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSqlDriverFactoryInterface
vfunc[0]:	NULL or QSqlDriverFactoryInterface::~~QSqlDriverFactoryInterface()
vfunc[1]:	NULL or QSqlDriverFactoryInterface::~~QSqlDriverFactoryInterface()
vfunc[2]:	__cxa_pure_virtual
vfunc[3]:	__cxa_pure_virtual

The Run Time Type Information for the QSqlDriverFactoryInterface class is described by Table 18-563

Table 18-563 typeinfo for QSqlDriverFactoryInterface

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSqlDriverFactoryInterface
basetype:	typeinfo for QFactoryInterface

18.13.1.3 Class data for QSqlDriverPlugin

The virtual table for the QSqlDriverPlugin class is described by Table 18-564

Table 18-564 Primary vtable for QSqlDriverPlugin

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSqlDriverPlugin
vfunc[0]:	QSqlDriverPlugin::metaObject() const
vfunc[1]:	QSqlDriverPlugin::qt_metacast(char const*)
vfunc[2]:	QSqlDriverPlugin::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSqlDriverPlugin::~~QSqlDriverPlugin()

vfunc[4]:	QSqlDriverPlugin::~~QSqlDriverPlugin()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	__cxa_pure_virtual
vfunc[13]:	__cxa_pure_virtual

The Run Time Type Information for the QSqlDriverPlugin class is described by Table 18-565

Table 18-565 typeinfo for QSqlDriverPlugin

Base Vtable	vtable for __cxxabiv1::__vmi_class _type_info	2
Name	typeinfo name for QSqlDriverPlugin	
flags:	0	
basetype:	typeinfo for QObject	
basetype:	typeinfo for QSqlDriverFactoryInter face	2050

18.13.1.4 Class data for QSqlDriver

The virtual table for the QSqlDriver class is described by Table 18-566

Table 18-566 Primary vtable for QSqlDriver

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSqlDriver
vfunc[0]:	QSqlDriver::metaObject() const
vfunc[1]:	QSqlDriver::qt_metacast(char const*)
vfunc[2]:	QSqlDriver::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSqlDriver::~~QSqlDriver()

vfunc[4]:	QSqlDriver::~~QSqlDriver()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QSqlDriver::isOpen() const
vfunc[13]:	QSqlDriver::beginTransaction()
vfunc[14]:	QSqlDriver::commitTransaction()
vfunc[15]:	QSqlDriver::rollbackTransaction()
vfunc[16]:	QSqlDriver::tables(QSql::TableType) const
vfunc[17]:	QSqlDriver::primaryIndex(QString const&) const
vfunc[18]:	QSqlDriver::record(QString const&) const
vfunc[19]:	QSqlDriver::formatValue(QSqlField const&, bool) const
vfunc[20]:	QSqlDriver::escapeIdentifier(QString const&, QSqlDriver::IdentifierType) const
vfunc[21]:	QSqlDriver::sqlStatement(QSqlDriver::StatementType, QString const&, QSqlRecord const&, bool) const
vfunc[22]:	QSqlDriver::handle() const
vfunc[23]:	__cxa_pure_virtual
vfunc[24]:	__cxa_pure_virtual
vfunc[25]:	__cxa_pure_virtual
vfunc[26]:	__cxa_pure_virtual
vfunc[27]:	QSqlDriver::setOpen(bool)
vfunc[28]:	QSqlDriver::setOpenError(bool)
vfunc[29]:	QSqlDriver::setLastError(QSqlError const&)

The Run Time Type Information for the QSqlDriver class is described by Table 18-567

Table 18-567 typeinfo for QSqlDriver

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSqlDriver
basetype:	typeinfo for QObject

18.13.1.5 Class data for QSqlQueryModel

The virtual table for the QSqlQueryModel class is described by Table 18-568

Table 18-568 Primary vtable for QSqlQueryModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSqlQueryModel
vfunc[0]:	QSqlQueryModel::metaObject() const
vfunc[1]:	QSqlQueryModel::qt_metacast(char const*)
vfunc[2]:	QSqlQueryModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSqlQueryModel::~~QSqlQueryModel()
vfunc[4]:	QSqlQueryModel::~~QSqlQueryModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractTableModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QAbstractTableModel::parent(QModelIndex const&) const
vfunc[14]:	QSqlQueryModel::rowCount(QModelIndex const&) const
vfunc[15]:	QSqlQueryModel::columnCount(QModelIndex const&) const

vfunc[16]:	QAbstractTableModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QSqlQueryModel::data(QModelIndex const&, int) const
vfunc[18]:	QAbstractItemModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QSqlQueryModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QSqlQueryModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractTableModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QAbstractItemModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QSqlQueryModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QAbstractItemModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QSqlQueryModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QSqlQueryModel::fetchMore(QModelIndex const&)
vfunc[32]:	QSqlQueryModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QAbstractItemModel::flags(QModelIndex const&) const
vfunc[34]:	QAbstractItemModel::sort(int, Qt::SortOrder)

vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QAbstractItemModel::submit()
vfunc[39]:	QAbstractItemModel::revert()
vfunc[40]:	QSqlQueryModel::clear()
vfunc[41]:	QSqlQueryModel::queryChange()

The Run Time Type Information for the QSqlQueryModel class is described by Table 18-569

Table 18-569 typeinfo for QSqlQueryModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSqlQueryModel
basetype:	typeinfo for QAbstractTableModel

18.13.1.6 Class data for QSqlTableModel

The virtual table for the QSqlTableModel class is described by Table 18-570

Table 18-570 Primary vtable for QSqlTableModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSqlTableModel
vfunc[0]:	QSqlTableModel::metaObject() const
vfunc[1]:	QSqlTableModel::qt_metacast(char const*)
vfunc[2]:	QSqlTableModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSqlTableModel::~QSqlTableModel()
vfunc[4]:	QSqlTableModel::~QSqlTableModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)

vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractTableModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QAbstractTableModel::parent(QModelIndex const&) const
vfunc[14]:	QSqlTableModel::rowCount(QModelIndex const&) const
vfunc[15]:	QSqlQueryModel::columnCount(QModelIndex const&) const
vfunc[16]:	QAbstractTableModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QSqlTableModel::data(QModelIndex const&, int) const
vfunc[18]:	QSqlTableModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QSqlTableModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QSqlQueryModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractTableModel::dropMimeType(QMimeType const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QSqlTableModel::insertRows(int, int, QModelIndex const&)

vfunc[28]:	QSqlQueryModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QSqlTableModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QSqlTableModel::removeColumns(int, int, QModelIndex const&)
vfunc[31]:	QSqlQueryModel::fetchMore(QModelIndex const&)
vfunc[32]:	QSqlQueryModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QSqlTableModel::flags(QModelIndex const&) const
vfunc[34]:	QSqlTableModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QSqlTableModel::submit()
vfunc[39]:	QSqlTableModel::revert()
vfunc[40]:	QSqlTableModel::clear()
vfunc[41]:	QSqlQueryModel::queryChange()
vfunc[42]:	QSqlTableModel::select()
vfunc[43]:	QSqlTableModel::setTable(QString const&)
vfunc[44]:	QSqlTableModel::setEditStrategy(QSqlTableModel::EditStrategy)
vfunc[45]:	QSqlTableModel::setSort(int, Qt::SortOrder)
vfunc[46]:	QSqlTableModel::setFilter(QString const&)
vfunc[47]:	QSqlTableModel::revertRow(int)
vfunc[48]:	QSqlTableModel::updateRowInTable(int, QSqlRecord const&)
vfunc[49]:	QSqlTableModel::insertRowIntoTable(QSqlRecord const&)
vfunc[50]:	QSqlTableModel::deleteRowFromTable(int)

vfunc[51]:	QSqlTableModel::orderByClause() const
vfunc[52]:	QSqlTableModel::selectStatement() const

The Run Time Type Information for the QSqlTableModel class is described by Table 18-571

Table 18-571 typeinfo for QSqlTableModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSqlTableModel
basetype:	typeinfo for QSqlQueryModel

18.13.1.7 Class data for QSqlRelationalTableModel

The virtual table for the QSqlRelationalTableModel class is described by Table 18-572

Table 18-572 Primary vtable for QSqlRelationalTableModel

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSqlRelationalTableModel
vfunc[0]:	QSqlRelationalTableModel::metaObject() const
vfunc[1]:	QSqlRelationalTableModel::qt_metacast(char const*)
vfunc[2]:	QSqlRelationalTableModel::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSqlRelationalTableModel::~~QSqlRelationalTableModel()
vfunc[4]:	QSqlRelationalTableModel::~~QSqlRelationalTableModel()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QAbstractTableModel::index(int, int, QModelIndex const&) const
vfunc[13]:	QAbstractTableModel::parent(QModelIndex const&) const
vfunc[14]:	QSqlTableModel::rowCount(QModelIndex const&) const
vfunc[15]:	QSqlQueryModel::columnCount(QModelIndex const&) const
vfunc[16]:	QAbstractTableModel::hasChildren(QModelIndex const&) const
vfunc[17]:	QSqlRelationalTableModel::data(QModelIndex const&, int) const
vfunc[18]:	QSqlRelationalTableModel::setData(QModelIndex const&, QVariant const&, int)
vfunc[19]:	QSqlTableModel::headerData(int, Qt::Orientation, int) const
vfunc[20]:	QSqlQueryModel::setHeaderData(int, Qt::Orientation, QVariant const&, int)
vfunc[21]:	QAbstractItemModel::itemData(QModelIndex const&) const
vfunc[22]:	QAbstractItemModel::setItemData(QModelIndex const&, QMap<int, QVariant> const&)
vfunc[23]:	QAbstractItemModel::mimeTypes() const
vfunc[24]:	QAbstractItemModel::mimeData(QList<QModelIndex> const&) const
vfunc[25]:	QAbstractTableModel::dropMimeData(QMimeData const*, Qt::DropAction, int, int, QModelIndex const&)
vfunc[26]:	QAbstractItemModel::supportedDropActions() const
vfunc[27]:	QSqlTableModel::insertRows(int, int, QModelIndex const&)
vfunc[28]:	QSqlQueryModel::insertColumns(int, int, QModelIndex const&)
vfunc[29]:	QSqlTableModel::removeRows(int, int, QModelIndex const&)
vfunc[30]:	QSqlTableModel::removeColumns(int, int, QModelIndex const&)

vfunc[31]:	QSqlQueryModel::fetchMore(QModelIndex const&)
vfunc[32]:	QSqlQueryModel::canFetchMore(QModelIndex const&) const
vfunc[33]:	QSqlTableModel::flags(QModelIndex const&) const
vfunc[34]:	QSqlTableModel::sort(int, Qt::SortOrder)
vfunc[35]:	QAbstractItemModel::buddy(QModelIndex const&) const
vfunc[36]:	QAbstractItemModel::match(QModelIndex const&, int, QVariant const&, int, QFlags<Qt::MatchFlag>) const
vfunc[37]:	QAbstractItemModel::span(QModelIndex const&) const
vfunc[38]:	QSqlTableModel::submit()
vfunc[39]:	QSqlTableModel::revert()
vfunc[40]:	QSqlRelationalTableModel::clear()
vfunc[41]:	QSqlQueryModel::queryChange()
vfunc[42]:	QSqlRelationalTableModel::select()
vfunc[43]:	QSqlRelationalTableModel::setTable(QString const&)
vfunc[44]:	QSqlTableModel::setEditStrategy(QSqlTableModel::EditStrategy)
vfunc[45]:	QSqlTableModel::setSort(int, Qt::SortOrder)
vfunc[46]:	QSqlTableModel::setFilter(QString const&)
vfunc[47]:	QSqlRelationalTableModel::revertRow(int)
vfunc[48]:	QSqlRelationalTableModel::updateRowInTable(int, QSqlRecord const&)
vfunc[49]:	QSqlTableModel::insertRowIntoTable(QSqlRecord const&)
vfunc[50]:	QSqlTableModel::deleteRowFromTable(int)
vfunc[51]:	QSqlRelationalTableModel::orderByClause() const
vfunc[52]:	QSqlRelationalTableModel::selectStatement() const

vfunc[53]:	QSqlRelationalTableModel::setRelation(int, QSqlRelation const&)
vfunc[54]:	QSqlRelationalTableModel::relationModel(int) const

The Run Time Type Information for the QSqlRelationalTableModel class is described by Table 18-573

Table 18-573 typeinfo for QSqlRelationalTableModel

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSqlRelationalTableModel
basetype:	typeinfo for QSqlTableModel

18.13.1.8 Interfaces for Qt4 SQL

An LSB conforming implementation shall provide the generic functions for Qt4 SQL specified in Table 18-574, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-574 libQtSql - Qt4 SQL Function Interfaces

_ZN10QSqlDriver11qt_metacallEN11QMetaObject4CallEiPPv [QtSql]	_ZN10QSqlDriver11qt_metacastEPKc [QtSql]
_ZN10QSqlDriver12setLastErrorERK9QSqlError [QtSql]	_ZN10QSqlDriver12setOpenErrorEb [QtSql]
_ZN10QSqlDriver16beginTransactionEv [QtSql]	_ZN10QSqlDriver17commitTransactionEv [QtSql]
_ZN10QSqlDriver19rollbackTransactionEv [QtSql]	_ZN10QSqlDriver7setOpenEb [QtSql]
_ZN10QSqlDriverC1EP7QObject [QtSql]	_ZN10QSqlDriverC2EP7QObject [QtSql]
_ZN10QSqlDriverD0Ev [QtSql]	_ZN10QSqlDriverD1Ev [QtSql]
_ZN10QSqlDriverD2Ev [QtSql]	_ZN10QSqlRecord11clearValuesEv [QtSql]
_ZN10QSqlRecord12setGeneratedERK7QStringb [QtSql]	_ZN10QSqlRecord12setGeneratedEib [QtSql]
_ZN10QSqlRecord5clearEv [QtSql]	_ZN10QSqlRecord6appendERK9QSqlField [QtSql]
_ZN10QSqlRecord6insertEiRK9QSqlField [QtSql]	_ZN10QSqlRecord6removeEi [QtSql]
_ZN10QSqlRecord7replaceEiRK9QSqlField [QtSql]	_ZN10QSqlRecord7setNullERK7QString [QtSql]
_ZN10QSqlRecord7setNullEi [QtSql]	_ZN10QSqlRecord8setValueERK7QStringRK8QVariant [QtSql]

_ZN10QSqlRecord8setValueEiRK8QVariant [QtSql]	_ZN10QSqlRecordC1ERKS_ [QtSql]
_ZN10QSqlRecordC1Ev [QtSql]	_ZN10QSqlRecordC2ERKS_ [QtSql]
_ZN10QSqlRecordC2Ev [QtSql]	_ZN10QSqlRecordD1Ev [QtSql]
_ZN10QSqlRecordD2Ev [QtSql]	_ZN10QSqlRecordaSERKS_ [QtSql]
_ZN10QSqlResult11savePrepareERK7QString [QtSql]	_ZN10QSqlResult12addBindValueERK8QVariant6QFlagsIN4QSql13ParamTypeFlagEE [QtSql]
_ZN10QSqlResult12setLastErrorERK9QSqlError [QtSql]	_ZN10QSqlResult12virtual_hookEiPv [QtSql]
_ZN10QSqlResult13fetchPreviousEv [QtSql]	_ZN10QSqlResult14setForwardOnlyEb [QtSql]
_ZN10QSqlResult4execEv [QtSql]	_ZN10QSqlResult5clearEv [QtSql]
_ZN10QSqlResult5setAtEi [QtSql]	_ZN10QSqlResult7prepareERK7QString [QtSql]
_ZN10QSqlResult8setQueryERK7QString [QtSql]	_ZN10QSqlResult9bindValueERK7QStringRK8QVariant6QFlagsIN4QSql13ParamTypeFlagEE [QtSql]
_ZN10QSqlResult9bindValueEiRK8QVariant6QFlagsIN4QSql13ParamTypeFlagEE [QtSql]	_ZN10QSqlResult9execBatchEb [LSB]
_ZN10QSqlResult9fetchNextEv [QtSql]	_ZN10QSqlResult9setActiveEb [QtSql]
_ZN10QSqlResult9setSelectEb [QtSql]	_ZN10QSqlResultC1EPK10QSqlDriver [QtSql]
_ZN10QSqlResultC2EPK10QSqlDriver [QtSql]	_ZN10QSqlResultD0Ev [QtSql]
_ZN10QSqlResultD1Ev [QtSql]	_ZN10QSqlResultD2Ev [QtSql]
_ZN12QSqlDatabase11addDatabaseEP10QSqlDriverRK7QString [QtSql]	_ZN12QSqlDatabase11addDatabaseERK7QStringS2_ [QtSql]
_ZN12QSqlDatabase11setHostNameERK7QString [QtSql]	_ZN12QSqlDatabase11setPasswordERK7QString [QtSql]
_ZN12QSqlDatabase11setUserNameERK7QString [QtSql]	_ZN12QSqlDatabase11transactionEv [QtSql]
_ZN12QSqlDatabase13cloneDatabaseERKS_RK7QString [QtSql]	_ZN12QSqlDatabase14removeDatabaseERK7QString [QtSql]
_ZN12QSqlDatabase15connectionNamesEv [QtSql]	_ZN12QSqlDatabase15setDatabaseNameERK7QString [QtSql]
_ZN12QSqlDatabase17isDriverAvailableERK7QString [QtSql]	_ZN12QSqlDatabase17registerSqlDriverERK7QStringP21QSqlDriverCreatorBase [QtSql]

_ZN12QSqlDatabase17setConnectOptionsERK7QString [QtSql]	_ZN12QSqlDatabase4openERK7QStringS2_ [QtSql]
_ZN12QSqlDatabase4openEv [QtSql]	_ZN12QSqlDatabase5closeEv [QtSql]
_ZN12QSqlDatabase6commitEv [QtSql]	_ZN12QSqlDatabase7driversEv [QtSql]
_ZN12QSqlDatabase7setPortEi [QtSql]	_ZN12QSqlDatabase8containsERK7QString [QtSql]
_ZN12QSqlDatabase8databaseERK7QStringb [QtSql]	_ZN12QSqlDatabase8rollbackEv [QtSql]
_ZN12QSqlDatabaseC1EP10QSqlDriver [QtSql]	_ZN12QSqlDatabaseC1ERK7QString [QtSql]
ZN12QSqlDatabaseC1ERKS [QtSql]	_ZN12QSqlDatabaseC1Ev [QtSql]
_ZN12QSqlDatabaseC2EP10QSqlDriver [QtSql]	_ZN12QSqlDatabaseC2ERK7QString [QtSql]
ZN12QSqlDatabaseC2ERKS [QtSql]	_ZN12QSqlDatabaseC2Ev [QtSql]
_ZN12QSqlDatabaseD1Ev [QtSql]	_ZN12QSqlDatabaseD2Ev [QtSql]
ZN12QSqlDatabaseaSERKS [QtSql]	_ZN14QSqlQueryModel11qt_metacallEN11QMetaObject4CallEiPPv [QtSql]
_ZN14QSqlQueryModel11qt_metacastEPKc [QtSql]	_ZN14QSqlQueryModel11queryChangeEvent [QtSql]
_ZN14QSqlQueryModel12setLastErrorERK9QSqlError [QtSql]	_ZN14QSqlQueryModel13insertColumnsEiiRK11QModelIndex [QtSql]
_ZN14QSqlQueryModel13removeColumnsEiiRK11QModelIndex [QtSql]	_ZN14QSqlQueryModel13setHeaderDataEiN2Qt11OrientationERK8QVarianti [QtSql]
_ZN14QSqlQueryModel5clearEv [QtSql]	_ZN14QSqlQueryModel8setQueryERK7QStringRK12QSqlDatabase [QtSql]
_ZN14QSqlQueryModel8setQueryERK9QSqlQuery [QtSql]	_ZN14QSqlQueryModel9fetchMoreERK11QModelIndex [QtSql]
_ZN14QSqlQueryModelC1EP7QObject [QtSql]	_ZN14QSqlQueryModelC2EP7QObject [QtSql]
_ZN14QSqlQueryModelD0Ev [QtSql]	_ZN14QSqlQueryModelD1Ev [QtSql]
_ZN14QSqlQueryModelD2Ev [QtSql]	_ZN14QSqlTableModel10insertRowsEiiRK11QModelIndex [QtSql]
_ZN14QSqlTableModel10removeRowsEiiRK11QModelIndex [QtSql]	_ZN14QSqlTableModel11primeInsertEiR10QSqlRecord [QtSql]

_ZN14QSqlTableModel11qt_metacallEN11QMetaObject4CallEiPPv [QtSql]	_ZN14QSqlTableModel11qt_metacastEPKc [QtSql]
_ZN14QSqlTableModel12beforeDeleteEi [QtSql]	_ZN14QSqlTableModel12beforeInsertER10QSqlRecord [QtSql]
_ZN14QSqlTableModel12beforeUpdateEiR10QSqlRecord [QtSql]	_ZN14QSqlTableModel12insertRecordEiRK10QSqlRecord [QtSql]
_ZN14QSqlTableModel13removeColumnsEiiRK11QModelIndex [QtSql]	_ZN14QSqlTableModel13setPrimaryKeyERK9QSqlIndex [QtSql]
_ZN14QSqlTableModel15setEditStrategyENS_12EditStrategyE [QtSql]	_ZN14QSqlTableModel16updateRowInTableEiRK10QSqlRecord [QtSql]
_ZN14QSqlTableModel18deleteRowFromTableEi [QtSql]	_ZN14QSqlTableModel18insertRowIntoTableERK10QSqlRecord [QtSql]
_ZN14QSqlTableModel4sortEiN2Qt9SortOrderE [QtSql]	_ZN14QSqlTableModel5clearEv [QtSql]
_ZN14QSqlTableModel6revertEv [QtSql]	_ZN14QSqlTableModel6selectEv [QtSql]
_ZN14QSqlTableModel6submitEv [QtSql]	_ZN14QSqlTableModel7setDataERK11QModelIndexRK8QVarianti [QtSql]
_ZN14QSqlTableModel7setSortEiN2Qt9SortOrderE [QtSql]	_ZN14QSqlTableModel8setQueryERK9QSqlQuery [QtSql]
_ZN14QSqlTableModel8setTableERK7QString [QtSql]	_ZN14QSqlTableModel9revertAllEv [QtSql]
_ZN14QSqlTableModel9revertRowEi [QtSql]	_ZN14QSqlTableModel9setFilterERK7QString [QtSql]
_ZN14QSqlTableModel9setRecordEiRK10QSqlRecord [QtSql]	_ZN14QSqlTableModel9submitAllEv [QtSql]
_ZN14QSqlTableModelC1EP7QObject12QSqlDatabase [QtSql]	_ZN14QSqlTableModelC2EP7QObject12QSqlDatabase [QtSql]
_ZN14QSqlTableModelD0Ev [QtSql]	_ZN14QSqlTableModelD1Ev [QtSql]
_ZN14QSqlTableModelD2Ev [QtSql]	_ZN16QSqlDriverPlugin11qt_metacallEN11QMetaObject4CallEiPPv [QtSql]
_ZN16QSqlDriverPlugin11qt_metacastEPKc [QtSql]	_ZN16QSqlDriverPluginC1EP7QObject [QtSql]
_ZN16QSqlDriverPluginC2EP7QObject [QtSql]	_ZN16QSqlDriverPluginD0Ev [QtSql]
_ZN16QSqlDriverPluginD1Ev [QtSql]	_ZN16QSqlDriverPluginD2Ev [QtSql]
_ZN24QSqlRelationalTableModel11qt_metacallEN11QMetaObject4CallEiPPv [QtSql]	_ZN24QSqlRelationalTableModel11qt_metacastEPKc [QtSql]

_ZN24QSqlRelationalTableModel11setRelationEiRK12QSqlRelation [QtSql]	_ZN24QSqlRelationalTableModel13removeColumnsEiRK11QModelIndex [QtXml]
_ZN24QSqlRelationalTableModel16updateRowInTableEiRK10QSqlRecord [QtSql]	_ZN24QSqlRelationalTableModel18insertRowIntoTableERK10QSqlRecord [QtXml]
_ZN24QSqlRelationalTableModel5clearEv [QtSql]	_ZN24QSqlRelationalTableModel6selectEv [QtSql]
_ZN24QSqlRelationalTableModel7setDataERK11QModelIndexRK8QVariant [QtSql]	_ZN24QSqlRelationalTableModel8setTableERK7QString [QtSql]
_ZN24QSqlRelationalTableModel9revertRowEi [QtSql]	_ZN24QSqlRelationalTableModelC1EP7QObject12QSqlDatabase [QtSql]
_ZN24QSqlRelationalTableModelC2EP7QObject12QSqlDatabase [QtSql]	_ZN24QSqlRelationalTableModelD0Ev [QtSql]
_ZN24QSqlRelationalTableModelD1Ev [QtSql]	_ZN24QSqlRelationalTableModelD2Ev [QtSql]
_ZN9QSqlError13setDriverTextERK7QString [QtSql]	_ZN9QSqlError15setDatabaseTextERK7QString [QtSql]
_ZN9QSqlError7setTypeENS_9ErrorTypeE [QtSql]	_ZN9QSqlError9setNumberEi [QtSql]
_ZN9QSqlErrorC1ERK7QStringS2_NS_9ErrorTypeEi [QtSql]	_ZN9QSqlErrorC1ERKS_ [QtSql]
_ZN9QSqlErrorC2ERK7QStringS2_NS_9ErrorTypeEi [QtSql]	_ZN9QSqlErrorC2ERKS_ [QtSql]
_ZN9QSqlErrorD1Ev [QtSql]	_ZN9QSqlErrorD2Ev [QtSql]
ZN9QSqlErroraSERKS [QtSql]	_ZN9QSqlField10setSqlTypeEi [QtSql]
_ZN9QSqlField11setReadOnlyEb [QtSql]	_ZN9QSqlField12setAutoValueEb [QtSql]
_ZN9QSqlField12setGeneratedEb [QtSql]	_ZN9QSqlField12setPrecisionEi [QtSql]
_ZN9QSqlField15setDefaultValueERK8QVariant [QtSql]	_ZN9QSqlField17setRequiredStatusENS_14RequiredStatusE [QtSql]
_ZN9QSqlField5clearEv [QtSql]	_ZN9QSqlField7setNameERK7QString [QtSql]
_ZN9QSqlField7setTypeEN8QVariant4TypeE [QtSql]	_ZN9QSqlField8setValueERK8QVariant [QtSql]
_ZN9QSqlField9setLengthEi [QtSql]	_ZN9QSqlFieldC1ERK7QStringN8QVariant4TypeE [QtSql]
ZN9QSqlFieldC1ERKS [QtSql]	_ZN9QSqlFieldC2ERK7QStringN8QVariant4TypeE [QtSql]

ZN9QSqlFieldC2ERKS [QtSql]	_ZN9QSqlFieldD1Ev [QtSql]
_ZN9QSqlFieldD2Ev [QtSql]	_ZN9QSqlFieldaSERKS_ [QtSql]
_ZN9QSqlIndex13setCursorNameERK7QString [QtSql]	_ZN9QSqlIndex13setDescendingEib [QtSql]
_ZN9QSqlIndex6appendERK9QSqlField [QtSql]	_ZN9QSqlIndex6appendERK9QSqlFieldb [QtSql]
_ZN9QSqlIndex7setNameERK7QString [QtSql]	_ZN9QSqlIndexC1ERK7QStringS2_ [QtSql]
ZN9QSqlIndexC1ERKS [QtSql]	_ZN9QSqlIndexC2ERK7QStringS2_ [QtSql]
ZN9QSqlIndexC2ERKS [QtSql]	_ZN9QSqlIndexD1Ev [QtSql]
_ZN9QSqlIndexD2Ev [QtSql]	_ZN9QSqlIndexaSERKS_ [QtSql]
_ZN9QSqlQuery12addBindValueERK8QVariant6QFlagsIN4QSql13ParamTypeFlagEE [QtSql]	_ZN9QSqlQuery14setForwardOnlyEb [QtSql]
_ZN9QSqlQuery4execERK7QString [QtSql]	_ZN9QSqlQuery4execEv [QtSql]
_ZN9QSqlQuery4lastEv [QtSql]	_ZN9QSqlQuery4nextEv [QtSql]
_ZN9QSqlQuery4seekEib [QtSql]	_ZN9QSqlQuery5clearEv [QtSql]
_ZN9QSqlQuery5firstEv [QtSql]	_ZN9QSqlQuery7prepareERK7QString [QtSql]
_ZN9QSqlQuery8previousEv [QtSql]	_ZN9QSqlQuery9bindValueERK7QStringRK8QVariant6QFlagsIN4QSql13ParamTypeFlagEE [QtSql]
_ZN9QSqlQuery9bindValueERK8QVariant6QFlagsIN4QSql13ParamTypeFlagEE [QtSql]	_ZN9QSqlQuery9execBatchENS_18BatchExecutionModeE [QtSql]
_ZN9QSqlQueryC1E12QSqlDatabase [QtSql]	_ZN9QSqlQueryC1EP10QSqlResult [QtSql]
_ZN9QSqlQueryC1ERK7QString12QSqlDatabase [QtSql]	_ZN9QSqlQueryC1ERKS_ [QtSql]
_ZN9QSqlQueryC2E12QSqlDatabase [QtSql]	_ZN9QSqlQueryC2EP10QSqlResult [QtSql]
_ZN9QSqlQueryC2ERK7QString12QSqlDatabase [QtSql]	_ZN9QSqlQueryC2ERKS_ [QtSql]
_ZN9QSqlQueryD1Ev [QtSql]	_ZN9QSqlQueryD2Ev [QtSql]
ZN9QSqlQueryaSERKS [QtSql]	_ZNK10QSqlDriver10metaObjectEv [QtSql]
_ZNK10QSqlDriver11formatValueERK9QSqlFieldb [QtSql]	_ZNK10QSqlDriver11isOpenErrorEv [QtSql]

_ZNK10QSqlDriver12primaryIndexERK7QString [QtSql]	_ZNK10QSqlDriver12sqlStatementENS_13StatementTypeERK7QStringRK10QSqlRecordb [QtSql]
_ZNK10QSqlDriver16escapeIdentifierERK7QStringNS_14IdentifierTypeE [QtSql]	_ZNK10QSqlDriver6handleEv [QtSql]
_ZNK10QSqlDriver6isOpenEv [QtSql]	_ZNK10QSqlDriver6recordERK7QString [QtSql]
_ZNK10QSqlDriver6tablesEN4QSql9TableTypeE [QtSql]	_ZNK10QSqlDriver9lastErrorEv [QtSql]
_ZNK10QSqlRecord11isGeneratedERK7QString [QtSql]	_ZNK10QSqlRecord11isGeneratedEi [QtSql]
_ZNK10QSqlRecord12toStringListERK7QString [QtSql]	_ZNK10QSqlRecord5countEv [QtSql]
_ZNK10QSqlRecord5fieldERK7QString [QtSql]	_ZNK10QSqlRecord5fieldEi [QtSql]
_ZNK10QSqlRecord5valueERK7QString [QtSql]	_ZNK10QSqlRecord5valueEi [QtSql]
_ZNK10QSqlRecord6isNullERK7QString [QtSql]	_ZNK10QSqlRecord6isNullEi [QtSql]
_ZNK10QSqlRecord7indexOfERK7QString [QtSql]	_ZNK10QSqlRecord7isEmptyEv [QtSql]
_ZNK10QSqlRecord8containsERK7QString [QtSql]	_ZNK10QSqlRecord8fieldPtrERK7QString [QtSql]
_ZNK10QSqlRecord8fieldPtrEi [QtSql]	_ZNK10QSqlRecord8toStringERK7QStringS2_ [QtSql]
_ZNK10QSqlRecord9fieldNameEi [QtSql]	_ZNK10QSqlRecordeqERKS_ [QtSql]
_ZNK10QSqlResult10boundValueERK7QString [QtSql]	_ZNK10QSqlResult10boundValueEi [QtSql]
_ZNK10QSqlResult11boundValuesEv [QtSql]	_ZNK10QSqlResult12hasOutValuesEv [QtSql]
_ZNK10QSqlResult12lastInsertIdEv [QtSql]	_ZNK10QSqlResult13bindValueTypeERK7QString [QtSql]
_ZNK10QSqlResult13bindValueTypeEi [QtSql]	_ZNK10QSqlResult13bindingSyntaxEv [QtSql]
_ZNK10QSqlResult13executedQueryEv [QtSql]	_ZNK10QSqlResult13isForwardOnlyEv [QtSql]
_ZNK10QSqlResult14boundValueNameEi [QtSql]	_ZNK10QSqlResult15boundValueCountEv [QtSql]
_ZNK10QSqlResult2atEv [QtSql]	_ZNK10QSqlResult6driverEv [QtSql]

_ZNK10QSqlResult6handleEv [QtSql]	_ZNK10QSqlResult6recordEv [QtSql]
_ZNK10QSqlResult7isValidEv [QtSql]	_ZNK10QSqlResult8isActiveEv [QtSql]
_ZNK10QSqlResult8isSelectEv [QtSql]	_ZNK10QSqlResult9lastErrorEv [QtSql]
_ZNK10QSqlResult9lastQueryEv [QtSql]	_ZNK12QSqlDatabase10driverNameEv [QtSql]
_ZNK12QSqlDatabase10recordInfoERK9QSqlQuery [QtSql]	_ZNK12QSqlDatabase11isOpenErrorEv [QtSql]
_ZNK12QSqlDatabase12databaseNameEv [QtSql]	_ZNK12QSqlDatabase12primaryIndexERK7QString [QtSql]
_ZNK12QSqlDatabase14connectOptionsEv [QtSql]	_ZNK12QSqlDatabase4execERK7QString [QtSql]
_ZNK12QSqlDatabase4portEv [QtSql]	_ZNK12QSqlDatabase6driverEv [QtSql]
_ZNK12QSqlDatabase6isOpenEv [QtSql]	_ZNK12QSqlDatabase6recordERK7QString [QtSql]
_ZNK12QSqlDatabase6recordERK9QSqlQuery [QtSql]	_ZNK12QSqlDatabase6tablesEN4QSql9TableTypeE [QtSql]
_ZNK12QSqlDatabase7isValidEv [QtSql]	_ZNK12QSqlDatabase8hostNameEv [QtSql]
_ZNK12QSqlDatabase8passwordEv [QtSql]	_ZNK12QSqlDatabase8userNameEv [QtSql]
_ZNK12QSqlDatabase9lastErrorEv [QtSql]	_ZNK14QSqlQueryModel10headerDataEiN2Qt11OrientationEi [QtSql]
_ZNK14QSqlQueryModel10metaObjectEv [QtSql]	_ZNK14QSqlQueryModel11columnCountERK11QModelIndex [QtSql]
_ZNK14QSqlQueryModel12canFetchMoreERK11QModelIndex [QtSql]	_ZNK14QSqlQueryModel12indexInQueryERK11QModelIndex [QtSql]
_ZNK14QSqlQueryModel4dataERK11QModelIndexi [QtSql]	_ZNK14QSqlQueryModel5queryEv [QtSql]
_ZNK14QSqlQueryModel6recordEi [QtSql]	_ZNK14QSqlQueryModel6recordEv [QtSql]
_ZNK14QSqlQueryModel8rowCountERK11QModelIndex [QtSql]	_ZNK14QSqlQueryModel9lastErrorEv [QtSql]
_ZNK14QSqlTableModel10fieldIndexERK7QString [QtSql]	_ZNK14QSqlTableModel10headerDataEiN2Qt11OrientationEi [QtSql]
_ZNK14QSqlTableModel10metaObjectEv [QtSql]	_ZNK14QSqlTableModel10primaryKeyEv [QtSql]
_ZNK14QSqlTableModel12editStrategyEv [QtSql]	_ZNK14QSqlTableModel12indexInQueryERK11QModelIndex [QtSql]

_ZNK14QSqlTableModel3orderByClauseEv [QtSql]	_ZNK14QSqlTableModel15selectStatementEv [QtSql]
_ZNK14QSqlTableModel4dataERK11QModelIndex [QtSql]	_ZNK14QSqlTableModel5flagsERK11QModelIndex [QtSql]
_ZNK14QSqlTableModel6filterEv [QtSql]	_ZNK14QSqlTableModel7isDirtyERK11QModelIndex [QtSql]
_ZNK14QSqlTableModel8databaseEv [QtSql]	_ZNK14QSqlTableModel8rowCountERK11QModelIndex [QtSql]
_ZNK14QSqlTableModel9tableNameEv [QtSql]	_ZNK16QSqlDriverPlugin10metaObjectEv [QtSql]
_ZNK24QSqlRelationalTableModel10metaObjectEv [QtSql]	_ZNK24QSqlRelationalTableModel13orderByClauseEv [QtSql]
_ZNK24QSqlRelationalTableModel13relationModelEi [QtSql]	_ZNK24QSqlRelationalTableModel15selectStatementEv [QtSql]
_ZNK24QSqlRelationalTableModel4dataERK11QModelIndex [QtSql]	_ZNK24QSqlRelationalTableModel8relationEi [QtSql]
_ZNK9QSqlError10driverTextEv [QtSql]	_ZNK9QSqlError12databaseTextEv [QtSql]
_ZNK9QSqlError4textEv [QtSql]	_ZNK9QSqlError4typeEv [QtSql]
_ZNK9QSqlError6numberEv [QtSql]	_ZNK9QSqlError7isValidEv [QtSql]
_ZNK9QSqlField10isReadOnlyEv [QtSql]	_ZNK9QSqlField11isAutoValueEv [QtSql]
_ZNK9QSqlField11isGeneratedEv [QtSql]	_ZNK9QSqlField12defaultValueEv [QtSql]
_ZNK9QSqlField14requiredStatusEv [QtSql]	_ZNK9QSqlField4nameEv [QtSql]
_ZNK9QSqlField4typeEv [QtSql]	_ZNK9QSqlField6isNullEv [QtSql]
_ZNK9QSqlField6lengthEv [QtSql]	_ZNK9QSqlField6typeIDev [LSB]
_ZNK9QSqlField7isValidEv [QtSql]	_ZNK9QSqlField9precisionEv [QtSql]
ZNK9QSqlFieldeqERKS [QtSql]	_ZNK9QSqlIndex12isDescendingEi [QtSql]
_ZNK9QSqlIndex12toStringListERK7QStringb [QtSql]	_ZNK9QSqlIndex8toStringERK7QStringS2_b [QtSql]
_ZNK9QSqlQuery10boundValueERK7QString [QtSql]	_ZNK9QSqlQuery10boundValueEi [QtSql]
_ZNK9QSqlQuery11boundValuesEv [QtSql]	_ZNK9QSqlQuery12lastInsertIdEv [QtSql]
_ZNK9QSqlQuery13executedQueryEv [QtSql]	_ZNK9QSqlQuery13isForwardOnlyEv [QtSql]

_ZNK9QSqlQuery15numRowsAffectedEv [QtSql]	_ZNK9QSqlQuery2atEv [QtSql]
_ZNK9QSqlQuery4sizeEv [QtSql]	_ZNK9QSqlQuery5valueEi [QtSql]
_ZNK9QSqlQuery6driverEv [QtSql]	_ZNK9QSqlQuery6isNullEi [QtSql]
_ZNK9QSqlQuery6recordEv [QtSql]	_ZNK9QSqlQuery6resultEv [QtSql]
_ZNK9QSqlQuery7isValidEv [QtSql]	_ZNK9QSqlQuery8isActiveEv [QtSql]
_ZNK9QSqlQuery8isSelectEv [QtSql]	_ZNK9QSqlQuery9lastErrorEv [QtSql]
_ZNK9QSqlQuery9lastQueryEv [QtSql]	_Zls6QDebugRK10QSqlRecord [QtSql]
_Zls6QDebugRK12QSqlDatabase [QtSql]	_Zls6QDebugRK9QSqlError [QtSql]
_Zls6QDebugRK9QSqlField [QtSql]	

An LSB conforming implementation shall provide the generic data interfaces for Qt4 SQL specified in Table 18-575, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-575 libQtSql - Qt4 SQL Data Interfaces

_ZN10QSqlDriver16staticMetaObjectE [QtSql]	_ZN12QSqlDatabase17defaultConnectionE [QtSql]
_ZN14QSqlQueryModel16staticMetaObjectE [QtSql]	_ZN14QSqlTableModel16staticMetaObjectE [QtSql]
_ZN16QSqlDriverPlugin16staticMetaObjectE [QtSql]	_ZN24QSqlRelationalTableModel16staticMetaObjectE [QtSql]
_ZTI10QSqlDriver [CXXABI-1.86]	_ZTI10QSqlResult [CXXABI-1.86]
_ZTI14QSqlQueryModel [CXXABI-1.86]	_ZTI14QSqlTableModel [CXXABI-1.86]
_ZTI16QSqlCachedResult [CXXABI-1.86]	_ZTI16QSqlDriverPlugin [CXXABI-1.86]
_ZTI17QFactoryInterface [CXXABI-1.86]	_ZTI24QSqlRelationalTableModel [CXXABI-1.86]
_ZTI26QSqlDriverFactoryInterface [CXXABI-1.86]	_ZTV10QSqlDriver [CXXABI-1.86]
_ZTV10QSqlResult [CXXABI-1.86]	_ZTV14QSqlQueryModel [CXXABI-1.86]
_ZTV14QSqlTableModel [CXXABI-1.86]	_ZTV16QSqlCachedResult [CXXABI-1.86]
_ZTV16QSqlDriverPlugin [CXXABI-1.86]	_ZTV17QFactoryInterface [CXXABI-1.86]

_ZTV24QSqlRelationalTableModel [CXXABI-1.86]	_ZTV26QSqlDriverFactoryInterface [CXXABI-1.86]

18.14 Data Definitions for libQtSql

This section defines global identifiers and their values that are associated with interfaces contained in libQtSql. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

18.14.1 QtSql/qsq.h

```
enum QSql::Location {
    AfterLastRow = -2,
    AfterLast = -2,
    BeforeFirstRow = -1,
    BeforeFirst = -1
};
enum QSql::ParamTypeFlag {
    In = 1,
    Out = 2,
    InOut = 3,
    Binary = 4
};
class QFlags < QSql::ParamTypeFlag >;
typedef class QFlags < QSql::ParamTypeFlag > QSql::ParamType;
enum QSql::TableType {
    Tables = 1,
    SystemTables = 2,
    Views = 4,
    AllTables = 255
};
enum QSql::Op {
    None = -1,
    Insert = 0,
    Update = 1,
    Delete = 2
};
enum QSql::Confirm {
    Cancel = -1,
    No = 0,
    Yes = 1
};
```

18.14.2 QtSql/qsqldatabase.h

```
class QSqlDriverCreatorBase;
class QSqlDatabase;
```

18.14.3 QSql/qsqldriver.h

```

class QSqlDriver;
enum QSqlDriver::DriverFeature {
    Transactions = 0,
    QuerySize = 1,
    BLOB = 2,
    Unicode = 3,
    PreparedQueries = 4,
    NamedPlaceholders = 5,
    PositionalPlaceholders = 6,
    LastInsertId = 7,
    BatchOperations = 8
};
enum QSqlDriver::StatementType {
    WhereStatement = 0,
    SelectStatement = 1,
    UpdateStatement = 2,
    InsertStatement = 3,
    DeleteStatement = 4
};
enum QSqlDriver::IdentifierType {
    FieldName = 0,
    TableName = 1
};

```

18.14.4 QSql/qsqldriverplugin.h

```

struct QSqlDriverFactoryInterface;
class QSqlDriverPlugin;

```

18.14.5 QSql/qsqlerror.h

```

class QSqlError;
enum QSqlError::ErrorType {
    NoError = 0,
    None = 0,
    ConnectionError = 1,
    Connection = 1,
    StatementError = 2,
    Statement = 2,
    TransactionError = 3,
    Transaction = 3,
    UnknownError = 4,
    Unknown = 4
};

```

18.14.6 QSql/qsqlfield.h

```

class QSqlField;
enum QSqlField::RequiredStatus {
    Unknown = -1,
    Optional = 0,
    Required = 1
};

```

18.14.7 QSql/qsqlindex.h

```

class QSqlIndex;

```

18.14.8 QSql/qsqlquery.h

```
class QSqlQuery;
```

18.14.9 QSql/qsqlquerymodel.h

```
class QSqlQueryModel;
```

18.14.10 QSql/qsqlrecord.h

```
class QSqlRecord;
```

18.14.11 QSql/qsqlrelationaltablemodel.h

```
typedef enum QtValidLicenseForSqlModule QSqlModule;
class QSqlRelation;
class QSqlRelationalTableModel;
```

18.14.12 QSql/qsqlresult.h

```
class QSqlResult;
enum QSqlResult::BindingSyntax {
    PositionalBinding = 0,
    BindByPosition = 0,
    NamedBinding = 1,
    BindByName = 1
};
enum QSqlResult::VirtualHookOperation {
    BatchOperation = 0
};
```

18.14.13 QSql/qsqltablemodel.h

```
class QSqlTableModel;
enum QSqlTableModel::EditStrategy {
    OnFieldChange = 0,
    OnRowChange = 1,
    OnManualSubmit = 2
};
```

18.15 Interface Definitions for libQtSql

The interfaces defined on the following pages are included in libQtSql and are defined by this specification. Unless otherwise noted, these interfaces shall be included in the source standard.

Other interfaces listed in Section 18.13 shall behave as described in the referenced base document.

_ZN10QSqlResult9execBatchEb**Name**

`QSqlResult::execBatch` — executes a prepared query in batch mode

Synopsis

```
#include <QtSql/qsqresult.h>
bool QSqlResult::execBatch (bool arrayBind);
```

Description

The `QSqlResult::execBatch()` function shall execute a prepared query in batch mode if the driver supports it, otherwise it should emulate a batch execution (for example, using `bindValue()` and `exec()` functions).

The `QSqlDriver::hasFeature()` can be used to find out whether a driver supports batch execution.

Batch execution can be faster for large amounts of data since it reduces network roundtrips.

For batch executions, bound values have to be provided as lists of variants (`QVariantList`). Each list must contain values of the same type. All lists must contain equal amount of values (rows).

NULL values should be passed in as typed `QVariants`, for example `QVariant(QVariant::Int)` should be used for an integer NULL value.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

_ZNK9QSqlField6typeIDEv**Name**

`QSqlField::typeID` — get the type identifier for the field

Synopsis

```
#include <QtSql/qsqfield.h>
int QSqlField::typeID(void);
```

Description

The `QSqlField::typeID()` function shall return the type ID for the field represented by the calling object.

If the returned value is negative, it means that the information is not available from the database.

This function is not intended to be used by applications directly, but may appear among application dependencies as a result of usage of other functions and macros.

18.16 Interfaces for libQtSvg

Table 18-576 defines the library name and shared object name for the libQtSvg library

Table 18-576 libQtSvg Definition

Library:	libQtSvg
SONAME:	libQtSvg.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI

[QtSvg] QtSvg 4.2.0

[QtXml] QtXml 4.2.0

18.16.1 Qt4 Svg

18.16.1.1 Class data for QSvgRenderer

The virtual table for the QSvgRenderer class is described by Table 18-577

Table 18-577 Primary vtable for QSvgRenderer

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSvgRenderer
vfunc[0]:	QSvgRenderer::metaObject() const
vfunc[1]:	QSvgRenderer::qt_metacast(char const*)
vfunc[2]:	QSvgRenderer::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSvgRenderer::~QSvgRenderer()
vfunc[4]:	QSvgRenderer::~QSvgRenderer()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QSvgRenderer class is described by Table 18-578

Table 18-578 typeinfo for QSvgRenderer

Base Vtable	vtable for __cxxabiv1::__si_class_type_info

Name	typeinfo name for QSvgRenderer
basetype:	typeinfo for QObject

18.16.1.2 Class data for QSvgWidget

The virtual table for the QSvgWidget class is described by Table 18-579

Table 18-579 Primary vtable for QSvgWidget

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QSvgWidget
vfunc[0]:	QSvgWidget::metaObject() const
vfunc[1]:	QSvgWidget::qt_metacast(char const*)
vfunc[2]:	QSvgWidget::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QSvgWidget::~QSvgWidget()
vfunc[4]:	QSvgWidget::~~QSvgWidget()
vfunc[5]:	QWidget::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QWidget::devType() const
vfunc[13]:	QWidget::setVisible(bool)
vfunc[14]:	QSvgWidget::sizeHint() const
vfunc[15]:	QWidget::minimumSizeHint() const
vfunc[16]:	QWidget::heightForWidth(int) const
vfunc[17]:	QWidget::paintEngine() const
vfunc[18]:	QWidget::mousePressEvent(QMouseEvent*)
vfunc[19]:	QWidget::mouseReleaseEvent(QMouseEvent*)
vfunc[20]:	QWidget::mouseDoubleClickEvent(QMouseEvent*)

vfunc[21]:	QWidget::mouseMoveEvent(QMouseEvent*)
vfunc[22]:	QWidget::wheelEvent(QWheelEvent*)
vfunc[23]:	QWidget::keyPressEvent(QKeyEvent*)
vfunc[24]:	QWidget::keyReleaseEvent(QKeyEvent*)
vfunc[25]:	QWidget::focusInEvent(QFocusEvent*)
vfunc[26]:	QWidget::focusOutEvent(QFocusEvent*)
vfunc[27]:	QWidget::enterEvent(QEvent*)
vfunc[28]:	QWidget::leaveEvent(QEvent*)
vfunc[29]:	QSvgWidget::paintEvent(QPaintEvent*)
vfunc[30]:	QWidget::moveEvent(QMoveEvent*)
vfunc[31]:	QWidget::resizeEvent(QResizeEvent*)
vfunc[32]:	QWidget::closeEvent(QCloseEvent*)
vfunc[33]:	QWidget::contextMenuEvent(QContextMenuEvent*)
vfunc[34]:	QWidget::tabletEvent(QTabletEvent*)
vfunc[35]:	QWidget::actionEvent(QActionEvent*)
vfunc[36]:	QWidget::dragEnterEvent(QDragEnterEvent*)
vfunc[37]:	QWidget::dragMoveEvent(QDragMoveEvent*)
vfunc[38]:	QWidget::dragLeaveEvent(QDragLeaveEvent*)
vfunc[39]:	QWidget::dropEvent(QDropEvent*)
vfunc[40]:	QWidget::showEvent(QShowEvent*)
vfunc[41]:	QWidget::hideEvent(QHideEvent*)
vfunc[42]:	QWidget::x11Event(_XEvent*)
vfunc[43]:	QWidget::changeEvent(QEvent*)
vfunc[44]:	QWidget::metric(QPaintDevice::PaintDeviceMetric) const

vfunc[45]:	QWidget::inputMethodEvent(QInputMethodEvent*)
vfunc[46]:	QWidget::inputMethodQuery(Qt::InputMethodQuery) const
vfunc[47]:	QWidget::focusNextPrevChild(bool)
vfunc[48]:	QWidget::styleChange(QStyle&)
vfunc[49]:	QWidget::enabledChange(bool)
vfunc[50]:	QWidget::paletteChange(QPalette const&)
vfunc[51]:	QWidget::fontChange(QFont const&)
vfunc[52]:	QWidget::windowActivationChange(bool)
vfunc[53]:	QWidget::languageChange()

The Run Time Type Information for the QSvgWidget class is described by Table 18-580

Table 18-580 typeinfo for QSvgWidget

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QSvgWidget
basetype:	typeinfo for QWidget

18.16.1.3 Interfaces for Qt4 Svg

An LSB conforming implementation shall provide the generic functions for Qt4 Svg specified in Table 18-581, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-581 libQtSvg - Qt4 Svg Function Interfaces

_ZN10QSvgWidget10paintEventEP11QPaintEvent [QtSvg]	_ZN10QSvgWidget11qt_metacallEN11QMetaObject4CallEiPPv [QtSvg]
_ZN10QSvgWidget11qt_metacastEPKc [QtSvg]	_ZN10QSvgWidget4loadERK10QByteArray [QtSvg]
_ZN10QSvgWidget4loadERK7QString [QtSvg]	_ZN10QSvgWidgetC1EP7QWidget [QtSvg]
_ZN10QSvgWidgetC1ERK7QStringP7QWidget [QtSvg]	_ZN10QSvgWidgetC2EP7QWidget [QtSvg]
_ZN10QSvgWidgetC2ERK7QStringP7QWidget [QtSvg]	_ZN10QSvgWidgetD0Ev [QtSvg]
_ZN10QSvgWidgetD1Ev [QtSvg]	_ZN10QSvgWidgetD2Ev [QtSvg]
_ZN12QSvgRenderer10setViewBoxERK6QRect [QtSvg]	_ZN12QSvgRenderer10setViewBoxERK6QRectF [QtXml]

_ZN12QSvgRenderer11qt_metacallE N11QMetaObject4CallEiPPv [QtSvg]	_ZN12QSvgRenderer11qt_metacastE PKc [QtSvg]
_ZN12QSvgRenderer13repaintNeedE dEv [QtSvg]	_ZN12QSvgRenderer15setCurrentFr ameEi [QtSvg]
_ZN12QSvgRenderer18setFramesPer SecondEi [QtSvg]	_ZN12QSvgRenderer4loadERK10QB yteArray [QtSvg]
_ZN12QSvgRenderer4loadERK7QStr ing [QtSvg]	_ZN12QSvgRenderer6renderEP8QP ainter [QtSvg]
_ZN12QSvgRenderer6renderEP8QP ainterRK6QRectF [QtXml]	_ZN12QSvgRenderer6renderEP8QP ainterRK7QStringRK6QRectF [QtXml]
_ZN12QSvgRendererC1EP7QObject [QtSvg]	_ZN12QSvgRendererC1ERK10QByte ArrayP7QObject [QtSvg]
_ZN12QSvgRendererC1ERK7QStrin gP7QObject [QtSvg]	_ZN12QSvgRendererC2EP7QObject [QtSvg]
_ZN12QSvgRendererC2ERK10QByte ArrayP7QObject [QtSvg]	_ZN12QSvgRendererC2ERK7QStrin gP7QObject [QtSvg]
_ZN12QSvgRendererD0Ev [QtSvg]	_ZN12QSvgRendererD1Ev [QtSvg]
_ZN12QSvgRendererD2Ev [QtSvg]	_ZN16QGraphicsSvgItem11qt_metac allEN11QMetaObject4CallEiPPv [QtXml]
_ZN16QGraphicsSvgItem11qt_metac astEPKc [QtXml]	_ZN16QGraphicsSvgItem12setEleme ntIdERK7QString [QtXml]
_ZN16QGraphicsSvgItem17setCachi ngEnabledEb [QtXml]	_ZN16QGraphicsSvgItem17setShare dRendererEP12QSvgRenderer [QtXml]
_ZN16QGraphicsSvgItem19setMaxi mumCacheSizeERK5QSize [QtXml]	_ZN16QGraphicsSvgItem5paintEP8 QPainterPK24QStyleOptionGraphics ItemP7QWidget [QtXml]
_ZN16QGraphicsSvgItemC1EP13QG raphicsItem [QtXml]	_ZN16QGraphicsSvgItemC1ERK7QS tringP13QGraphicsItem [QtXml]
_ZN16QGraphicsSvgItemC2EP13QG raphicsItem [QtXml]	_ZN16QGraphicsSvgItemC2ERK7QS tringP13QGraphicsItem [QtXml]
_ZNK10QSvgWidget10metaObjectEv [QtSvg]	_ZNK10QSvgWidget8rendererEv [QtSvg]
_ZNK10QSvgWidget8sizeHintEv [QtSvg]	_ZNK12QSvgRenderer10metaObject Ev [QtSvg]
_ZNK12QSvgRenderer11defaultSize Ev [QtSvg]	_ZNK12QSvgRenderer12currentFra meEv [QtSvg]
_ZNK12QSvgRenderer13elementExis tsERK7QString [QtXml]	_ZNK12QSvgRenderer15boundsOnE lementERK7QString [QtXml]
_ZNK12QSvgRenderer15framesPerS econdEv [QtSvg]	_ZNK12QSvgRenderer16matrixForEl ementERK7QString [QtXml]

_ZNK12QSvgRenderer17animationDurationEv [QtSvg]	_ZNK12QSvgRenderer7isValidEv [QtSvg]
_ZNK12QSvgRenderer7viewBoxEv [QtSvg]	_ZNK12QSvgRenderer8animatedEv [QtSvg]
_ZNK12QSvgRenderer8viewBoxFEv [QtXml]	_ZNK16QGraphicsSvgItem10metaObjectEv [QtXml]
_ZNK16QGraphicsSvgItem12boundingRectEv [QtXml]	_ZNK16QGraphicsSvgItem16isCachingEnabledEv [QtXml]
_ZNK16QGraphicsSvgItem16maximumCacheSizeEv [QtXml]	_ZNK16QGraphicsSvgItem4typeEv [QtXml]
_ZNK16QGraphicsSvgItem8rendererEv [QtXml]	_ZNK16QGraphicsSvgItem9elementIdEv [QtXml]

An LSB conforming implementation shall provide the generic data interfaces for Qt4 Svg specified in Table 18-582, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-582 libQtSvg - Qt4 Svg Data Interfaces

_ZN10QSvgWidget16staticMetaObjectE [QtSvg]	_ZN12QSvgRenderer16staticMetaObjectE [QtSvg]
_ZN16QGraphicsSvgItem16staticMetaObjectE [QtXml]	_ZTI10QSvgWidget [CXXABI-1.86]
_ZTI12QSvgRenderer [CXXABI-1.86]	_ZTI16QGraphicsSvgItem [CXXABI-1.86]
_ZTI16QSvgTinyDocument [CXXABI-1.86]	_ZTV10QSvgWidget [CXXABI-1.86]
_ZTV12QSvgRenderer [CXXABI-1.86]	_ZTV16QGraphicsSvgItem [CXXABI-1.86]
_ZTV16QSvgTinyDocument [CXXABI-1.86]	

18.17 Data Definitions for libQtSvg

This section defines global identifiers and their values that are associated with interfaces contained in libQtSvg. These definitions are organized into groups that correspond to system headers. This convention is used as a convenience for the reader, and does not imply the existence of these headers, or their content. Where an interface is defined as requiring a particular system header file all of the data definitions for that system header file presented here shall be in effect.

This section gives data definitions to promote binary application portability, not to repeat source interface definitions available elsewhere. System providers and application developers should use this ABI to supplement - not to replace - source interface definition specifications.

This specification uses the ISO C (1999) C Language as the reference programming language, and data definitions are specified in ISO C format. The C language is used here as a convenient notation. Using a C language description of these data objects does not preclude their use by other programming languages.

18.17.1 QtSvg/qsvgrenderer.h

```
class QSvgRenderer;
```

18.17.2 QtSvg/qsvgwidget.h

```
typedef enum QtValidLicenseForSvgModule QtSvgModule;  
class QSvgWidget;
```

18.18 Interfaces for libQtNetwork

Table 18-583 defines the library name and shared object name for the libQtNetwork library

Table 18-583 libQtNetwork Definition

Library:	libQtNetwork
SONAME:	libQtNetwork.so.4

The behavior of the interfaces in this library is specified by the following specifications:

[CXXABI-1.86] Itanium™ C++ ABI
[LSB] This Specification
[QtNetwork] QtNetwork 4.2.0
[QtXml] QtXml 4.2.0

18.18.1 Qt4 Network

18.18.1.1 Class data for QHttpHeader

The virtual table for the QHttpHeader class is described by Table 18-584

Table 18-584 Primary vtable for QHttpHeader

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHttpHeader
vfunc[0]:	QHttpHeader::~~QHttpHeader()
vfunc[1]:	QHttpHeader::~~QHttpHeader()
vfunc[2]:	QHttpHeader::toString() const
vfunc[3]:	__cxa_pure_virtual
vfunc[4]:	__cxa_pure_virtual
vfunc[5]:	QHttpHeader::parseLine(QString const&, int)

The Run Time Type Information for the QHttpHeader class is described by Table 18-585

Table 18-585 typeinfo for QHttpHeader

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QHttpHeader

18.18.1.2 Class data for QHttpResponseHeader

The virtual table for the QHttpResponseHeader class is described by Table 18-586

Table 18-586 Primary vtable for QHttpResponseHeader

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHttpResponseHeader
vfunc[0]:	NULL or QHttpResponseHeader::~~QHttpResponseHeader()
vfunc[1]:	NULL or QHttpResponseHeader::~~QHttpResponseHeader()
vfunc[2]:	QHttpResponseHeader::toString() const
vfunc[3]:	QHttpResponseHeader::majorVersion() const
vfunc[4]:	QHttpResponseHeader::minorVersion() const
vfunc[5]:	QHttpResponseHeader::parseLine(QString const&, int)

The Run Time Type Information for the QHttpResponseHeader class is described by Table 18-587

Table 18-587 typeinfo for QHttpResponseHeader

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHttpResponseHeader
basetype:	typeinfo for QHttpHeader

18.18.1.3 Class data for QHttpRequestHeader

The virtual table for the QHttpRequestHeader class is described by Table 18-588

Table 18-588 Primary vtable for QHttpRequestHeader

Base Offset	0

Virtual Base Offset	0
RTTI	typeinfo for QHttpRequestHeader
vfunc[0]:	NULL or QHttpRequestHeader::~~QHttpRequestHeader()
vfunc[1]:	NULL or QHttpRequestHeader::~~QHttpRequestHeader()
vfunc[2]:	QHttpRequestHeader::toString() const
vfunc[3]:	QHttpRequestHeader::majorVersion() const
vfunc[4]:	QHttpRequestHeader::minorVersion() const
vfunc[5]:	QHttpRequestHeader::parseLine(QString const&, int)

The Run Time Type Information for the QHttpRequestHeader class is described by Table 18-589

Table 18-589 typeinfo for QHttpRequestHeader

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHttpRequestHeader
basetype:	typeinfo for QHttpHeader

18.18.1.4 Class data for QHttp

The virtual table for the QHttp class is described by Table 18-590

Table 18-590 Primary vtable for QHttp

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QHttp
vfunc[0]:	QHttp::metaObject() const
vfunc[1]:	QHttp::qt_metacast(char const*)
vfunc[2]:	QHttp::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QHttp::~~QHttp()
vfunc[4]:	QHttp::~~QHttp()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QHttp class is described by Table 18-591

Table 18-591 typeinfo for QHttp

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QHttp
basetype:	typeinfo for QObject

18.18.1.5 Class data for QAbstractSocket

The virtual table for the QAbstractSocket class is described by Table 18-592

Table 18-592 Primary vtable for QAbstractSocket

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QAbstractSocket
vfunc[0]:	QAbstractSocket::metaObject() const
vfunc[1]:	QAbstractSocket::qt_metacast(char const*)
vfunc[2]:	QAbstractSocket::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QAbstractSocket::~QAbstractSocket()
vfunc[4]:	QAbstractSocket::~QAbstractSocket()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QAbstractSocket::isSequential() const
vfunc[13]:	QIODevice::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QAbstractSocket::close()
vfunc[15]:	QIODevice::pos() const
vfunc[16]:	QIODevice::size() const
vfunc[17]:	QIODevice::seek(long long)
vfunc[18]:	QAbstractSocket::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QAbstractSocket::bytesAvailable() const
vfunc[21]:	QAbstractSocket::bytesToWrite() const
vfunc[22]:	QAbstractSocket::canReadLine() const
vfunc[23]:	QAbstractSocket::waitForReadyRead(int)
vfunc[24]:	QAbstractSocket::waitForBytesWritten(int)
vfunc[25]:	QAbstractSocket::readData(char*, long long)
vfunc[26]:	QAbstractSocket::readLineData(char*, long long)
vfunc[27]:	QAbstractSocket::writeData(char const*, long long)

The Run Time Type Information for the QAbstractSocket class is described by Table 18-593

Table 18-593 typeinfo for QAbstractSocket

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QAbstractSocket
basetype:	typeinfo for QIODevice

18.18.1.6 Class data for QUrlInfo

The virtual table for the QUrlInfo class is described by Table 18-594

Table 18-594 Primary vtable for QUrlInfo

Base Offset	0
Virtual Base Offset	0

RTTI	typeinfo for QUrlInfo
vfunc[0]:	QUrlInfo::~~QUrlInfo()
vfunc[1]:	QUrlInfo::~~QUrlInfo()
vfunc[2]:	QUrlInfo::setName(QString const&)
vfunc[3]:	QUrlInfo::setDir(bool)
vfunc[4]:	QUrlInfo::setFile(bool)
vfunc[5]:	QUrlInfo::setSymLink(bool)
vfunc[6]:	QUrlInfo::setOwner(QString const&)
vfunc[7]:	QUrlInfo::setGroup(QString const&)
vfunc[8]:	QUrlInfo::setSize(long long)
vfunc[9]:	QUrlInfo::setWritable(bool)
vfunc[10]:	QUrlInfo::setReadable(bool)
vfunc[11]:	QUrlInfo::setPermissions(int)
vfunc[12]:	QUrlInfo::setLastModified(QDateTime const&)

The Run Time Type Information for the QUrlInfo class is described by Table 18-595

Table 18-595 typeinfo for QUrlInfo

Base Vtable	vtable for __cxxabiv1::__class_type_info
Name	typeinfo name for QUrlInfo

18.18.1.7 Class data for QFtp

The virtual table for the QFtp class is described by Table 18-596

Table 18-596 Primary vtable for QFtp

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QFtp
vfunc[0]:	QFtp::metaObject() const
vfunc[1]:	QFtp::qt_metacast(char const*)
vfunc[2]:	QFtp::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QFtp::~~QFtp()
vfunc[4]:	QFtp::~~QFtp()
vfunc[5]:	QObject::event(QEvent*)

vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

The Run Time Type Information for the QFtp class is described by Table 18-597

Table 18-597 typeinfo for QFtp

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QFtp
basetype:	typeinfo for QObject

18.18.1.8 Class data for QTcpServer

The virtual table for the QTcpServer class is described by Table 18-598

Table 18-598 Primary vtable for QTcpServer

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTcpServer
vfunc[0]:	QTcpServer::metaObject() const
vfunc[1]:	QTcpServer::qt_metacast(char const*)
vfunc[2]:	QTcpServer::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTcpServer::~QTcpServer()
vfunc[4]:	QTcpServer::~~QTcpServer()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)

vfunc[12]:	QTcpServer::hasPendingConnections () const
vfunc[13]:	QTcpServer::nextPendingConnection ()
vfunc[14]:	QTcpServer::incomingConnection(int)

The Run Time Type Information for the QTcpServer class is described by Table 18-599

Table 18-599 typeinfo for QTcpServer

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTcpServer
basetype:	typeinfo for QObject

18.18.1.9 Class data for QUdpSocket

The virtual table for the QUdpSocket class is described by Table 18-600

Table 18-600 Primary vtable for QUdpSocket

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QUdpSocket
vfunc[0]:	QUdpSocket::metaObject() const
vfunc[1]:	QUdpSocket::qt_metacast(char const*)
vfunc[2]:	QUdpSocket::qt_metacall(QMetaObj ect::Call, int, void**)
vfunc[3]:	QUdpSocket::~~QUdpSocket()
vfunc[4]:	QUdpSocket::~~QUdpSocket()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractSocket::isSequential() const

vfunc[13]:	QIODevice::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QAbstractSocket::close()
vfunc[15]:	QIODevice::pos() const
vfunc[16]:	QIODevice::size() const
vfunc[17]:	QIODevice::seek(long long)
vfunc[18]:	QAbstractSocket::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QAbstractSocket::bytesAvailable() const
vfunc[21]:	QAbstractSocket::bytesToWrite() const
vfunc[22]:	QAbstractSocket::canReadLine() const
vfunc[23]:	QAbstractSocket::waitForReadyRead(int)
vfunc[24]:	QAbstractSocket::waitForBytesWritten(int)
vfunc[25]:	QAbstractSocket::readData(char*, long long)
vfunc[26]:	QAbstractSocket::readLineData(char*, long long)
vfunc[27]:	QAbstractSocket::writeData(char const*, long long)

The Run Time Type Information for the QUdpSocket class is described by Table 18-601

Table 18-601 typeinfo for QUdpSocket

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QUdpSocket
basetype:	typeinfo for QAbstractSocket

18.18.1.10 Class data for QTcpSocket

The virtual table for the QTcpSocket class is described by Table 18-602

Table 18-602 Primary vtable for QTcpSocket

Base Offset	0
Virtual Base Offset	0
RTTI	typeinfo for QTcpSocket

vfunc[0]:	QTcpSocket::metaObject() const
vfunc[1]:	QTcpSocket::qt_metacast(char const*)
vfunc[2]:	QTcpSocket::qt_metacall(QMetaObject::Call, int, void**)
vfunc[3]:	QTcpSocket::~QTcpSocket()
vfunc[4]:	QTcpSocket::~QTcpSocket()
vfunc[5]:	QObject::event(QEvent*)
vfunc[6]:	QObject::eventFilter(QObject*, QEvent*)
vfunc[7]:	QObject::timerEvent(QTimerEvent*)
vfunc[8]:	QObject::childEvent(QChildEvent*)
vfunc[9]:	QObject::customEvent(QEvent*)
vfunc[10]:	QObject::connectNotify(char const*)
vfunc[11]:	QObject::disconnectNotify(char const*)
vfunc[12]:	QAbstractSocket::isSequential() const
vfunc[13]:	QIODevice::open(QFlags<QIODevice::OpenModeFlag>)
vfunc[14]:	QAbstractSocket::close()
vfunc[15]:	QIODevice::pos() const
vfunc[16]:	QIODevice::size() const
vfunc[17]:	QIODevice::seek(long long)
vfunc[18]:	QAbstractSocket::atEnd() const
vfunc[19]:	QIODevice::reset()
vfunc[20]:	QAbstractSocket::bytesAvailable() const
vfunc[21]:	QAbstractSocket::bytesToWrite() const
vfunc[22]:	QAbstractSocket::canReadLine() const
vfunc[23]:	QAbstractSocket::waitForReadyRead(int)
vfunc[24]:	QAbstractSocket::waitForBytesWritten(int)
vfunc[25]:	QAbstractSocket::readData(char*, long long)
vfunc[26]:	QAbstractSocket::readLineData(char*, long long)

vfunc[27]:	QAbstractSocket::writeData(char const*, long long)

The Run Time Type Information for the QTcpSocket class is described by Table 18-603

Table 18-603 typeinfo for QTcpSocket

Base Vtable	vtable for __cxxabiv1::__si_class_type_info
Name	typeinfo name for QTcpSocket
basetype:	typeinfo for QAbstractSocket

18.18.1.11 Interfaces for Qt4 Network

An LSB conforming implementation shall provide the generic functions for Qt4 Network specified in Table 18-604, with the full mandatory functionality as described in the referenced underlying specification.

Table 18-604 libQtNetwork - Qt4 Network Function Interfaces

_ZN5qHashRK12QHostAddress [QtXml]	_ZN10QTcpServer11qt_metacallEN11QMetaObject4CallEiPPv [QtNetwork]
_ZN10QTcpServer11qt_metacastEPKc [QtNetwork]	_ZN10QTcpServer13newConnectionEv [QtNetwork]
_ZN10QTcpServer18incomingConnectionEi [QtNetwork]	_ZN10QTcpServer19setSocketDescriptorEi [QtNetwork]
_ZN10QTcpServer20waitForNewConnectionEiPb [QtNetwork]	_ZN10QTcpServer21nextPendingConnectionEv [QtNetwork]
_ZN10QTcpServer24setMaxPendingConnectionsEi [QtNetwork]	_ZN10QTcpServer5closeEv [QtNetwork]
_ZN10QTcpServer6listenERK12QHostAddress [QtNetwork]	_ZN10QTcpServer8setProxyERK13QNetworkProxy [QtNetwork]
_ZN10QTcpServerC1EP7QObject [QtNetwork]	_ZN10QTcpServerC2EP7QObject [QtNetwork]
_ZN10QTcpServerD0Ev [QtNetwork]	_ZN10QTcpServerD1Ev [QtNetwork]
_ZN10QTcpServerD2Ev [QtNetwork]	_ZN10QTcpSocket11qt_metacallEN11QMetaObject4CallEiPPv [QtNetwork]
_ZN10QTcpSocket11qt_metacastEPKc [QtNetwork]	_ZN10QTcpSocketC1EP7QObject [QtNetwork]
_ZN10QTcpSocketC2EP7QObject [QtNetwork]	_ZN10QTcpSocketD0Ev [QtNetwork]
_ZN10QTcpSocketD1Ev [QtNetwork]	_ZN10QTcpSocketD2Ev [QtNetwork]

_ZN10QUdpSocket11qt_metacallEN11QMetaObject4CallEiPPv [QtNetwork]	_ZN10QUdpSocket11qt_metacastEPKc [QtNetwork]
_ZN10QUdpSocket12readDatagramEPcxP12QHostAddressPt [QtNetwork]	_ZN10QUdpSocket13writeDatagramEPKcxRK12QHostAddressst [QtNetwork]
_ZN10QUdpSocket4bindERK12QHostAddressst [QtNetwork]	_ZN10QUdpSocket4bindERK12QHostAddressst6QFlagsINS_8BindFlagEE [QtNetwork]
_ZN10QUdpSocket4bindEt [QtNetwork]	_ZN10QUdpSocket4bindEt6QFlagsINS_8BindFlagEE [QtNetwork]
_ZN10QUdpSocketC1EP7QObject [QtNetwork]	_ZN10QUdpSocketC2EP7QObject [QtNetwork]
_ZN10QUdpSocketD0Ev [QtNetwork]	_ZN10QUdpSocketD1Ev [QtNetwork]
_ZN10QUdpSocketD2Ev [QtNetwork]	_ZN11QHttpHeader11removeValueERK7QString [QtNetwork]
_ZN11QHttpHeader14setContentTyperERK7QString [QtNetwork]	_ZN11QHttpHeader15removeAllValuesERK7QString [QtNetwork]
_ZN11QHttpHeader16setContentLengthEi [QtNetwork]	_ZN11QHttpHeader5parseERK7QString [LSB]
ZN11QHttpHeader8addValueERK7QStringS2 [QtNetwork]	_ZN11QHttpHeader8setValidEb [QtNetwork]
ZN11QHttpHeader8setValueERK7QStringS2 [QtNetwork]	_ZN11QHttpHeader9parseLineERK7QStringi [LSB]
_ZN11QHttpHeader9setValuesERK5QListI5QPairI7QStringS2_EE [QtNetwork]	_ZN11QHttpHeaderC1ERK7QString [QtNetwork]
ZN11QHttpHeaderC1ERKS [QtNetwork]	_ZN11QHttpHeaderC1Ev [QtNetwork]
_ZN11QHttpHeaderC2ERK7QString [QtNetwork]	_ZN11QHttpHeaderC2ERKS_ [QtNetwork]
_ZN11QHttpHeaderC2Ev [QtNetwork]	_ZN11QHttpHeaderD0Ev [QtNetwork]
_ZN11QHttpHeaderD1Ev [QtNetwork]	_ZN11QHttpHeaderD2Ev [QtNetwork]
ZN11QHttpHeaderaSERKS [QtNetwork]	_ZN12QHostAddress10setAddressEPK8sockaddr [QtNetwork]
_ZN12QHostAddress10setAddressEPH [QtNetwork]	_ZN12QHostAddress10setAddressEPK12QIPv6Address [QtNetwork]
_ZN12QHostAddress10setAddressEPK7QString [QtNetwork]	_ZN12QHostAddress10setAddressEj [QtNetwork]

_ZN12QHostAddress10setScopeIdERK7QString [QtNetwork]	_ZN12QHostAddress5clearEv [QtNetwork]
_ZN12QHostAddressC1ENS_14SpecialAddressE [QtNetwork]	_ZN12QHostAddressC1EPK8sockaddr [QtNetwork]
_ZN12QHostAddressC1EPh [QtNetwork]	_ZN12QHostAddressC1ERK12QIPv6Address [QtNetwork]
_ZN12QHostAddressC1ERK7QString [QtNetwork]	_ZN12QHostAddressC1ERKS_ [QtNetwork]
_ZN12QHostAddressC1Ej [QtNetwork]	_ZN12QHostAddressC1Ev [QtNetwork]
_ZN12QHostAddressC2ENS_14SpecialAddressE [QtNetwork]	_ZN12QHostAddressC2EPK8sockaddr [QtNetwork]
_ZN12QHostAddressC2EPh [QtNetwork]	_ZN12QHostAddressC2ERK12QIPv6Address [QtNetwork]
_ZN12QHostAddressC2ERK7QString [QtNetwork]	_ZN12QHostAddressC2ERKS_ [QtNetwork]
_ZN12QHostAddressC2Ej [QtNetwork]	_ZN12QHostAddressC2Ev [QtNetwork]
_ZN12QHostAddressD1Ev [QtNetwork]	_ZN12QHostAddressD2Ev [QtNetwork]
_ZN12QHostAddressaSERK7QString [QtNetwork]	_ZN12QHostAddressaSERKS_ [QtNetwork]
_ZN13QNetworkProxy11setHostNameERK7QString [QtNetwork]	_ZN13QNetworkProxy11setPasswordERK7QString [QtNetwork]
_ZN13QNetworkProxy16applicationProxyEv [QtNetwork]	_ZN13QNetworkProxy19setApplicationProxyERKS_ [QtNetwork]
_ZN13QNetworkProxy7setPortEt [QtNetwork]	_ZN13QNetworkProxy7setTypeENS_9ProxyTypeE [QtNetwork]
_ZN13QNetworkProxy7setUserERK7QString [QtNetwork]	_ZN13QNetworkProxyC1ENS_9ProxyTypeERK7QStringtS3_S3_ [QtNetwork]
ZN13QNetworkProxyC1ERKS [QtNetwork]	_ZN13QNetworkProxyC1Ev [QtNetwork]
_ZN13QNetworkProxyC2ENS_9ProxyTypeERK7QStringtS3_S3_ [QtNetwork]	_ZN13QNetworkProxyC2ERKS_ [QtNetwork]
_ZN13QNetworkProxyC2Ev [QtNetwork]	_ZN13QNetworkProxyD1Ev [QtNetwork]
_ZN13QNetworkProxyD2Ev [QtNetwork]	_ZN13QNetworkProxyaSERKS_ [QtNetwork]
_ZN15QAbstractSocket11qt_metacallEN11QMetaObject4CallEiPPv [QtNetwork]	_ZN15QAbstractSocket11qt_metacastEPKc [QtNetwork]

_ZN15QAbstractSocket11setPeerNameERK7QString [QtNetwork]	_ZN15QAbstractSocket11setPeerPortEt [QtNetwork]
_ZN15QAbstractSocket12disconnectdEv [QtNetwork]	_ZN15QAbstractSocket12readLineDataEPcx [QtNetwork]
_ZN15QAbstractSocket12setLocalPortEt [QtNetwork]	_ZN15QAbstractSocket12stateChangedENS_11SocketStateE [QtNetwork]
_ZN15QAbstractSocket13connectToHostERK12QHostAddresst6QFlagsIN9QIODevice12OpenModeFlagEE [QtNetwork]	_ZN15QAbstractSocket13connectToHostERK7QStringt6QFlagsIN9QIODevice12OpenModeFlagEE [QtNetwork]
_ZN15QAbstractSocket14setPeerAddressERK12QHostAddress [QtNetwork]	_ZN15QAbstractSocket14setSocketErrorENS_11SocketErrorE [QtNetwork]
_ZN15QAbstractSocket14setSocketStateENS_11SocketStateE [QtNetwork]	_ZN15QAbstractSocket15setLocalAddressERK12QHostAddress [QtNetwork]
_ZN15QAbstractSocket16connectionClosedEv [QtNetwork]	_ZN15QAbstractSocket16waitForConnectedEi [QtNetwork]
_ZN15QAbstractSocket16waitForReadyReadEi [QtNetwork]	_ZN15QAbstractSocket17setReadBufferSizeEx [QtNetwork]
_ZN15QAbstractSocket18disconnectFromHostEv [QtNetwork]	_ZN15QAbstractSocket19setSocketDescriptorEiNS_11SocketStateE6QFlagsIN9QIODevice12OpenModeFlagEE [QtNetwork]
_ZN15QAbstractSocket19waitForBytesWrittenEi [QtNetwork]	_ZN15QAbstractSocket19waitForDisconnectedEi [QtNetwork]
_ZN15QAbstractSocket20delayedCloseFinishedEv [QtNetwork]	_ZN15QAbstractSocket27connectToHostImplementationERK7QStringt6QFlagsIN9QIODevice12OpenModeFlagEE [QtNetwork]
_ZN15QAbstractSocket32disconnectFromHostImplementationEv [QtNetwork]	_ZN15QAbstractSocket5abortEv [QtNetwork]
_ZN15QAbstractSocket5closeEv [QtNetwork]	_ZN15QAbstractSocket5errorENS_11SocketErrorE [QtNetwork]
_ZN15QAbstractSocket5flushEv [QtNetwork]	_ZN15QAbstractSocket8readDataEPcx [QtNetwork]
_ZN15QAbstractSocket8setProxyERK13QNetworkProxy [QtNetwork]	_ZN15QAbstractSocket9connectedEv [QtNetwork]
_ZN15QAbstractSocket9hostFoundEv [QtNetwork]	_ZN15QAbstractSocket9writeDataEPKcx [QtNetwork]
_ZN15QAbstractSocketC1ENS_10SocketTypeEP7QObject [QtNetwork]	_ZN15QAbstractSocketC2ENS_10SocketTypeEP7QObject [QtNetwork]

_ZN15QAbstractSocketD0Ev [QtNetwork]	_ZN15QAbstractSocketD1Ev [QtNetwork]
_ZN15QAbstractSocketD2Ev [QtNetwork]	_ZN17QNetworkInterface12allAddressesEv [QtXml]
_ZN17QNetworkInterface13allInterfacesEv [QtXml]	_ZN17QNetworkInterface17interfaceFromNameERK7QString [QtXml]
_ZN17QNetworkInterface18interfaceFromIndexEi [QtXml]	_ZN17QNetworkInterfaceC1ERKS_ [QtXml]
_ZN17QNetworkInterfaceC1Ev [QtXml]	_ZN17QNetworkInterfaceC2ERKS_ [QtXml]
_ZN17QNetworkInterfaceC2Ev [QtXml]	_ZN17QNetworkInterfaceD1Ev [QtXml]
_ZN17QNetworkInterfaceD2Ev [QtXml]	_ZN17QNetworkInterfaceaSERKS_ [QtXml]
_ZN18QHttpRequestHeader10setRequestERK7QStringS2_ii [QtNetwork]	_ZN18QHttpRequestHeader9parseLineERK7QStringi [QtNetwork]
_ZN18QHttpRequestHeaderC1ERK7QString [QtNetwork]	_ZN18QHttpRequestHeaderC1ERK7QStringS2_ii [QtNetwork]
ZN18QHttpRequestHeaderC1ERKS [QtNetwork]	_ZN18QHttpRequestHeaderC1Ev [QtNetwork]
_ZN18QHttpRequestHeaderC2ERK7QString [QtNetwork]	_ZN18QHttpRequestHeaderC2ERK7QStringS2_ii [QtNetwork]
ZN18QHttpRequestHeaderC2ERKS [QtNetwork]	_ZN18QHttpRequestHeaderC2Ev [QtNetwork]
ZN18QHttpRequestHeaderaSERKS [QtNetwork]	_ZN19QHttpResponseHeader13setStatusLineEiRK7QStringii [QtNetwork]
_ZN19QHttpResponseHeader9parseLineERK7QStringi [QtNetwork]	_ZN19QHttpResponseHeaderC1ERK7QString [QtNetwork]
ZN19QHttpResponseHeaderC1ERKS [QtNetwork]	_ZN19QHttpResponseHeaderC1EiRK7QStringii [QtNetwork]
_ZN19QHttpResponseHeaderC1Ev [QtNetwork]	_ZN19QHttpResponseHeaderC2ERK7QString [QtNetwork]
ZN19QHttpResponseHeaderC2ERKS [QtNetwork]	_ZN19QHttpResponseHeaderC2EiRK7QStringii [QtNetwork]
_ZN19QHttpResponseHeaderC2Ev [QtNetwork]	_ZN19QHttpResponseHeaderaSERKS_ [QtNetwork]
_ZN20QNetworkAddressEntry10setNetmaskERK12QHostAddress [QtXml]	_ZN20QNetworkAddressEntry12setBroadcastERK12QHostAddress [QtXml]
_ZN20QNetworkAddressEntry5setIpERK12QHostAddress [QtXml]	_ZN20QNetworkAddressEntryC1ERKS_ [QtXml]

_ZN20QNetworkAddressEntryC1Ev [QtXml]	_ZN20QNetworkAddressEntryC2ERKS_ [QtXml]
_ZN20QNetworkAddressEntryC2Ev [QtXml]	_ZN20QNetworkAddressEntryD1Ev [QtXml]
_ZN20QNetworkAddressEntryD2Ev [QtXml]	_ZN20QNetworkAddressEntryaSERKS_ [QtXml]
_ZN4QFtp10rawCommandERK7QString [QtNetwork]	_ZN4QFtp11qt_metacallEN11QMetaObject4CallEiPPv [QtNetwork]
_ZN4QFtp11qt_metacastEPKc [QtNetwork]	_ZN4QFtp12stateChangedEi [QtNetwork]
_ZN4QFtp13connectToHostERK7QStringt [QtNetwork]	_ZN4QFtp14commandStartedEi [QtNetwork]
_ZN4QFtp15commandFinishedEib [QtNetwork]	_ZN4QFtp15rawCommandReplyEiRK7QString [QtNetwork]
_ZN4QFtp15setTransferModeENS_12TransferModeE [QtNetwork]	_ZN4QFtp20clearPendingCommandsEv [QtNetwork]
_ZN4QFtp20dataTransferProgressEx [QtNetwork]	_ZN4QFtp20dataTransferProgressEx [QtNetwork]
_ZN4QFtp3getERK7QStringP9QIODeviceNS_12TransferTypeE [QtNetwork]	_ZN4QFtp3putEP9QIODeviceRK7QStringNS_12TransferTypeE [QtNetwork]
_ZN4QFtp3putERK10QByteArrayRK7QStringNS_12TransferTypeE [QtNetwork]	_ZN4QFtp4doneEb [QtNetwork]
_ZN4QFtp4listERK7QString [QtNetwork]	_ZN4QFtp4readEPcx [QtNetwork]
_ZN4QFtp5abortEv [QtNetwork]	_ZN4QFtp5closeEv [QtNetwork]
ZN4QFtp5loginERK7QStringS2 [QtNetwork]	_ZN4QFtp5mkdirERK7QString [QtNetwork]
_ZN4QFtp5rmdirERK7QString [QtNetwork]	_ZN4QFtp6removeERK7QString [QtNetwork]
ZN4QFtp6renameERK7QStringS2 [QtNetwork]	_ZN4QFtp7readAllEv [QtNetwork]
_ZN4QFtp8listInfoERK8QUrlInfo [QtNetwork]	_ZN4QFtp8setProxyERK7QStringt [QtNetwork]
_ZN4QFtp9readyReadEv [QtNetwork]	_ZN4QFtpC1EP7QIODevice [QtNetwork]
_ZN4QFtpC1EP7QIODevicePKc [QtNetwork]	_ZN4QFtpC2EP7QIODevice [QtNetwork]
_ZN4QFtpC2EP7QIODevicePKc [QtNetwork]	_ZN4QFtpD0Ev [QtNetwork]
_ZN4QFtpD1Ev [QtNetwork]	_ZN4QFtpD2Ev [QtNetwork]

_ZN5QHttp11qt_metacallEN11QMetaObject4CallEiPPv [QtNetwork]	_ZN5QHttp11qt_metacastEPKc [QtNetwork]
_ZN5QHttp12stateChangedEi [QtNetwork]	_ZN5QHttp14requestStartedEi [QtNetwork]
_ZN5QHttp15closeConnectionEv [QtNetwork]	_ZN5QHttp15requestFinishedEib [QtNetwork]
_ZN5QHttp16dataReadProgressEii [QtNetwork]	_ZN5QHttp16dataSendProgressEii [QtNetwork]
_ZN5QHttp20clearPendingRequestsEv [QtNetwork]	_ZN5QHttp22responseHeaderReceivedERK19QHttpResponseHeader [QtNetwork]
_ZN5QHttp3getERK7QStringP9QIODevice [QtNetwork]	_ZN5QHttp4doneEb [QtNetwork]
_ZN5QHttp4headERK7QString [QtNetwork]	_ZN5QHttp4postERK7QStringP9QIODeviceS4_ [QtNetwork]
_ZN5QHttp4postERK7QStringRK10QByteArrayP9QIODevice [QtNetwork]	_ZN5QHttp4readEPcx [QtNetwork]
_ZN5QHttp5abortEv [QtNetwork]	_ZN5QHttp5closeEv [QtNetwork]
_ZN5QHttp7readAllEv [QtNetwork]	_ZN5QHttp7requestERK18QHttpRequestHeaderP9QIODeviceS4_ [QtNetwork]
_ZN5QHttp7requestERK18QHttpRequestHeaderRK10QByteArrayP9QIODevice [QtNetwork]	_ZN5QHttp7setHostERK7QStringt [QtNetwork]
ZN5QHttp7setUserERK7QStringS2 [QtNetwork]	_ZN5QHttp8setProxyERK7QStringiS2_S2_ [QtNetwork]
_ZN5QHttp9readyReadERK19QHttpResponseHeader [QtNetwork]	_ZN5QHttp9setSocketEP10QTcpSocket [QtNetwork]
_ZN5QHttpC1EP7QObject [QtNetwork]	_ZN5QHttpC1ERK7QStringtP7QObject [QtNetwork]
_ZN5QHttpC2EP7QObject [QtNetwork]	_ZN5QHttpC2ERK7QStringtP7QObject [QtNetwork]
_ZN5QHttpD0Ev [QtNetwork]	_ZN5QHttpD1Ev [QtNetwork]
_ZN5QHttpD2Ev [QtNetwork]	_ZN8QUrlInfo10setSymLinkEb [QtNetwork]
_ZN8QUrlInfo11greaterThanERKS_S1_i [QtNetwork]	_ZN8QUrlInfo11setReadableEb [QtNetwork]
_ZN8QUrlInfo11setWritableEb [QtNetwork]	_ZN8QUrlInfo14setPermissionsEi [QtNetwork]
_ZN8QUrlInfo15setLastModifiedERK9QDateTime [QtNetwork]	_ZN8QUrlInfo5equalERKS_S1_i [QtNetwork]

_ZN8QUrlInfo6setDirEb [QtNetwork]	_ZN8QUrlInfo7setFileEb [QtNetwork]
_ZN8QUrlInfo7setNameERK7QString [QtNetwork]	_ZN8QUrlInfo7setSizeEx [QtNetwork]
_ZN8QUrlInfo8lessThanERKS_S1_i [QtNetwork]	_ZN8QUrlInfo8setGroupERK7QString [QtNetwork]
_ZN8QUrlInfo8setOwnerERK7QString [QtNetwork]	_ZN8QUrlInfoC1ERK4QUrlIRK7QStringS5_xRK9QDateTimeS8_bbbbbbb [QtNetwork]
_ZN8QUrlInfoC1ERK7QStringIS2_S2_xRK9QDateTimeS5_bbbbbbb [QtNetwork]	_ZN8QUrlInfoC1ERKS_ [QtNetwork]
_ZN8QUrlInfoC1Ev [QtNetwork]	_ZN8QUrlInfoC2ERK4QUrlIRK7QStringS5_xRK9QDateTimeS8_bbbbbbb [QtNetwork]
_ZN8QUrlInfoC2ERK7QStringIS2_S2_xRK9QDateTimeS5_bbbbbbb [QtNetwork]	_ZN8QUrlInfoC2ERKS_ [QtNetwork]
_ZN8QUrlInfoC2Ev [QtNetwork]	_ZN8QUrlInfoD0Ev [QtNetwork]
_ZN8QUrlInfoD1Ev [QtNetwork]	_ZN8QUrlInfoD2Ev [QtNetwork]
ZN8QUrlInfoaSERKS [QtNetwork]	_ZN9QHostInfo10lookupHostERK7QStringP7QObjectPKc [QtNetwork]
_ZN9QHostInfo11setHostNameERK7QString [QtNetwork]	_ZN9QHostInfo11setLookupIdEi [QtNetwork]
_ZN9QHostInfo12setAddressesERK5QListI12QHostAddressE [QtNetwork]	_ZN9QHostInfo13localHostNameEv [QtNetwork]
_ZN9QHostInfo14setErrorStringERK7QString [QtNetwork]	_ZN9QHostInfo15abortHostLookupEi [QtNetwork]
_ZN9QHostInfo8fromNameERK7QString [QtNetwork]	_ZN9QHostInfo8setErrorENS_13HostInfoErrorE [QtNetwork]
ZN9QHostInfoC1ERKS [QtNetwork]	_ZN9QHostInfoC1Ei [QtNetwork]
ZN9QHostInfoC2ERKS [QtNetwork]	_ZN9QHostInfoC2Ei [QtNetwork]
_ZN9QHostInfoD1Ev [QtNetwork]	_ZN9QHostInfoD2Ev [QtNetwork]
ZN9QHostInfoaSERKS [QtNetwork]	_ZNK10QTcpServer10metaObjectEv [QtNetwork]
_ZNK10QTcpServer10serverPortEv [QtNetwork]	_ZNK10QTcpServer11errorStringEv [QtNetwork]
_ZNK10QTcpServer11isListeningEv [QtNetwork]	_ZNK10QTcpServer11serverErrorEv [QtNetwork]