INTERNATIONAL STANDARD

ISO/IEC 14496-22

Third edition
2015-10-01

AMENDMENT 1
2017-01

Information technology — Coding of audio-visual objects —

Part 22: Open Font Format

AMENDMENT 1: Updates for font collections functionality

Technologies de l'information — Codage des objets audiovisuels —

Partie 22: Format de police de caractères ouvert

AMENDEMENT 1: Mises à jour des fonctionnalités des collections de polices de caractères

Reference number
ISO/IEC 14496-22:2015/Amd.1:2017(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

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 ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO 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).

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

For an explanation on 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 the following URL: www.iso.org/iso/foreword.html.

Amendment 1 to ISO/IEC 14496-22:2015 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Information technology — Coding of audio-visual objects —

Part 22:
Open Font Format

AMENDMENT 1: Updates for font collections functionality

Page 6, 4.6

Replace the last sentence of the first paragraph with the following.

By allowing multiple fonts to share glyph sets and other common font tables, font collections can result in a significant saving of file space.

Replace the text of the "NOTE" with the following.

Even though the original definition of TrueType Collection (as part of the TrueType specification) was intended to be used with fonts containing TrueType outlines, this is no longer the case. TTC files may contain various types of outlines (or a mix of them), regardless of whether or not fonts have layout tables present. For backward compatibility and simplicity, the description of the font collection file structure is using the term "TrueType Collection" though it should be understood that it is used to identify a generic font collection structure containing any type of outline tables.

Page 20, 5.2.2

Replace the description of the "checkSumAdjustment" field with the following.

To compute: set it to 0, sum the entire font as ULONG, then store 0xB1B0AFBA - sum. If the font is used as a component in a font collection file, the value of this field will be invalidated by changes to the file structure and font table directory, and must be ignored.

Page 38, 5.2.6.3

Replace the description of the Name ID 6 with the following.

PostScript name for the font; Name ID 6 specifies a string which is used to invoke a PostScript language font that corresponds to this OFF font. When translated to ASCII, the name string shall be no longer than 63 characters and restricted to the printable ASCII subset, codes 33-126, except for the 10 characters '[', ']', '(', ')', '{', '}', '<', '>', '/', '%'.

In a CFF OT font, there is no requirement that this name be the same as the font name in the CFF's Name INDEX. Thus, the same CFF may be shared among multiple font components in an OFF Font Collection. See 'name' table section of Clause 7 for additional information.

Page 61, 5.2.7.25

Replace the reference to "usTypoLineGap" with "sTypoLineGap"

Page 71, 5.3.3.1.2

Replace the paragraph describing the flag's bit "WE_HAVE_A_TWO_BY_TWO" with the following.

The bit WE_HAVE_A_TWO_BY_TWO allows for linear transformation of the x and y coordinates by specifying a 2 × 2 matrix. This could be used for scaling and 90° rotations of the glyph components, for example.

Page 71, 5.3.3.1.2

In the table describing the meaning of the flags' bits, update the description of the flag's bit "OVERLAP_COMPOUND" and "SCALED_COMPONENT_OFFSET" with the following.

For OVERLAP_COMPOUND: "If set, the components of this compound glyph overlap (used in TrueType GX fonts).

For SCALED_COMPONENT_OFFSET: Composite designed to have the component Offset scaled.

Page 74, 5.4.1

Replace the third paragraph with the following.

The Name INDEX in the CFF shall contain only one entry; that is, there shall be only one font in the CFF FontSet. The CFF Top DICT shall specify a CharstringType value of 2. It is not a requirement that this name be the same as Name ID 6 in the OFF font's 'name' table. Thus, the same CFF may be shared among multiple font components in an OFF Font Collection.

Page 343, 6.4.3.1

Add the line to the "Registered features" table with the new tag 'vrtr' named "Vertical Alternates for Rotation".

Page 401, 6.4.3.2

Add the new tag 'vrtr' named "Vertical Alternates for Rotation" with the following description.

Tag: 'vrtr'

Friendly name: Vertical Alternates For Rotation

Registered by: Adobe/Microsoft/W3C

Function: Transforms default glyphs into glyphs that are appropriate for sideways presentation in vertical writing mode. While the glyphs for most characters in East Asian writing systems remain upright when set in vertical writing mode, glyphs for other characters (such as those of other scripts or for particular Western-style punctuation) are expected to be presented sideways in vertical writing.

Example: As a first example, the glyphs for FULLWIDTH LESS-THAN SIGN (U+FF1C; "<") and FULLWIDTH GREATER-THAN SIGN (U+FF1E; ">") in a font with a non-square em-box are transformed into glyphs whose aspect ratio differs from the default glyphs, which are properly sized for sideways presentation in vertical writing mode. As a second example, the glyph for LEFT SQUARE BRACKET (U+005B, "[") in a brush-script font that exhibits slightly rising horizontal strokes may use an obtuse angle for its upper-left corner when in horizontal writing mode, but an alternate glyph with an acute angle for that corner is supplied for vertical writing mode.

Recommended implementation: The font includes versions of the glyphs covered by this feature that, when rotated 90° clockwise by the layout engine for sideways presentation in vertical writing, differ in some visual way from rotated versions of the default glyphs, such as by shifting or shape. The 'vrtr' feature maps the default glyphs to the corresponding to-be-rotated glyphs (GSUB lookup type 1).

Application interface: For GIDs found in the 'vrtr' coverage table, the layout engine passes GIDs to the feature, then gets back new GIDs.

UI suggestion: This feature should be active by default for sideways runs in vertical writing mode.

Script/language sensitivity: Applies to any script when set in vertical writing mode.

Feature interaction: The 'vrtr' and 'vert' features are intended to be used in conjunction; 'vrtr' for glyphs intended to be presented sideways in vertical writing, and 'vert' for glyphs to be presented upright. Since they shall never be activated simultaneously for a given glyph, there should be no interaction between the two features. These features are intended for layout engines that graphically rotate glyphs

for sideways runs in vertical writing mode, such as those conforming to UTR#50 (layout engines that instead depend on the font to supply pre-rotated glyphs for all sideways glyphs should use the 'vrt2' feature in lieu of 'vrtr' and vert). Because 'vrt2' supplies pre-rotated glyphs, the 'vrtr' feature should never be used with 'vrt2', but may be used in addition to any other feature.

Page 398, 6.4.3.2

Replace the current description of the tag 'vert' with the following description.

Tag: 'vert'

Friendly name: Vertical Alternates

Registered by: Adobe/Microsoft

Function: Transforms default glyphs into glyphs that are appropriate for upright presentation in vertical writing mode. While the glyphs for most characters in East Asian writing systems remain upright when set in vertical writing mode, some should be transformed (usually by rotation, shifting, or different component ordering) for upright presentation in vertical writing mode.

Example: As a first example, the glyph for HIRAGANA LETTER SMALL A (U+3041; "ぁ") is transformed into a glyph that is shifted up and to the right, which is properly positioned for upright presentation in vertical writing mode. As a second example, the glyph for SQUARE MAIKURO (U+3343; "㍇"), whose component katakana characters are ordered from left to right then top to bottom (like horizontal writing mode), is transformed into a glyph whose component katakana characters are ordered from top to bottom then right to left (like vertical writing mode).

Recommended implementation: The font includes versions of the glyphs covered by this feature that differ in some visual way from the default glyphs, such as by rotation, shifting, or different component ordering. The 'vert' feature maps the default glyphs to the corresponding vertical glyphs (GSUB lookup type 1).

Application interface: For GIDs found in the 'vert' coverage table, the layout engine passes GIDs to the feature, then gets back new GIDs.

UI suggestion: This feature should be active by default in vertical writing mode.

Script/language sensitivity: Applies only to scripts with vertical writing capability.

Feature interaction: The 'vert' and 'vrtr' features are intended to be used in conjunction; 'vert' for glyphs to be presented upright, and 'vrtr' for glyphs intended to be presented sideways in vertical writing. Since they shall never be activated simultaneously for a given glyph, there should be no interaction between the two features. These features are intended for layout engines that graphically rotate glyphs for sideways runs in vertical writing mode, such as those conforming to UTR#50 (layout engines that instead depend on the font to supply pre-rotated glyphs for all sideways glyphs should use the 'vrt2' feature in lieu of 'vert' and 'vrtr'). Because 'vrt2' supplies pre-rotated glyphs, the 'vert' feature should never be used with 'vrt2', but may be used in addition to any other feature.

Page 408, 7.6

Delete 7.6.

Page 408, 7.7

Change the subclause title to "Shape of Glyph ID 0 (the .notdef glyph)" and replace the first two paragraphs of the subclause with the following.

Glyph ID 0, the required .notdef glyph, is very important for providing the user feedback that a glyph is not found in the font. This glyph should not be left without an outline as the user will only see what looks like a space if a glyph is missing and not be aware of the active font's limitation.

It is recommended that the shape of Glyph ID 0 be either an empty rectangle, a rectangle with a question mark inside of it, or a rectangle with an 'X'. Creative shapes, like swirls or other symbols, may not be recognized by users as indicating that a glyph is missing from the font and is not being displayed at that location.

Page 412, 7.21

Delete the following paragraph.

In CFF OFF fonts, these two name strings, when translated to ASCII, must also be identical to the font name as stored in the CFF's Name INDEX.

Page 413, 7.22

Replace the last sentence with the following.

If the font is a symbol font, the first byte of the PANOSE value must be set to 'Latin Pictorial' (value = 5).

Page 414, 8.2

Replace the last sentence of the second paragraph with the following.

Additionally, the first byte of the PANOSE value in the 'OS/2' table must be set to 'Latin Pictorial' (value = 5).

Page 448, C.1

In the description of the "sTypoLineGap" field, replace the reference to "usTypoLineGap" with "sTypoLineGap".

Page 468, C.2

In the description of the "sTypoLineGap" field, replace the reference to "usTypoLineGap" with "sTypoLineGap".

Page 486, C.3

In the description of the "sTypoLineGap" field, replace the first sentence of the second paragraph with the following.

The suggested usage for sTypoLineGap is that it be used in conjunction with unitsPerEm to compute typographically correct default line spacing.

Page 505, C.4

In the description of the "sTypoLineGap" field, replace the first sentence of the second paragraph with the following:

The suggested usage for sTypoLineGap is that it be used in conjunction with unitsPerEm to compute typographically correct default line spacing.