INTERNATIONAL STANDARD

ISO/IEC 14496-4

Second edition
2004-12-15

AMENDMENT 16
2008-02-15

Information technology — Coding of audio-visual objects —

Part 4: Conformance testing

AMENDMENT 16: MPEG-J GFX conformance

Technologies de l'information — Codage des objets audiovisuels —

Partie 4: Essai de conformité

AMENDEMENT 16: Conformité au GFX MPEG-J

Reference number
ISO/IEC 14496-4:2004/Amd.16:2008(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web 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. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

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.

Amendment 16 to ISO/IEC 14495-4:2004 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This amendment adds conformance of MPEG-J GFX as defined in ISO/IEC 14496-21:2006.

Information technology — Coding of audio-visual objects —

Part 4:
Conformance testing

AMENDMENT 16: MPEG-J GFX conformance

Add the following to Clause 2 Normative References:

ISO/IEC 14496-11:2005, *Information technology — Coding of audio-visual objects — Part 11: Scene description and application engine*

ISO/IEC 14496-21:2006, *Information technology — Coding of audio-visual objects — Part 21: MPEG-J Graphics Framework eXtensions (GFX)*

Replace subclause 4.6 on MPEG-J with the following:

4.6 MPEG-J

4.6.1 MPEG-J Conformance Points

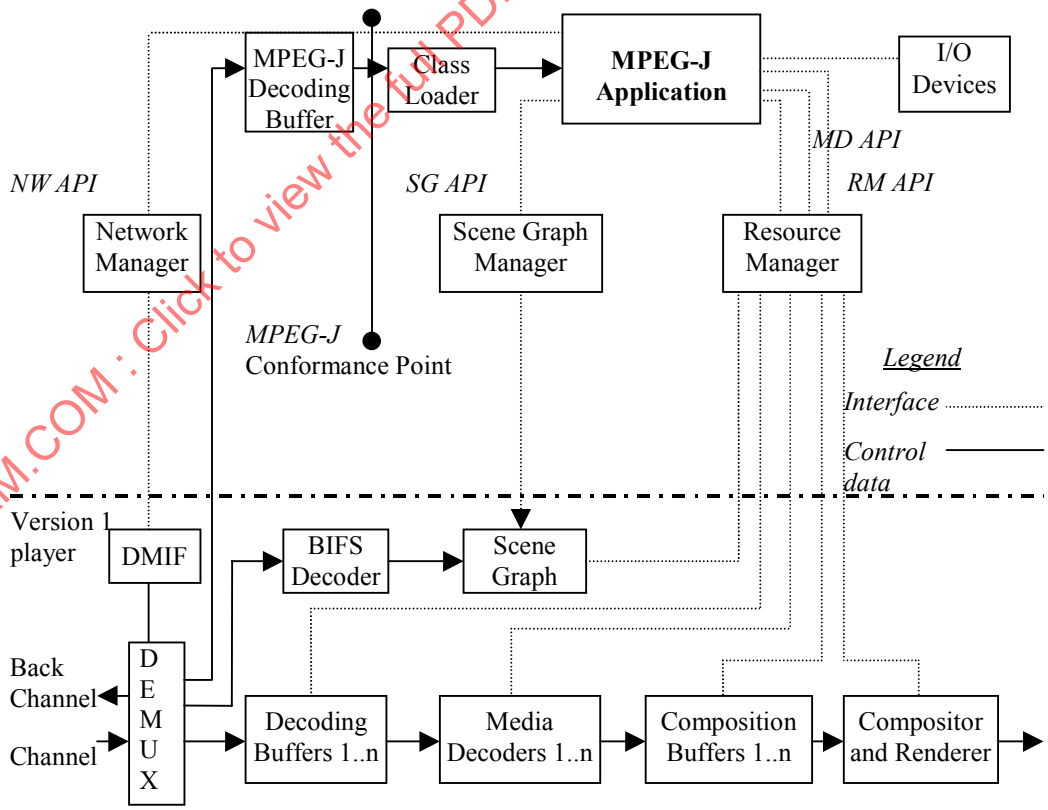

Figure AMD16.1 — MPEG-J Architecture with Conformance Point

The architecture of MPEG-J is explained in ISO/IEC 14496-11 subclauses 10.1 and 10.2. MPEG-J data is defined and the delivery mechanism explained in ISO/IEC 14496-11 subclause 10.3. MPEG-J data is delivered as an elementary stream similar to video, audio and other elementary streams.

This is de-multiplexed and stored in MPEG-J Decoding Buffers. These buffers feed the MPEG-J Decoder, which "decodes" it. In the case of classes (Java byte code), decoding means loading, while for the object and other data it means making the data available to the terminal.

The MPEG-J Decoding Buffer consists of MPEG-J Access Units. Each MPEG-J Access Unit contains either one class or one serialized object or one archive (a zip file) with a header. When this is decoded, the class file or the object data or the zip file is extracted and fed into the MPEG-J Class Loader as shown in Figure AMD16.1.

Bitstream conformance point for MPEG-J is:

- MPEG-J Decoding

At a bitstream conformance point, bitstreams will be acquired for use in testing.

Terminal conformance point for MPEG-J is:

- MPEG-J Decoding Buffer
- MPEG-J API conformance
- Java Platform conformance

An MPEG-J conformance point can be either an MPEG-J bitstream conformance point or an MPEG-J Terminal conformance point. The MPEG-J bitstream conformance points deal with the syntactic aspects while the MPEG-J terminal conformance points address the semantics.

4.6.2 Bitstream Conformance

Each bitstream shall meet the syntactic and semantic requirements specified in ISO/IEC 14496-1. This subclause describes a set of tests to be performed on bitstreams. In the description of the tests it is assumed that the tested bitstream contains no errors due to transmission or other causes. For each test the condition or conditions that must be satisfied are given, as well as the prerequisites or conditions in which the test can be applied. Note that the application of these tests requires parsing of the bitstream to the appropriate levels. Parsing and interpretation of ODs is also required. In some cases of IPMP-protected data, de-scrambling may be required before the tests can be performed on non IPMP-related features.

4.6.2.1 MPEG-J Conformance

4.6.2.1.1 Conformance Requirements

MPEG-J bitstreams shall comply with the specifications in subclause 10.3 of ISO/IEC 14496-11. The terminal shall strictly adhere to the syntax specified in subclause 10.3.3.

When the bitstream carries classes, these classes shall only use the classes, interfaces, or API (Application Programming Interface) calls from the following:

1. MPEG-J APIs defined in subclause 10.4 of ISO/IEC 14496-11 and in ISO/IEC 14496-21 (org.iso.*) for the relevant profile. [Profiles are defined in the subclause 7.9 of ISO/IEC 14496 11].

2. Java APIs supported by the underlying Java Platform for the relevant profile. These are (typically) in the java.* packages.

3. Classes or Interfaces carried in the bitstream.

These classes shall obey the security rules defined in subclause 10.2.5 of ISO/IEC 14496-11.

4.6.2.1.2 Measurement procedure

Syntax of the bitstream shall meet the requirements of subclause 10.3.3 of ISO/IEC 14496-11.

The classes should compile with only the Java Platform APIs and the MPEG-J APIs relevant to that profile.

MPEGlets and other Java classes that form the MPEG-J bitstreams shall be run through the reference MPEG-J implementation. This is a necessary but not a sufficient condition.

Verification mechanism: The API implementations shall output a trace file for every bitstream. This trace file shall be compared to see if the behavior is the same in two implementations. This idea is similar to the dump format used for BIFS.

Method packagename.classname.methodName with parameter $parameter_1$ $parameter_2$ $parameter_{3...}$ $parameter_n$ was called
where: *method_name* is the name of the method and $parameter_n$ is:

- value of the parameter - when it is a primitive data type
- the instance name - otherwise.

E.g. a method foo(var_1, var_2) would print the trace

Method org.iso.mpeg.mpegj.foo with parameter var_1 var_2
Exception packagename.*exception_name* was thrown (or)
Exception packagename.*exception_name* was thrown or with parameter *var*$_1$

4.6.2.1.3 Tolerance

There is no tolerance for bitstream syntax checking. The diagnosis is pass or fail.

4.6.3 Terminal Conformance

This subclause describes procedures to verify conformance of terminals. Each compliant decoder shall be able to decode all compliant ISO/IEC 14496-1 streams within the subset of the standard defined by the specified capabilities of the decoder.

All tests are performed using error free bitstreams. To test for correct interpretation of syntax and semantics, test sequences covering a wide range of parameters shall be supplied to the decoder under test and its output sequence shall be compared with the known expected output as described for the specific test sequence or bitstream. The comparison can be done, for example, by performing subjective evaluation, by verification of the expected result, or by comparing the timing performance. Such tests are necessary but not sufficient to prove conformance. They are helpful for discovering non-compliant implementations.

Tests are expected to be used for testing ISO/IEC 14496 decoders, including video and audio decoding, as it is generally not practical to test system decoders (or ISO/IEC 14496-1 decoders) alone. Practical test results depend on successful (or expected) output of the entire ISO/IEC 14496 decoder (systems, video, audio and DMIF).

4.6.3.1 MPEG-J Conformance

4.6.3.1.1 Conformance Requirements

Figure AMD16.1 shows the architecture an MPEG-J Terminal and the conformance points. The terminal shall follow all the rules regarding:

- MPEG-J Session and Lifecycle specified in subclause 10.2 of ISO/IEC 14496-11.
- MPEG-J Decoding and Loading specified in subclause 10.3 of ISO/IEC 14496-11.

- Semantics of the timestamps specified in subclause 10.3.2 of ISO/IEC 14496-11.

All the defined and normatively referred APIs defined in subclause 10.4 of ISO/IEC 14496-11 and in ISO/IEC 14496-21 shall be strictly followed.

4.6.3.1.1.1 MPEG-J Decoding

The Decoding process of MPEG-J data involves two steps:

a. Recovering the access unit data (class, object, or zip file) from the bit stream. This is input to the MPEG-J Class Loader.

b. Loading and Executing:

   - If the data is a class file it is loaded according to the rules specified in subclause 10.3 of ISO/IEC 14496-11.

   - If the data is a zip file the classes specified in the header are loaded according to the rules specified in subclause 10.3 of ISO/IEC 14496-11.

   - If the data is neither a class nor a zip file, it is made available according to the rules specified in subclause 10.3 of ISO/IEC 14496-11.

4.6.3.1.1.2 MPEG-J API conformance

The terminal shall implement all the APIs that are defined or normatively referenced by ISO/IEC 14496-11 and ISO/IEC 14496-21 for the relevant profile.

4.6.3.1.1.3 Java Platform conformance

The Terminal shall implement the Java Platform according to the profile. This is further elaborated in Annex A and in the Java Technology Test Suite Development Guide.

4.6.3.1.2 Measurement procedure

The recovered MPEG-J data (classes, objects, or zip files) shall be compared bit-wise with the original data. This is tested with appropriately authored MPEG-J bitstreams composed of MPEGlets and other Java classes.

The terminal shall strictly adhere to the class/interface definitions in subclause 10.4 of ISO/IEC 14496-11 and in ISO/IEC 14496-21. E.g., the class/interface names, method signatures, variable (if any) names and types, constant names and values shall be implemented as defined. This is tested with appropriately authored MPEG-J classes. Various appropriately authored MPEGlets are provided to test the various parts of the terminal implementation. There are also other Java test programs that are provided that simply exercise the different APIs that are required to be implemented and checks the appropriate behaviour. These test the API method signatures and some preliminary behavioural testing. Passing these tests is also a necessary but not a sufficient condition.

The measurement procedure for Java Platform conformance is described in detail in Annex E and in the Java Technology Test Suite Development Guide.

4.6.3.1.2.1 GFX Feature List

- The test suite shall verify the different MPEG-J features listed below.

Table AMD16.1 — GFX Test Suite Information

N°	Feature	Reference of Test sequence and associated method
1	Resource manager (J2SE)	GfxJp01.java, GfxJp01.mp4, GfxJp01.jar
2	Pluggable rendering engine (J2SE)	GfxJp02.java, GfxJp02.mp4, GfxJp02.jar
3	Media API (J2SE)	GfxJp03.java, GfxJp03.mp4, GfxJp03.jar
7	Resource Manager & Media API (J2ME)	GfxJp07.java, GfxJp07.mp4, GfxJp07.{jad,jar}
8	Terminal Capability API (J2ME)	GfxJp08.java, GfxJp08.mp4, GfxJp08.{jad,jar}

4.6.3.1.3 Bitstreams

Each bitstream contains a zip archive of the compiled java classes for that test encoded as an elementary stream.

Name	Provider	Content	
GfxJp01.mp4	SUN, Mindego, HI	Resource manager (J2SE)	
GfxJp02.mp4	Mindego, HI	Pluggable rendering engine (J2SE)	
GfxJp03.mp4	SUN, Mindego	Media API (J2SE)	
GfxJp06.mp4	HI	Media API (J2ME)	
GfxJp07.mp4	HI	Terminal Capability API (J2ME)	

4.6.3.1.4 Compiled Java Classes

The java classes encoded in each bitstream of subclause 4.6.3.1.3 are also provided as JAR format archives. The files have the same names as the bitstream files but have the extension. jar.

4.6.3.1.5 Test Java Programs

Name	Provider	Test Program	
GfxJp01.java	SUN, Mindego, HI	Resource manager (J2SE)	
GfxJp02.java	Mindego	Pluggable rendering engine (J2SE)	
GfxJp03.java	SUN, Mindego	Media API (J2SE)	
GfxJp07.java, GfxJp07.jad	HI	Resource Manager & Media API (J2ME)	
GfxJp08.java GfxJp08.jad	HI	Terminal capability (J2ME)	