

First edition  
2004-07-01

AMENDMENT 1  
2014-10-01

---

---

**Information technology — Coding of  
audio-visual objects —**

**Part 18:**

**Font compression and streaming**

**AMENDMENT 1: Updated semantics  
of decoderSpecificInfo and font data  
description for ISOBMFF**

*Technologies de l'information — Codage des objets audiovisuels —*

*Partie 18: Compression et transmission de polices de caractères*

*AMENDEMENT 1: Sémantique de decoderSpecificInfo mise à jour et  
description des données de police pour ISOBMFF*



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2014

All rights reserved. Unless otherwise specified, 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  
Case postale 56 • CH-1211 Geneva 20  
Tel. + 41 22 749 01 11  
Fax + 41 22 749 09 47  
E-mail [copyright@iso.org](mailto:copyright@iso.org)  
Web [www.iso.org](http://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.

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. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](http://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 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](http://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 WTO principles in the Technical Barriers to Trade (TBT) see the following URL: Foreword — Supplementary information.

The committee responsible for this document is ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

This amendment modifies the description and format of some existing fields of the decoderSpecificInfo and also provides a definition of font data box for ISO Base Media File Format.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 14496-18:2004/Amd 1:2014

# Information technology — Coding of audio-visual objects —

## Part 18: Font compression and streaming

### AMENDMENT 1: Updated semantics of decoderSpecificInfo and font data description for ISOBMFF

*In clause 2 “Normative References”, add the following text in the end of the clause:*

ISO/IEC 14496-22, *Information technology — Coding of audio-visual objects — Open Font Format*

ISO/IEC 14496-28, *Information technology — Coding of audio-visual objects — Composite Font Representation*

*In clause 3 “Font Data Format”, replace the second paragraph with the following text:*

MPEG-4 adopts ISO Open Font Format defined in ISO/IEC 14496-22 as its font data format for the purposes of uniform font data transmission and predictable text rendering. Originally based on OpenType®<sup>1)</sup>, it has emerged as the font solution for high-quality text processing, multimedia applications and cross platform Internet document portability. OFF is a full-featured font format that enables the highest quality of text rendering on low-resolution displays, advanced typographic features and international character support. It is fully compatible with the existing and widely adopted TrueType™<sup>2)</sup> fonts.

*In subclause 5.2.3 “Semantics”, replace “Table 10” with the following:*

**Table 10 — fontFormat values for EnhancedAccessUnit**

| fontFormat  | Access unit content                                                                                                  |
|-------------|----------------------------------------------------------------------------------------------------------------------|
| 0x00        | Forbidden                                                                                                            |
| 0x01        | ISO OFF / OpenType font with TrueType outlines (uncompressed)                                                        |
| 0x02        | ISO OFF / OpenType font with TrueType outlines compressed using the compression mechanism described in subclause 4.2 |
| 0x03        | ISO OFF / OpenType font with CFF outlines                                                                            |
| 0x04        | Composite Font Representation (as defined by ISO/IEC 14496-28)                                                       |
| 0x05 – 0x3F | ISO reserved                                                                                                         |
| 0x40 – 0x7F | User private                                                                                                         |

*In subclause 5.2.3 “Semantics”, add the following text in the end of the subclause:*

The ‘fontName’ field in the enhanced access unit header defines the font name that, together with the supplied font data, shall be used for font matching and shall override the font name specified by the underlying font data. This allows a content author to have a complete control over the font chosen for

1) OpenType is a registered trademark of Microsoft Corporation.

2) TrueType is a trademark of Apple Computer Incorporated.

content presentation and to avoid any potential conflicts with similarly named fonts that may be present in a given presentation environment.

The enhanced access units of a font stream can be used to deliver multiple fonts at the same time, and each of those fonts can be delivered either as a single complete font or subdivided into smaller subsets. In case when font subsets are used, each font subset shall have its own unique 'fontSubsetID'. A font subset may either contain a complete collection of glyphs that are needed to display a certain text fragment and, therefore, be used for a period of time and then be replaced with another subset, or it may be used to deliver only missing glyphs that were not included in the previously sent subset(s) of the same font. In the latter case, the initial font subset shall have 'storeFont' flag set to indicate that a terminal shall temporarily store the initial font subset, and all subsequent font subsets shall have 'storeFont' and 'fontSubsetExtensionFlag' set to indicate that a newly delivered font subset is an extension of the previously delivered font subset.

A terminal should either combine multiple font subset extensions into a single font file or should treat them as a virtual font collection, with each font subset being considered a component of the same logical font identified by its unique 'fontName'. In this case, if a particular glyph required for content presentation is missing in one component font, another component should be used to look up a needed glyph, and the process should be repeated until the required glyph is found. Only after all component font choices are exhausted, a missing glyph symbol should be displayed.

In cases where a font data format delivered by enhanced access unit represents a Composite Font Representation format, the referenced component fonts shall be delivered ahead of time as part of the same font stream.

*In subclause 5.4 "Font Data Decoder Configuration", replace the content of subclauses 5.4.1 and 5.4.2 with the following:*

#### 5.4.1 Syntax

```
class FontDataDecoderConfiguration extends DecoderSpecificInfo : bit(8)
tag=DecSpecificInfoTag {
    bit(7) fontFormat;
    bit(1) reserved = 0;
    if (fontFormat != 0x00) {
        bit(8) fontNameLength;
        bit(8) fontName[fontNameLength];
        bit(8) reserved = 0;
        bit(8) fontSpecInfo[sizeOfInstance - fontNameLength - 4];
    }
}
```

#### 5.4.2 Semantics

fontFormat — indicates the format of the access unit as follows.

**Table 11 — fontFormat values for FontDataDecoderConfiguration**

| fontFormat  | Access unit content                                                                                                  |
|-------------|----------------------------------------------------------------------------------------------------------------------|
| 0x00        | enhanced access unit format                                                                                          |
| 0x01        | ISO OFF / OpenType font with TrueType outlines (uncompressed)                                                        |
| 0x02        | ISO OFF / OpenType font with TrueType outlines compressed using the compression mechanism described in subclause 4.2 |
| 0x03        | ISO OFF / OpenType font with CFF outlines                                                                            |
| 0x04 – 0x3F | ISO reserved                                                                                                         |
| 0x40 – 0x7F | User private                                                                                                         |

A `fontFormat` of 0x00 indicates the use of the enhanced access unit format for this stream. In this format, all information about the font data is conveyed within each access unit.

`fontNameLength` — specifies the length in characters of the `fontName` field.

`fontName` — a Unicode (ISO/IEC 10646-1) encoded string that indicates the name of the font. See subclause 5.2.3 for semantic details of this field.

`fontSpecInfo` — is an opaque container with information for a specific font handler.

*In subclause 5.5 “Accessing the Font Data”, move the existing content of subclause 5.5 into subclause 5.5.1 “Font References” and replace it in its entirety with the following text:*

A font presented in either `BasicFontAccessUnit` or `EnhancedFontAccessUnit` of a `FontDataStream` can be identified by a number of different attributes. One common identifier for fonts is the font name. However, a font can be presented either as a whole font, or as an arbitrary subset of an original font. In both cases the font name will be the same while the actual font data included in access unit may vary. In order to explicitly identify a font (or a subset thereof) carried in MPEG-4 systems defined by ISO/IEC 14496-1 (i.e. using `ObjectDescriptor` and `ESDescriptors`), the following syntax shall be used as the font name:

“<fontName>[;OD:<odid>][;FSID:<fsid>]”, where:

- <fontName> is a Unicode encoded string that indicates the name of the font, as conveyed by `fontName` field in either `BasicFontAccessUnit` or `EnhancedFontAccessUnit` of associated font data stream;
- <odid> is the numeric value of the `objectDescriptorID` of the associated font data stream; this syntax element is optional;
- <fsid> is the numeric value of the requested font (or a font subset) as conveyed by `fontSubsetID` within the `BasicFontAccessUnit` or `EnhancedFontAccessUnit` of the associated font data stream; this syntax element is optional.

For example, a `fontName` “Times New Roman; OD:16; FSID:11” would indicate that a font data presented in the font data stream associated with the `objectDescriptorID` = 16 and is conveyed by the `FontAccessUnit` with `fontSubsetID` = 11 shall be used. If the referenced font data stream is not present, a font “Times New Roman” may be used for text rendering, if available in the MPEG-4 terminal; however, it should be understood that using an externally available font will not guarantee content presentation layout or needed character set coverage.

Similarly, in order to explicitly identify a font (or a subset thereof) carried in ISO/IEC 14496-12 (i.e. using `MPEGFontDataStreamSampleEntry`), the following syntax shall be used as the font name:

“<fontName>[;track:<tkid>][;FSID:<fsid>]”, where:

- <fontName> is a Unicode encoded string that indicates the name of the font, as conveyed by `fontName` field in either `BasicFontAccessUnit` or `EnhancedFontAccessUnit` of associated font data track;
- <tkid> is the numeric value of the `trackID` of the associated font data track; this syntax element is optional;
- <fsid> is the numeric value of the requested font (or a font subset) as conveyed by `fontSubsetID` within the `BasicFontAccessUnit` or `EnhancedFontAccessUnit` of the associated font data track; this syntax element is optional;

For example, a `fontName` “Times New Roman; track:16; FSID:11” would indicate that a font data presented in the font data track with ID = 16 and is conveyed by the `FontAccessUnit` with `fontSubsetID` = 11 shall be used.

*In subclause 5.5 “Accessing the Font Data”, add new subclause 5.5.2 “Stream Dependencies” with the following text:*

In an MPEG-4 systems context, when a font stream is used to supply fonts for a text stream, that text stream shall indicate that dependency by using the dependsonES\_ID field in the ES\_Descriptor of the text stream, containing the ES\_ID of the font stream. Other systems contexts should indicate that dependency in an appropriate way.

*In clause 5 “Font Data Stream”, add new subclause 5.6 as follows:*

## **5.6 Storage in ISO/IEC 14496-12**

Font data streams may be stored in ISO/IEC 14496-12. This subclause defines the storage of font data, when used outside of MPEG-4 Systems framework. Any media formats hosted in the ISO Base Media File Format may benefit from font data streams. This section describes the storage of font data in streams, also named font data tracks.

Box Type: fds1

Container: stbl

Mandatory: no

Quantity: One or more sample entries may be present

```
class MPEGFontDataStreamSampleEntry extends FontSampleEntry('fnt1')
{
    MPEGFontDataStreamConfigurationBox fontConfig;
}
```

```
class MPEGFontDataStreamConfigurationBox extends Box('fntC')
{
    bit(7) fontFormat;
    bit(1) reserved;
    if (fontFormat != 0x00) {
        bit(8) fontNameLength;
        bit(8) fontName[fontNameLength];
        bit(8) fontSpecInfo ; // to end of the box
    }
}
```

## **Semantics**

A font data stream is stored in ISO/IEC 14496-12 using a MPEGFontDataStreamSampleEntry sample entry in the ‘stsd’ table. The MPEGFontDataStreamConfigurationBox box gives the configuration of the font data stream. Semantics for the fields of the MPEGFontDataStreamConfigurationBox are given in section 5.4.2.

Media tracks using text carried in font data tracks shall indicate their dependency to each font data track using the ‘font’ track reference, as defined in ISO/IEC 14496-12.