INTERNATIONAL STANDARD

ISO/IEC 14496-3

Fourth edition
2009-09-01

AMENDMENT 1
2009-12-01

Information technology — Coding of audio-visual objects —

Part 3: **Audio**

AMENDMENT 1: HD-AAC profile and MPEG Surround signaling

Technologies de l'information — Codage des objets audiovisuels —

Partie 3: Codage audio

AMENDEMENT 1: Profil HD-AAC et signalisation périphérique MPEG

Reference number
ISO/IEC 14496-3:2009/Amd.1:2009(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

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 1 to ISO/IEC 14496-3:2009 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 3: **Audio**

AMENDMENT 1: HD-AAC profile and MPEG Surround signaling

In 1.5.2.1 (Profiles), add:

13. The **High Definition AAC (HD-AAC) Profile** contains the audio object types 2 (AAC LC) and 37 (SLS).

In 1.5.2.1 (Profiles), Table 1.3 (Audio Profiles definition), add (changes are highlighted in grey):

Object Type ID	Audio Object Type	…	High Definition AAC Profile
…			
2	AAC LC		X
…			
37	SLS		X
…			

In 1.5.2.3 (Levels within the profiles), immediately before 1.5.2.4 add:

- **Levels for the High Definition AAC Profile**

Table AMD1-1 — Level for the High Definition AAC Profile

Level	Max. channels/object	Max. sampling rate [kHz]	Max. word length [bit]
1	2	96	24

CBAC is not permitted.

In 1.5.2.4 (audioProfileLevelIndication), insert the following new entries into Table 1.14 (audioProfileLevelIndication values) and adapt the "reserved for ISO use" range accordingly (changes are highlighted in grey):

Value	Profile	Level
…		
0x3B	High Definition AAC Profile	L1
0x3C - 0x7F	reserved for ISO use	-
…		

In 1.6.2.1, replace Table 1.15 with the following table (where changes are highlighted in grey):

Table 1.15 — Syntax of AudioSpecificConfig()

Syntax	No. of bits	Mnemonic
AudioSpecificConfig ()		
{		
audioObjectType = GetAudioObjectType();		
samplingFrequencyIndex;	**4**	**bslbf**
if (samplingFrequencyIndex == 0xf) {		
samplingFrequency;	**24**	**uimsbf**
}		
channelConfiguration;	**4**	**bslbf**
sbrPresentFlag = -1;		
psPresentFlag = -1;		
mpsPresentFlag = -1;		
if (audioObjectType == 5 ||		
audioObjectType == 29) {		
extensionAudioObjectType = 5;		
sbrPresentFlag = 1;		
if (audioObjectType == 29) {		
psPresentFlag = 1;		
}		
extensionSamplingFrequencyIndex;	**4**	**uimsbf**
if (extensionSamplingFrequencyIndex == 0xf)		
extensionSamplingFrequency;	**24**	**uimsbf**
audioObjectType = GetAudioObjectType();		
if (audioObjectType == 22)		
extensionChannelConfiguration;	**4**	**uimsbf**
}		
else {		
extensionAudioObjectType = 0;		
}		
switch (audioObjectType) {		
case 1:		
case 2:		
case 3:		
case 4:		
case 6:		
case 7:		
case 17:		
case 19:		
case 20:		
case 21:		
case 22:		
case 23:		
GASpecificConfig();		
break:		
case 8:		
CelpSpecificConfig();		
break;		
case 9:		
HvxcSpecificConfig();		
break:		
case 12:		
TTSSpecificConfig();		
break;		

Table 1.15 (*continued*)

Syntax	No. of bits	Mnemonic
case 13:		
case 14:		
case 15:		
case 16:		
StructuredAudioSpecificConfig();		
break;		
case 24:		
ErrorResilientCelpSpecificConfig();		
break;		
case 25:		
ErrorResilientHvxcSpecificConfig();		
break;		
case 26:		
case 27:		
ParametricSpecificConfig();		
break;		
case 28:		
SSCSpecificConfig();		
break;		
case 30:		
mpsPresentFlag = 1;		
sacPayloadEmbedding;	**1**	**uimsbf**
SpatialSpecificConfig();		
break;		
case 32:		
case 33:		
case 34:		
MPEG_1_2_SpecificConfig();		
break;		
case 35:		
DSTSpecificConfig();		
break;		
case 36:		
fillBits;	**5**	**bslbf**
ALSSpecificConfig();		
break;		
case 37:		
case 38:		
SLSSpecificConfig();		
break;		
case 39:		
ELDSpecificConfig(channelConfiguration);		
break:		
case 40:		
case 41:		
SymbolicMusicSpecificConfig();		
break;		
default:		
/* reserved */		
}		
switch (audioObjectType) {		

Table 1.15 (*continued*)

```
Syntax                                                                No. of bits   Mnemonic
    case 17:
    case 19:
    case 20:
    case 21:
    case 22:
    case 23:
    case 24:
    case 25:
    case 26:
    case 27:
    case 39:
        epConfig;                                                     2             bslbf
        if ( epConfig == 2 || epConfig == 3 ) {
            ErrorProtectionSpecificConfig();
        }
        if ( epConfig == 3 ) {
            directMapping;                                            1             bslbf
            if ( ! directMapping ) {
                /* tbd */
            }
        }
    }
    extensionIdentifier = -1;
    if ( bits_to_decode() >= 11 ) {
        extensionIdentifier;                                          11            bslbf
    }
    if ( extensionIdentifier == 0x2b7 ) {
        extensionIdentifier = -1;
        if ( extensionAudioObjectType != 5 && bits_to_decode() >= 5 ) {
            extensionAudioObjectType = GetAudioObjectType();
            if ( extensionAudioObjectType == 5 ) {
                sbrPresentFlag;                                       1             uimsbf
                if (sbrPresentFlag == 1) {
                    extensionSamplingFrequencyIndex;                  4             uimsbf
                    if ( extensionSamplingFrequencyIndex == 0xf ) {
                        extensionSamplingFrequency;                   24            uimsbf
                    }
                }
                if ( bits_to_decode() >= 12 ) {
                    extensionIdentifier;                              11            bslbf
                    if (extensionIdentifier == 0x548) {
                        extensionIdentifier = -1;
                        psPresentFlag;                                1             uimsbf
                    }
                }
            }
            if ( extensionAudioObjectType == 22 ) {
                sbrPresentFlag;                                       1             uimsbf
                if (sbrPresentFlag == 1) {
                    extensionSamplingFrequencyIndex;                  4             uimsbf

                    if ( extensionSamplingFrequencyIndex == 0xf )
                        extensionSamplingFrequency;                   24            uimsbf
                }
                extensionChannelConfiguration;                        4             uimsbf
            }
        }
```

Table 1.15 (*continued*)

Syntax	No. of bits	Mnemonic
}		
if (extensionIdentifier == -1 && bits_to_decode() >= 11) {		
extensionIdentifier;	**11**	**bslbf**
}		
if (extensionIdentifier == 0x76a) {		
if (audioObjectType != 30 && bits_to_decode() >= 1) {		
mpsPresentFlag;	**1**	**uimsbf**
if (mpsPresentFlag == 1) {		
sacPayloadEmbedding = 1;		
sscLen;	**8**	**uimsbf**
if (sscLen == 0xff) {		
sscLenExt;	**16**	**uimsbf**
sscLen += sscLenExt;		
}		
SpatialSpecificConfig();		
}		
}		
}		
}		

In 1.6.3.11, replace **syncExtensionType** *by* **extensionIdentifier**.

After 1.6.3.17, add the following three new subclauses:

1.6.3.18 mpsPresentFlag

A one bit field indicating the presence or absence of MPEG Surround data. The value –1 indicates that the mpsPresentFlag was not conveyed in the AudioSpecificConfig(). In this case, a decoder supporting the Baseline MPEG Surround Profile shall support implicit signaling as defined in 7.2.3 and 7.2.4 of ISO/IEC 23003-1:2007.

1.6.3.19 sscLen

A helper variable indicating the number of bytes of the subsequent SpatialSpecificConfig() data function including possible fill bits.

1.6.3.20 sscLenExt

A helper variable indicating the additional number of bytes of the subsequent SpatialSpecificConfig() data function including possible fill bits.