INTERNATIONAL STANDARD

ISO/IEC 14496-12

Fifth edition
2015-12-15

AMENDMENT 1
2017-03

Information technology — Coding of audio-visual objects —

Part 12: ISO base media file format

AMENDMENT 1: DRC Extensions

Technologies de l'information — Codage des objets audiovisuels —
Partie 12: Format ISO de base pour les fichiers médias
AMENDEMENT 1: Extensions DRC

Reference number
ISO/IEC 14496-12:2015/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) 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 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 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 the following URL: www.iso.org/iso/foreword.html.

Amendment 1 to ISO/IEC 14496-12:2015 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 12:
ISO base media file format

AMENDMENT 1: DRC Extensions

Page 1, Normative references

Add the following new reference:

ISO/IEC 23003-4:2015/Amd.1, *Information technology — MPEG audio technologies — Part 4: Dynamic range control, AMENDMENT 1: Parametric DRC, gain mapping and equalization tools.*

Replace the normative reference:

ITU-R, Recommendation ITU-R BS.1770-3. *Algorithm to measure audio programme loudness and true-peak audio level,* August 2012.

with:

ITU-R, Recommendation ITU-R BS.1770-4. *Algorithms to measure audio programme loudness and true-peak audio level,* October 2015.

Page 2, Terms and definitions

Add the following new definition after 3.1.8 and adjust subsequent term numbers:

3.1.9

mod

modulo operator: (x mod y) = x-y floor (x/y)

Page 160, 12.2.3.1

Add the following paragraph at the end of the subclause:

Encoders should encode the DRC-related boxes in the `AudioSampleEntry` in the order given in 12.2.3.2. Decoders may ignore and discard the DRC-related boxes if they are not in that order. DRC-related boxes include `ChannelLayout`, `DownMixInstructions`, `DRCCoefficientsBasic`, `DRCInstructionsBasic`, `DRCCoefficientsUniDrc`, `DRCInstructionsUniDrc`, and `UniDrcConfigExtension`. The `DownMixInstructions` and `DRCInstructionsUniDrc` box cannot occur more than once if the box has `version==1`, but it can occur multiple times if `version==0`.

Page 161, 12.2.3.2

Replace the definition of AudioSampleEntry and AudioSampleEntryV1 with:

```
class AudioSampleEntry(codingname) extends SampleEntry (codingname){
    const unsigned int(32)[2] reserved = 0;
    template unsigned int(16) channelcount = 2;
    template unsigned int(16) samplesize = 16;
    unsigned int(16) pre_defined = 0;
    const unsigned int(16) reserved = 0 ;
    template unsigned int(32) samplerate = { default samplerate of media}<<16;
    // optional boxes follow
    Box ();         // further boxes as needed
    ChannelLayout();
    DownMixInstructions() [];
    DRCCoefficientsBasic() [];
    DRCInstructionsBasic() [];
    DRCCoefficientsUniDRC() [];
    DRCInstructionsUniDRC() [];
    // we permit only one DRC Extension box:
    UniDrcConfigExtension();
    Box ();         // further boxes as needed
}
aligned(8) class SamplingRateBox extends FullBox('srat') {
    unsigned int(32) sampling_rate;
}
class AudioSampleEntryV1(codingname) extends SampleEntry (codingname){
    unsigned int(16) entry_version;    // must be 1,
                        // and must be in an stsd with version ==1
    const unsigned int(16)[3] reserved = 0;
    template unsigned int(16) channelcount;    // must be correct
    template unsigned int(16) samplesize = 16;
    unsigned int(16) pre_defined = 0;
    const unsigned int(16) reserved = 0 ;
    template unsigned int(32) samplerate = 1<<16;
    // optional boxes follow
    SamplingRateBox();
    Box ();         // further boxes as needed
    ChannelLayout();
    DownMixInstructions() [];
    DRCCoefficientsBasic() [];
    DRCInstructionsBasic() [];
    DRCCoefficientsUniDRC() [];
    DRCInstructionsUniDRC() [];
    // we permit only one DRC Extension box:
    UniDrcConfigExtension();
    Box ();         // further boxes as needed
}
```

Page 163, 12.2.5.2

Replace the definition of DownMixInstructions with:

```
aligned(8) class DownMixInstructions extends FullBox('dmix', version, flags=0) {
    if (version >= 1) {        bit(1) reserved = 0;
        bit(7) downmix_instructions_count;
    } else {
        int downmix_instructions_count = 1;
    }
    for (a=1; a<=downmix_instructions_count; a++) {
        unsigned int(8) targetLayout;
        unsigned int(1) reserved = 0;
        unsigned int(7) targetChannelCount;
        bit(1) in_stream;
        unsigned int(7) downmix_ID;
        if (in_stream==0)
        {    // downmix coefficients are out of stream and supplied here
            int i, j;
            if (version >= 1) {
```

```
                bit(4) bs_downmix_offset;
                int size = 4;
                for (i=1; i <= targetChannelCount; i++){
                    for (j=1; j <= baseChannelCount; j++) {
                    bit(5) bs_downmix_coefficient_v1;
                    size += 5;
                  }
                }
                bit(size mod 8) reserved = 0;
        } else {
                for (i=1; i <= targetChannelCount; i++){
                    for (j=1; j <= baseChannelCount; j++) {
                        bit(4) bs_downmix_coefficient;
                    }
                }
            }
        }
    }
}
```

Page 163, 12.2.5.3

Add the following two new definitions after the definition of downmix_ID:

`version` is an integer that specifies the version of this box (0 or 1)

`bs_downmix_offset` is an offset in dB for all downmix coefficients that are defined in the `bs_downmix_coefficient_v1` field. It is encoded as defined in the following table using the following expression for a:

$$a = 20\log_{10}\left(\frac{\text{targetChannelCount}}{\text{baseChannelCount}}\right)$$

Table 5 — Downmix Offset Encoding

Value [dB]	Hex Encoding (3 bits)
0,0	0x0
$0{,}5\,\text{round}(a)$	0x1
$0{,}5\,\text{round}(2a)$	0x2
reserved	other

Add the following new definition and table at the end of the subclause (after Table 7):

`bs_downmix_coefficient_v1` is encoded as defined in the following table:

Table 8 — Downmix Coefficient Encoding for version>=1 (bs_downmix_coefficient_v1)

Value	Hex Encoding (5 bits)
10,00 dB	0x00
6,00 dB	0x01
4,50 dB	0x02
3,00 dB	0x03
1,50 dB	0x04
0,00 dB	0x05
-0,50 dB	0x06

Table 8 *(continued)*

Value	Hex Encoding (5 bits)
-1,00 dB	0x07
-1,50 dB	0x08
-2,00 dB	0x09
-2,50 dB	0x0A
-3,00 dB	0x0B
-3,50 dB	0x0C
-4,00 dB	0x0D
-4,50 dB	0x0E
-5,00 dB	0x0F
-5,50 dB	0x10
-6,00 dB	0x11
-6,50 dB	0x12
-7,00 dB	0x13
-7,50 dB	0x14
-8,00 dB	0x15
-9,00 dB	0x16
-10,00 dB	0x17
-11,00 dB	0x18
-12,00 dB	0x19
-15,00 dB	0x1A
-20,00 dB	0x1B
-25,00 dB	0x1C
-30,00 dB	0x1D
-40,00 dB	0x1E
-∞ dB	0x1F

Page 164, Table 5

Replace the table number and title with:

Table 6 — Downmix Coefficient Encoding for non-LFE Channel and version==0 (bs_downmix_coefficient)

Page 164, Table 6

Replace the table number and title with:

Table 7 — Downmix Coefficient Encoding for LFE Channel and version==0 (bs_downmix_coefficient)

Page 165, 12.2.6

Replace the first paragraph with:

A DRC is used in the encoder to generate gain values using one of the pre-defined DRC characteristics as defined in ISO/IEC 23001-8 or a characteristic defined in ISO/IEC 23003-4:2015/Amd.1. The coefficients

are placed either in-stream or in an associated meta-data track. Alternatively, coefficients are generated at the decoder based on transmitted parametric DRC configurations.

Replace the last paragraph with:

The boxes `DRCCoefficientsBasic`, `DRCCoefficientsUniDRC`, `DRCInstructionsBasic`, `DRCInstructionsUniDRC` and `UniDrcConfigExtension` may occur in an AudioSampleEntry and are defined in ISO/IEC 23003-4:2015 and ISO/IEC 23003-4:2015/Amd.1.

Page 165, 12.2.7.1

Replace the last paragraph with:

The program loudness is measured using ITU-R BS.1770-4 over the associated content; the 'anchor loudness' is the loudness of the anchor content, where what that content is, is determined by the content author; one suitable value (especially for content for which the main content is speech) is 'dialog normal level' or DialNorm as defined in ATSC Doc. A/52:2012. ISO/IEC 23003-4 specifies the measurement systems, measurement methods and the coding of all loudness and peak-related values.

Page 166, 12.2.7.2

Replace the definition of Syntax with:

```
aligned(8) class LoudnessBaseBox extends FullBox(loudnessType, version, flags=0) {
    if (version >= 1) {
    unsigned int(2) reserved = 0;
    unsigned int(6) loudness_base_count;
    } else {
    int loudness_base_count = 1;
    }
    for (a=1; a<=loudness_base_count; a++) {
    if (version >= 1) {
    unsigned int(2) reserved = 0;
    unsigned int(6) EQ_set_ID;    // to match an EQ box
    }
    unsigned int(3) reserved = 0;
    unsigned int(7) downmix_ID;    // matching downmix
    unsigned int(6) DRC_set_ID;    // to match a DRC box
    signed int(12) bs_sample_peak_level;
    signed int(12) bs_true_peak_level;
    unsigned int(4) measurement_system_for_TP;
    unsigned int(4) reliability_for_TP;
    unsigned int(8) measurement_count;
    int i;
    for (i = 1 ; i <= measurement_count; i++){
    unsigned int(8) method_definition;
    unsigned int(8) method_value;
    unsigned int(4) measurement_system;
    unsigned int(4) reliability;
    }
    }
}
aligned(8) class TrackLoudnessInfo extends LoudnessBaseBox('tlou') { }
aligned(8) class AlbumLoudnessInfo extends LoudnessBaseBox ('alou') { }
aligned(8) class LoudnessBox extends Box('ludt') {
    // not more than one TrackLoudnessInfo box with version>=1 is allowed
    Loudness    TrackLoudnessInfo[];
    // not more than one AlbumLoudnessInfo box with version>=1 is allowed
    albumLoudness    AlbumLoudnessInfo[];
}
```

Page 167, 12.2.7.3