

---

---

**Information technology — Generic  
coding of moving pictures and  
associated audio information —**

**Part 1:  
Systems**

**AMENDMENT 6: Carriage of Quality  
Metadata in MPEG-2 Systems**

*Technologies de l'information — Codage générique des images  
animées et du son associé —*

*Partie 1: Systèmes*

*AMENDEMENT 6: Transport des métadonnées de qualité dans les  
systèmes MPEG-2*



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2016, Published in Switzerland

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  
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 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](#)

Amendment 6 to ISO/IEC 13818-1:2015 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*, in collaboration with ITU-T. The identical text is published as Rec. ITU-T H.222.0 (07/2016)/Amd.6.

IECNORM.COM : Click to view the full PDF of ISO/IEC 13818-1:2015/Amd 6:2016

INTERNATIONAL STANDARD  
ITU-T RECOMMENDATION

Information technology – Generic coding of moving pictures and  
associated audio information: Systems

Amendment 6

Carriage of quality metadata in MPEG-2 systems

1) **Clause 1.2.3**

In clause 1.2.3 add:

- ISO/IEC 23001-10:2015, *Information technology — MPEG systems technologies — Part 10: Carriage of timed metadata metrics of media in ISO base media file format.*

2) **Clause 2.1**

In clause 2.1 add the following after clause 2.1.128:

**2.1.129 quality access unit:** An access unit that contains dynamic quality metadata as defined in ISO/IEC 23001-10.

3) **Table 2-31**

Replace Table 2-31 by the following:

**Table 2-31 – table\_id assignment values**

| Value     | Description                                                    |
|-----------|----------------------------------------------------------------|
| 0x00      | program_association_section                                    |
| 0x01      | conditional_access_section (CA_section)                        |
| 0x02      | TS_program_map_section                                         |
| 0x03      | TS_description_section                                         |
| 0x04      | ISO_IEC_14496_scene_description_section                        |
| 0x05      | ISO_IEC_14496_object_descriptor_section                        |
| 0x06      | Metadata_section                                               |
| 0x07      | IPMP_Control_Information_section (defined in ISO/IEC 13818-11) |
| 0x08      | ISO_IEC_14496_section                                          |
| 0x09      | ISO/IEC 23001-11 (Green Access Unit) section                   |
| 0x0A      | ISO/IEC 23001-10 (Quality Access Unit) section                 |
| 0x0B-0x37 | Rec. ITU-T H.222.0   ISO/IEC 13818-1 reserved                  |
| 0x38-0x3F | Defined in ISO/IEC 13818-6                                     |
| 0x40-0xFE | User private                                                   |
| 0xFF      | Forbidden                                                      |

4) **Table 2-34**

In Table 2-34 add the following:

|           |                                               |
|-----------|-----------------------------------------------|
| 0x2F      | Quality Access Units carried in sections      |
| 0x30-0x7E | Rec. ITU-T H.222.0   ISO/IEC 13818-1 reserved |

## 5) Clause 2.6.90

Replace Table 2-105 with:

Table 2-105 – Extension descriptor

| Syntax                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | No. of bits        | Mnemonic                          |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|-----------------------------------|
| <pre> Extension_descriptor () {     <b>descriptor_tag</b>     <b>descriptor_length</b>     <b>extension_descriptor_tag</b>     if ( extension_descriptor_tag == 0x02 ) {         <b>ObjectDescriptorUpdate()</b>     }     else if ( extension_descriptor_tag == 0x03 ) {         <b>HEVC_timing_and_HRD_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x04 ) {         <b>af_extension_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x05 ) {         <b>HEVC_operation_point_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x06 ) {         <b>HEVC_hierachy_extension_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x07 ) {         <b>Green_extension_descriptor ()</b>     }     else if ( extension_descriptor_tag == 0x08 ) {         <b>MPEG-H_3dAudio_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x09 ) {         <b>MPEG-H_3dAudio_config_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x0A ) {         <b>MPEG-H_3dAudio_scene_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x0B ) {         <b>MPEG-H_3dAudio_text_label_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x0C ) {         <b>MPEG-H_3dAudio_multi-stream_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x0D ) {         <b>MPEG-H_3dAudio_drc_loudness_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x0E ) {         <b>MPEG-H_3dAudio_command_descriptor()</b>     }     else if ( extension_descriptor_tag == 0x0F ) {         <b>Quality_extension_descriptor ()</b>     }     else { </pre> | <pre> 8 8 8 </pre> | <pre> uimsbf uimsbf uimsbf </pre> |

Table 2-105 – Extension descriptor

| Syntax                                                 | No. of bits | Mnemonic |
|--------------------------------------------------------|-------------|----------|
| <pre> for ( i=0; i&lt;N; i++ ) {     reserved } </pre> | 8           | bslbf    |

6) **Clause 2.6.91**

Add the following immediately before Table 2-106:

**Quality\_extension\_descriptor()** – This structure is defined in 2.6.119.

Replace Table 2-106 with:

Table 2-106 – Extension descriptor tag values

| Extension_descriptor_tag | TS  | PS  | Identification                                |
|--------------------------|-----|-----|-----------------------------------------------|
| 0                        | n/a | n/a | Reserved                                      |
| 1                        | n/a | X   | Forbidden                                     |
| 2                        | X   | X   | ODUpdate_descriptor                           |
| 3                        | X   | n/a | HEVC_timing_and_HRD_descriptor()              |
| 4                        | X   | n/a | af_extensions_descriptor()                    |
| 5                        | X   | n/a | HEVC_operation_point_descriptor()             |
| 6                        | X   | n/a | hierarchy_extension_descriptor()              |
| 7                        | X   | n/a | Green_extension_descriptor()                  |
| 8                        | X   | n/a | MPEG-H_3dAudio_descriptor()                   |
| 9                        | X   | n/a | MPEG-H_3dAudio_config_descriptor()            |
| 0x0A                     | X   | n/a | MPEG-H_3dAudio_scene_descriptor()             |
| 0x0B                     | X   | n/a | MPEG-H_3dAudio_text_label_descriptor()        |
| 0x0C                     | X   | n/a | MPEG-H_3dAudio_multi-stream_descriptor()      |
| 0x0D                     | X   | n/a | MPEG-H_3dAudio_drc_loudness_descriptor()      |
| 0x0E                     | X   | n/a | MPEG-H_3dAudio_command_descriptor()           |
| 0x0F                     | X   | n/a | Quality_extension_descriptor()                |
| 0x10-0xFF                | n/a | n/a | Rec. ITU-T H.222.0   ISO/IEC 13818-1 Reserved |

7) **Clause 2.6.119**

Add the following clause after clause 2.6.118:

**2.6.119 Quality extension descriptor**

The Quality extension descriptor shall be sent once per event or program and hence is signalled using a descriptor in the Program Map Table. This descriptor shall appear in the elementary stream loop of the PID for which quality information is provided.

Dynamic quality metadata is stored in access units and is associated with one or more video frames. These access units are encapsulated in MPEG sections identified by stream\_type value of 0x2F.

The Quality extension descriptor describes metrics that are present in each Quality Access Unit, and the constant field size that is used for the values. The quality metrics are defined in clause 4.3 of ISO/IEC 23001-10.

The syntax of Quality extension descriptor containing static metadata is shown in Table 111terdecies.

Table 2-111 *terdecies* – Quality extension descriptor

| Syntax                                                                                                                                                      | No. bits     | Mnemonic                                        |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|-------------------------------------------------|
| Quality_extension_descriptor() {<br><b>field_size_bytes</b><br><b>metric_count</b><br>for (i=0; i < metric_count; i++) {<br><b>metric_code[i]</b><br>}<br>} | 8<br>8<br>32 | <b>uimsbf</b><br><b>uimsbf</b><br><b>uimsbf</b> |

**field\_size\_bytes** indicates the constant size in bytes of the value for a metric in each Quality Access Unit

**metric\_count**: The number of metrics for quality values in each Quality Access Unit

**metric\_code** is the code name of the metrics in the Quality Access Unit

Semantics for all the syntax elements above are specified in clause 4.2 of ISO/IEC 23001-10. The quality metrics to be signalled can also be found in clause 4.3 of ISO/IEC 23001-10 and include:

- PSNR = Peak Signal to Noise Ratio
- SSIM = Structural Similarity Index
- MS-SSIM = Multi-Scale Structural Similarity Index
- VQM = Video Quality Metric
- PEVQ = Perceptual Evaluation of Video Quality
- MOS = Mean Opinion Score
- FSIG = Frame significance

## 7) Clause 2.20

*Add the following clause after clause 2.19:*

### 2.20 Carriage of Quality Access Units in MPEG-2 sections

#### 2.20.1 General Description

Quality Access Units carrying dynamic quality metadata associated with one or more video frames, i.e., a given quality metric sample can be applicable for multiple video frames for a given DTS interval, until a new quality metric sample is declared. These access units are encapsulated in MPEG sections identified by stream\_type value of 0x2F.

Each Quality Access Unit contains configuration and timing information, as well as an array of quality values, corresponding one for one to the declared metrics. Each value is padded by preceding zero bytes, as needed, to the number of bytes indicated by field\_size\_bytes.

Quality\_access\_unit shall be carried in MPEG sections.

Every Quality Access Unit shall be a random access point.

NOTE – In order to make processing easier, each Quality Access Unit should be contained in a single TS packet.

The syntax for the Quality Access Units is provided in Table 2-124. The attribute definitions are aligned with those in ISO/IEC 23001-10.

Table 2-124 – Quality Access Unit

| Syntax                                | No. bits | Mnemonic      |
|---------------------------------------|----------|---------------|
| Quality_Access_Unit() {               |          |               |
| <b>field_size_bytes</b>               | 8        | <b>uimsbf</b> |
| <b>metric_count</b>                   | 8        | <b>uimsbf</b> |
| for ( i = 0; i < metric_count; i++) { |          |               |
| <b>metric_code</b>                    | 32       | <b>uimsbf</b> |
| <b>sample_count</b>                   | 8        | <b>uimsbf</b> |
| for ( j = 0; j < sample_count; j++) { |          |               |
| '0010'                                | 4        | <b>bslbf</b>  |
| <b>media_DTS [32..30]</b>             | 3        | <b>uimsbf</b> |
| <b>marker_bit</b>                     | 1        | <b>bslbf</b>  |
| <b>media_DTS [29..15]</b>             | 15       | <b>uimsbf</b> |
| <b>marker_bit</b>                     | 1        | <b>bslbf</b>  |
| <b>media_DTS [14..0]</b>              | 15       | <b>uimsbf</b> |
| <b>marker_bit</b>                     | 1        | <b>bslbf</b>  |
| <b>quality_metric_sample</b>          | N        | <b>uimsbf</b> |
| }                                     |          |               |
| }                                     |          |               |
| }                                     |          |               |

**field\_size\_bytes**: size of the quality\_metric\_sample field. As such,  $N=8*\text{field\_size\_bytes}$

**metric\_count**: The number of metrics for quality values in the Quality Access Unit, as defined in clause 4.2 of ISO/IEC 23001-10

**metric\_code**: The code name of the metrics in the Quality Access unit, as defined in clause 4.2 of ISO/IEC 23001-10

**sample\_count**: Number of quality metric samples per metric

**media\_DTS**: DTS of the media access unit described by the quality metric sample.

As such, quality\_metric\_sample is an array signalled using the 'i' and 'j' values of the loop.

### 2.20.2 Buffer model for processing Quality Access Units

The buffer model reflects processing required to handle Quality Access Units. The model can be used to establish constraints which can be used to verify the validity of dynamic Quality metadata prepared in accordance with this Standard.

Figure 2-3bis illustrates the buffer model for processing Quality Access Units.

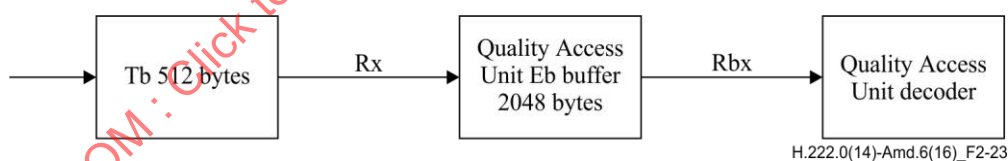


Figure 2-23 – Quality Access Unit decoder processing model

MPEG-2 Transport Stream packets come into the model at the left, and are filtered by PID. Packets whose PID matches the Quality Access Unit PID flow into the 512 byte transport buffer. These buffered packets are removed at a rate of  $R_x = 300$  kbps (kilobits/second) and stored in the Quality Access Unit Eb buffer (2048 Bytes). Quality Access Unit table sections are removed from the Eb buffer immediately after the full access unit is available (based on section length) and these are passed onto the Quality Access Unit decoder at a rate  $R_{bx} = 300$  kbps for decoding and association with the corresponding video frames. Eb buffer shall not overflow and the Quality Access Unit section shall be available in the Eb buffer to meet the requirement of DTS earlier than the DTS of the latest media access unit described in the corresponding Quality Access Unit.