



INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.222.0

Amendment 7
(03/2017)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Transmission
multiplexing and synchronization

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

Amendment 7: Virtual segmentation

CAUTION !

PREPUBLISHED RECOMMENDATION

This prepublication is an unedited version of a recently approved Recommendation. It will be replaced by the published version after editing. Therefore, there will be differences between this prepublication and the published version.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU [had/had not] received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at <http://www.itu.int/ITU-T/ipr/>.

© ITU 2017

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

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

Amendment 7

Virtual segmentation

Summary

Amendment 7 to ITU-T H.222.0 (2014) | ISO/IEC 13818-1:2015 enables signaling and notation of a stream with virtual segmentation carried in MPEG-2 transport stream. The signaling of the stream that is comprised of virtual segmentation is indicated within the PMT using virtual segmentation descriptor. The virtual segment itself is made up of a boundary point af_descriptor to indicate virtual segment locations within a stream, a labeling af_descriptor which notes the different types of label associated with the virtual segment, and an associated TEMI timeline af_descriptor.

In clause 2.6.90, replace Table 2-107 with:

Table 2-107 – Extension descriptor

Syntax	No. of bits	Mnemonic
<pre> Extension_descriptor () { descriptor_tag descriptor_length extension_descriptor_tag if (extension_descriptor_tag == 0x02) { ObjectDescriptorUpdate() } else if (extension_descriptor_tag == 0x03) { HEVC_timing_and_HRD_descriptor() } else if (extension_descriptor_tag == 0x04) { af_extension_descriptor() } else if (extension_descriptor_tag == 0x05) { HEVC_operation_point_descriptor() } else if (extension_descriptor_tag == 0x06) { HEVC_hierachy_extension_descriptor() } else if (extension_descriptor_tag == 0x07) { Green_extension_descriptor () } else if (extension_descriptor_tag == 0x08) { MPEG-H_3dAudio_descriptor() } else if (extension_descriptor_tag == 0x09) { MPEG-H_3dAudio_config_descriptor() } else if (extension_descriptor_tag == 0x0A) { MPEG-H_3dAudio_scene_descriptor() } </pre>	<p>8</p> <p>8</p> <p>8</p>	<p>uimsbf</p> <p>uimsbf</p> <p>uimsbf</p>

Syntax	No. of bits	Mnemonic
<pre> } else if (extension_descriptor_tag == 0x0B) { MPEG-H_3dAudio_text_label_descriptor() } else if (extension_descriptor_tag == 0x0C) { MPEG-H_3dAudio_multi-stream_descriptor() } else if (extension_descriptor_tag == 0x0D) { MPEG-H_3dAudio_drc_loudness_descriptor() } else if (extension_descriptor_tag == 0x0E) { MPEG-H_3dAudio_command_descriptor() } else if (extension_descriptor_tag == 0x0F) { Quality_extension_descriptor () } else if (extension_descriptor_tag == 0x10) { Virtual_segmentation_descriptor () } else { for (i=0; i<N; i++) { reserved } } } </pre>	8	bslbf

Replace Table 2-108 with:

Table 2-108: 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	HEVC_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	X	N/A	Virtual_segmentation_descriptor()
0x11-0xFF	N/A	N/A	Rec. ITU-T H.222.0 ISO/IEC 13818-1 Reserved

explicit_boundary_flag: If set to 0, this elementary stream is a dependent stream, and boundary data for it is provided on a reference partition on a different PID, specified by `boundary_PID`; otherwise, the current PID carries boundary descriptors.

partition_id: ID of the partition described in the boundary descriptor.

boundary_PID: PID carrying `boundary_descriptor()` that is used by this partition of this elementary stream.

SAP_type_max: Maximum possible value of SAP in this partition. If `SAP_type_max` value is 0, any SAP value may appear in the stream.

maximum_duration: Maximum virtual segment duration for a segment on partition `partition_id`, expressed in units of `ticks_per_second`. For consecutive virtual segments $S(i)$ and $S(i+1)$ on the above partition, if $PTS(i)$ stands for the earliest PTS in segment $S(i)$, and $PTS(i+1)$ stands for the earliest PTS in segment $S(i+1)$ this duration equals $(PTS(i+1) - PTS(i)) * ticks_per_second / 90000$. If set to 0, virtual segment duration is unlimited.

Add the following changes to ISO/IEC 13818-1 Annex U

Make current text of U.1 a clause named U.1.1 General

Append the following text to the end of clause U.1 (Introduction):

...

This annex also specifies a format for carriage of boundary and labeling descriptors that may be used to indicate a boundary type for seamless content splicing or switching in the applications of Ad insertion, cloud DVR recording and segmentation of adaptive bit rate streaming. The possible resolving and consumption of the boundary descriptor and labeling descriptor indicated in the stream are out of scope of this Recommendation | International Standard.

U.1.2 Notation

This annex makes extensive use of variable-length, where field length is specified prior to the field itself. An additional short-hand notation is used to improve readability in these cases: length field names are referenced within the "number of bits" column of syntax tables. The alias name for the length field is provided in parenthesis in non-bold font at the same line as the length field, and the number of bits is given as a function of that field.

In the example below, SFL is an alias for the `some_field_length_qwords` field. As the latter can have values of 0..3, `some_field` can have lengths of 0, 64, 128 and 192 bits. Stating that `some_field` is a 0-bit field implies that `some_field` is not present (in the example below this would result in `some_structure()` being a 1-byte structure).

Table U-1 – Variable field length notation example

Syntax	No. bits	Mnemonic
<code>some_structure {</code>		
some_field_length_qwords (SFL)	2	uimsbf
reserved	6	bslbf
some_field	SFL*64	uismbf
<code>}</code>		

The full notation of the same structure is given in Table U-1bis.

Table U-1bis – Table U-1 in equivalent full notation

Syntax	No. bits	Mnemonic
<code>some_structure {</code>		
some_field_length_qwords	2	uimsbf
reserved	6	bslbf
if (some_field_length_qwords > 0) {		
if (some_field_length_qwords == 1) {		
some_field	64	uismbf
}		
if (some_field_length_qwords == 2) {		
some_field	128	uismbf
}		
if (some_field_length_qwords == 3) {		
some_field	192	uismbf
}		
}		
<code>}</code>		

U.1.3 Annex References

- ANSI/SCTE 35 2016, *Digital Program Insertion Cueing Message for Cable*.

In clause U.3 renumber the table caption “Table U-1 TEMI Location Descriptor” to “Table U-2” in AMD1_Cor2

In clause U.3 renumber the table U-2 with the following and replace table caption to Table U-3—AF Descriptor Tags”:

Table U-3 – AF Descriptor Tags

AF Descriptor Tag	Identification
0x00-0x03	Rec. ITU-T H.222.0 ISO/IEC 13818-1 Reserved
0x04	Timeline Descriptor
0x05	Location Descriptor
0x06	BaseURL Descriptor
0x07	Cets_byte_range_descriptor (see NOTE 1)
0x08	MPEG-H_3dAudio_extStreamID_descriptor
0x09	AF_MPEG-H_3dAudio_multi-stream_descriptor
0x0A	AF_MPEG-H_3dAudio_command_descriptor
0x0B	Boundary Descriptor
0x0C	Labeling Descriptor
0x0D-0x7F	Rec. ITU-T H.222.0 ISO/IEC 13818-1 Reserved
0x80-0xFF	User Private

In clause U.3.2 renumber table caption Table U-3 to Table U-4

In clause U.3.3 renumber table caption U-4 to Table U-5

In clause U.3.3 renumber table caption Table U-5 to Table U-6

In clause U.3.4 renumber table caption Table U-6 to Table U-7

In clause U.3.6 renumber table caption Table U-7 to Table U-8

Add the following to the end of Annex U:

U.3.11 Boundary Descriptor

The Boundary Descriptor is used to indicate segment boundary information that can be used to support content partitioning requirements in different types of applications. When the descriptor is carried explicitly within an elementary stream, it indicates the frame accurate location of that boundary. Information carried in the boundary descriptor describes the boundary type, e.g. SAP type, partition and an optional sequence number of the partitioned virtual segment. When it is used together with the TEMI time descriptor, it may also be used to indicate a wall clock timestamp on the boundary point.

Partitions that are indicated by the Boundary Descriptor are a set of continuous segments divided by boundary descriptors within a media stream. A stream can be partitioned in several ways. For example, Partition A corresponds to 2-second segment, while Partition B corresponds to 5-second segment, both share a common boundary point every 10 seconds.

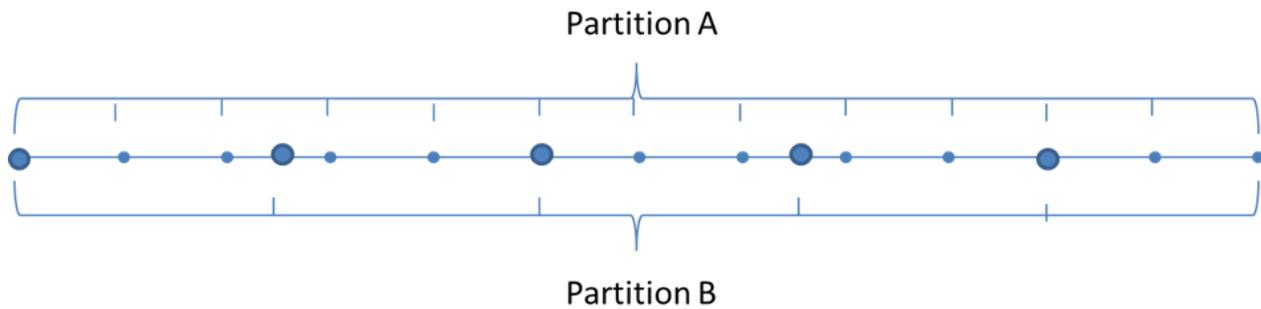


Figure U-1 – Stream partitioning into 2 and 5 second segments

Table U-9 – Boundary Descriptor

Syntax	No. of bits	Format
boundary_descriptor() {		
af_descr_tag	8	uimbsf
af_descr_length	8	uimbsf
num_partitions_minus_1	3	uimbsf
SAP	3	uimbsf
concealment_flag	1	bslbf
reserved	1	"0"
for (i=0; i <= num_partitions_minus_1; i++) {		
partition_id	3	uimbsf
partition_info_flag	1	bslbf
reserved	4	bslbf
if (partition_info_flag == 1) {		
sequence_number_length_code (SNL)	2	bslbf
reserved	6	uimbsf
sequence_number	see SNL	uismbf
}		
}		
}		

U.3.12 Semantic definition of fields in boundary descriptor

SAP: SAP type, as defined 14496-12.

num_partitions_minus_1: Number of partitions to which this boundary belongs. By default (i.e., if this field is zero), each descriptor corresponds to a single partition.

partition_id: partition identifier

partition_info_flag: if set to 1 and SNL is not equal to zero, then extended partition information is available.

NOTE 2 – If *partition_info_flag* is set to 1 and SNL is equal to zero, then this is reserved for future extension points.

concealment_flag: Indicates if the source stream may have lost a frame that was at a boundary point. If flag is set to 1, indicates this is a repair boundary, inserted due to a previous lost boundary point frame. If set to 0, indicates no repair was done to the stream. “Repair” is meant to create a boundary point where it may have been skipped due to a lost boundary point frame.

sequence_number_length_code: specifies length of *sequence_number* field. The width values have the following semantics:

Table U-10 – sequence_number_length_code interpretation

sequence_number_length_code	length, bits
0	0
1	16
2	32
3	64

sequence_number: field providing a unique identifier for a segment within the context of this partition and multiplex. Length of this field is defined in *sequence_number_length_code*. If the *sequence_number_length_code* is 0, *sequence_number* field does not appear.

U.3.13 Labeling Descriptor

The Labeling Descriptor is used to carry one or more labels. When the descriptor is carried explicitly within an elementary system, it indicates a frame accurate location for the label or set of labels as a notation to the virtual segment. When used together with a boundary descriptor, it can note the start of an ad, program, chapter, or multi-period asset and carry identifiers (e.g. EIDR, or user defined) in the stream.

Table U-11 – Labeling Descriptor

Syntax	No. of bits	Format
labeling_descriptor() {		
af_descr_tag	8	uimbsf
af_descr_length	8	uimbsf
l = 0		
while (l < af_descr_length)		
{		
label_length_code	3	bslbf
label_type	13	bslbf
l += 2		
if (label_length_code == 7) {		
label_length	8	uimbsf
l++		
}		
}		

Syntax	No. of bits	Format
<pre> label_bytes l += N } } </pre>	N*8	bslbf

NOTE 1 – In table U.11, the value of N is derived from *label_length_code* and *label_length* (if present), as described in U.3.10.

For example, N would be 20 when *label_length_code* value is 6, and 7 when *label_length_code* and *label_length* both have value of 7

U.3.11 Semantic definition of fields in labeling descriptor

label_length_code: length of the label field in bytes, as provided in Table U.12.

Table U-12 – *label_length_code* interpretation

value	length, bytes
0	0 (not present)
1	2
2	4
3	8
4	12
5	16
6	20
7	Explicit value provided in <i>label_length</i>

label_type: type of the label field, as provided in Table U.13.

Table U-13 – *label_type* values

range	Definition
0x00-0xFF	Reserved for MPEG standardization
0x100-0x1FF	Value defined in ANSI/SCTE 35 Table 8-8 (<i>segmentation_type_id</i>) + 0x100
0x200-0x2FF	Value defined in ANSI/SCTE 35 Table 8-7 (<i>upid_type_id</i>) + 0x200
0x300-0xFFF	Reserved for MPEG standardization
0x1000-0x1FFF	User Private types

label_length: length, in bytes, of the label field (i.e., number of *label_bytes*)

label_bytes: bytes carrying the label. Length N is defined by *label_length_code* or *label_length*. Interpretation of these bytes depends on label types.

NOTE 3 – If *label_length_code* is zero, this allows for *label_type* to exist with no *label_bytes*. This can serve as a marker type to the indicated PES frame.