INTERNATIONAL STANDARD

ISO/IEC 13818-1

Fifth edition
2015-07-01

AMENDMENT 1
2015-07-15

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

Part 1: **Systems**

AMENDMENT 1: Delivery of timeline for external data

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

Partie 1: Systèmes

AMENDEMENT 1: Livraison d'un calendrier pour données externes

Reference number
ISO/IEC 13818-1:2015/Amd.1:2015(E)

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).

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).

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.

ISO/IEC 13818-1 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 ITU-T H.222.0 (04/2014).

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

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

Amendment 1

Delivery of timeline for external data

1) Clause 1.2

Add the following references to clause 1.2.3:

– IETF RFC 3986 (2005), *Uniform Resource Identifier (URI): Generic Syntax*.

– IETF RFC 5484 (2009), *Associating Time-Codes with RTP Streams*.

2) Clause 2.4.3.4, Table 2-6

Replace Table 2-6 with the following table:

Table 2-6 – Transport stream adaptation field

Syntax	No. of bits	Mnemonic
adaptation_field() {		
adaptation_field_length	**8**	**uimsbf**
if (adaptation_field_length > 0) {		
discontinuity_indicator	**1**	**bslbf**
random_access_indicator	**1**	**bslbf**
elementary_stream_priority_indicator	**1**	**bslbf**
PCR_flag	**1**	**bslbf**
OPCR_flag	**1**	**bslbf**
splicing_point_flag	**1**	**bslbf**
transport_private_data_flag	**1**	**bslbf**
adaptation_field_extension_flag	**1**	**bslbf**
if (PCR_flag = = '1') {		
program_clock_reference_base	**33**	**uimsbf**
reserved	**6**	**bslbf**
program_clock_reference_extension	**9**	**uimsbf**
}		
if (OPCR_flag = = '1') {		
original_program_clock_reference_base	**33**	**uimsbf**
reserved	**6**	**bslbf**
original_program_clock_reference_extension	**9**	**uimsbf**
}		
if (splicing_point_flag = = '1') {		
splice_countdown	**8**	**tcimsbf**
}		
if (transport_private_data_flag = = '1') {		
transport_private_data_length	**8**	**uimsbf**
for (i = 0; i < transport_private_data_length; i++) {		

Table 2-6 – Transport stream adaptation field

Syntax	No. of bits	Mnemonic
private_data_byte	**8**	**bslbf**
}		
}		
if (adaptation_field_extension_flag = = '1') {		
adaptation_field_extension_length	**8**	**uimsbf**
ltw_flag	**1**	**bslbf**
piecewise_rate_flag	**1**	**bslbf**
seamless_splice_flag	**1**	**bslbf**
af_descriptor_not_present_flag	**1**	**bslbf**
reserved	**4**	**bslbf**
if (ltw_flag = = '1') {		
ltw_valid_flag	**1**	**bslbf**
ltw_offset	**15**	**uimsbf**
}		
if (piecewise_rate_flag = = '1') {		
reserved	**2**	**bslbf**
piecewise_rate	**22**	**uimsbf**
}		
if (seamless_splice_flag = = '1') {		
splice_type	**4**	**bslbf**
DTS_next_AU[32..30]	**3**	**bslbf**
marker_bit	**1**	**bslbf**
DTS_next_AU[29..15]	**15**	**bslbf**
marker_bit	**1**	**bslbf**
DTS_next_AU[14..0]	**15**	**bslbf**
marker_bit	**1**	**bslbf**
}		
if (af_descriptor_not_present_flag = = '0') {		
for (i = 0; i < N; i++) {		
af_descriptor()		
}		
}		
for (i = 0; i < N; i++) {		
reserved	**8**	**bslbf**
}		
}		
for (i = 0; i < N; i++) {		
stuffing_byte	**8**	**bslbf**
}		
}		
}		

3) Clause 2.4.3.5

3.1) af_descriptor_not_present_flag

In 2.4.3.5, add to semantics, after seamless_splice_flag and right before ltw_valid_flag:

af_descriptor_not_present_flag – This 1-bit field when set to '0' signals the presence of one or several af_descriptor() construct in the adaptation header. When this flag is set to '1' it indicates that the af_descriptor() is not present in the adaptation header.

3.2) af_descriptor

In 2.4.3.5, add to semantics, after DTS_next_AU:

af_descriptor may carry one or more descriptors as defined in Annex U. For descriptors carrying information associated with specific access units of an elementary stream, the descriptor applies to the first access unit that starts in the PES packet immediately following this adaptation field. There may be several TS packets carrying no payload before the start of the PES, in which case these descriptors apply to the next TS packet with payload on the same PID.

The adaptation field shall contain only complete af_descriptor() descriptors, i.e., a single descriptor is always contained in a single transport stream packet.

NOTE 5 – The adaptation field should remain relatively small; it is therefore recommended for large descriptors to use PES carriage as defined in Annex U.

4) Clause 2.4.3.7, Table 2-22

Replace Table 2-22 with the following table:

Table 2-22 – Stream_id assignments

stream_id	Note	stream coding
1011 1100	1	program_stream_map
1011 1101	2, 9,10	private_stream_1
1011 1110		padding_stream
1011 1111	3	private_stream_2
110x xxxx		ISO/IEC 13818-3 or ISO/IEC 11172-3 or ISO/IEC 13818-7 or ISO/IEC 14496-3 audio stream number x xxxx
1110 xxxx		Rec. ITU-T H.262 | ISO/IEC 13818-2 or ISO/IEC 11172-2 or ISO/IEC 14496-2 or Rec. ITU-T H.264 | ISO/IEC 14496-10 video stream number xxxx
1111 0000	3	ECM_stream
1111 0001	3	EMM_stream
1111 0010	5	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Annex A or ISO/IEC 13818-6_DSMCC_stream
1111 0011	2	ISO/IEC_13522_stream
1111 0100	6	Rec. ITU-T H.222.1 type A
1111 0101	6	Rec. ITU-T H.222.1 type B
1111 0110	6	Rec. ITU-T H.222.1 type C
1111 0111	6	Rec. ITU-T H.222.1 type D
1111 1000	6	Rec. ITU-T H.222.1 type E
1111 1001	7	ancillary_stream
1111 1010		ISO/IEC14496-1_SL-packetized_stream
1111 1011		ISO/IEC14496-1_FlexMux_stream
1111.1100		metadata stream
1111.1101	8	extended_stream_id
1111 1110		reserved data stream
1111 1111	4	program_stream_directory

stream_id	Note	stream coding
The notation x means that the values '0' or '1' are both permitted and results in the same stream type. The stream number is given by the values taken by the x's. NOTE 1 – PES packets of type program_stream_map have unique syntax specified in 2.5.4.1. NOTE 2 – PES packets of type private_stream_1 and ISO/IEC_13552_stream follow the same PES packet syntax as those for Rec. ITU-T H.262 | ISO/IEC 13818-2 video and ISO/IEC 13818-3 audio streams. NOTE 3 – PES packets of type private_stream_2, ECM_stream and EMM_stream are similar to private_stream_1 except no syntax is specified after PES_packet_length field. NOTE 4 – PES packets of type program_stream_directory have a unique syntax specified in 2.5.5. NOTE 5 – PES packets of type DSM-CC_stream have a unique syntax specified in ISO/IEC 13818-6. NOTE 6 – This stream_id is associated with stream_type 0x09 in Table 2-29. NOTE 7 – This stream_id is only used in PES packets, which carry data from a Program Stream or an ISO/IEC 11172-1 System Stream, in a Transport Stream (refer to 2.4.3.8). NOTE 8 – The use of stream_id 0xFD (extended_stream_id) identifies that this PES packet employs an extended syntax to permit additional stream types to be identified. NOTE 9 – JPEG 2000 video streams (stream_type = 0x21) are carried using the same PES packet syntax as private_stream_1. NOTE 10 – Timeline and External Media Information streams (stream_type = 0x27) are carried using the same PES packet syntax as private_stream_1.		

5) Clause 2.4.4.9

Replace in Table 2-34 the following row:

0x27-0x7E	ITU-T Rec. H.222.0 | ISO/IEC 13818-1 Reserved

with:

0x27	Timeline and External Media Information Stream (see Annex T)
0x28-0x7E	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Reserved

6) Clause 2.6.90, Table 2-105

Replace Table 2-105 with:

Table 2-105– Extension descriptor

Syntax	No. of bits	Mnemonic
Extension_descriptor () {		
descriptor_tag	8	**uimsbf**
descriptor_length	8	**uimsbf**
extension_descriptor_tag	8	**uimsbf**
if (extension_descriptor_tag == 0x02) {		
ObjectDescriptorUpdate()		
}		
else if (extension_descriptor_tag == 0x03) {		
HEVC_timing_and_HRD_descriptor()		
}		
else if (extension_descriptor_tag == 0x04) {		
af_extensions_descriptor ()		
}		
else {		
for (i=0; i<N; i++) {		
reserved	8	**bslbf**
}		
}		
}		

7) Clause 2.6.91

Add the following description for af_extensions_descriptor() immediately after the description for HEVC_timing_and_HRD_descriptor(), and replace Table 2-106 as follows:

af_extensions_descriptor() – This structure is defined in 2.6.99.

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()
5-255	n/a	n/a	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Reserved

8) New clause 2.6.99 and shifting of table numbers

8.1) New clause 2.6.99

Add the following new clause immediately after clause 2.6.98, and shift numbering in subsequent tables in clause 7 accordingly:

2.6.99 AF extensions descriptor

The AF extensions descriptor is used to signal that adaptation field descriptors could be present in the adaptation header of the component, as defined in 2.4.3.5.

NOTE – There may be AF descriptors in an adaptation field of a TS packet even though this descriptor is not set for the component.

Table 2-111 – Adaptation field extension descriptor

Syntax	No. of bits	Mnemonic

af_extensions_descriptor() {
}		

8.2) Renumbering of tables in clause 2

Renumber the tables in clauses 2.7 and 2.14 as follows:

Clause 2.7:

Table 2-104 becomes Table 2-212

Table 2-105 becomes Table 2-213

Table 2-106 becomes Table 2-214

Table 2-107 becomes Table 2-215

Table 2-108 becomes Table 2-216

Table 2-109 becomes Table 2-217

Table 2-110 becomes Table 2-218

Table 2-111 becomes Table 2-219

Clause 2.14:

Table 2-112 becomes Table 2-220

9) Clause 2.14.1, Note 5

In clause 2.14.1 add to Note 5, after "The NAL unit type 24 may be used in a different way by other specifications out of scope of this Specification.":

"When carrying AVC base and SVC enhancement layers in different elementary streams, usage of VDRD is strongly recommended if access units are not aligned with PES packets."

10) New Annex U

Add the following new Annex U after Annex T:

Annex U

Carriage of timeline and external media information over MPEG-2 transport streams

(This annex forms an integral part of this Recommendation | International Standard.)

U.1 Introduction

This annex specifies a format for carriage of timeline and location of external media resource that may be used as a synchronized enhancement of an MPEG-2 transport stream. The possible resolving, consumption and rendering of external media indicated in the stream are out of scope of this Recommendation | International Standard.

The format specifies the mapping of the transport stream program clock to an embedded timeline, the signalling of associated external resources, hereafter called add-on(s), and the signalling of prefetching events. The format is designed to be compact in order to fit within one TS packet for common use cases. The mapping of the embedded timeline indicated in the PES packet payload or in the adaptation field descriptor with the PTS value of the PES header of the PES packet provides a stable timeline for media streams in the program, regardless of PCR discontinuities or other timestamps rewriting that may happen in the network.

In the context of this annex, the "timeline and external media information" stream is called TEMI stream.

The TEMI stream describes external data and associated timing for the program in the MPEG-2 transport stream with which the TEMI stream is associated through the program map table.

U.2 TEMI access unit and TEMI elementary stream

The format of the TEMI access unit is defined in Table U.1. TEMI access units shall be carried as PES packets using private_stream_1 streamID and identified in the program map table by the stream type 0x26. There shall be at most one TEMI elementary stream declared in the program map table.

The payload of a TEMI PES packet is a single complete TEMI_AU, i.e., there shall be one and only one complete TEMI access unit in a TEMI PES packet.

The TEMI PES packet header shall contain a PTS timestamp, whose value is used to match the current system time clock with the timeline value embedded in the TEMI packet payload, as defined in Table U.1.

A TEMI_AU is made of one or several AF descriptors. These AF descriptors may be sent in different access units and at different rates, and are independently decodable. All TEMI access units are therefore random access points.

NOTE 1 – In order to avoid interpolation issues when frame-accurate synchronization is required, the indicated PTS should be the same as the PTS of the associated video or audio stream for which frame accurate sync is needed.

NOTE 2 – It is possible to perform timeline interpolation in-between TEMI access units, for example if multiple audio frames are packed in a single PES packet, or when the TEMI AU frequency is less than the media AU frequency. However, receivers detecting PCR discontinuities in-between TEMI AUs should be careful when performing interpolation.

Table U.1 – TEMI access unit

Syntax	No. of bits	Mnemonic
`TEMI_AU {`		
`CRC_flag`	**1**	**bslbf**
`reserved`	**7**	**bslbf**
`for (i=0; i<N; i++) {`		
`af_descriptor();`		
`}`		
`if (CRC_flag) {`		
`CRC_32`	**32**	**rpchof**
`}`		
`}`		

Each TEMI AU is composed of an entire number of AF descriptors.

CRC_flag – A 1-bit flag, which when set to '1' indicates that a CRC field is present in the packet.

CRC_32 – This is a 32-bit field that contains the CRC value that gives a zero output of the registers in the decoder defined in Annex A after processing the entire payload of the TEMI access unit.

U.3 AF descriptors

U.3.1 Introduction

AF descriptors are structures used to carry various features of the timeline or other information. All AF descriptors have a format that begins with an 8-bit tag value. The tag value is followed by an 8-bit AF descriptor length and data fields. The following semantics apply to the descriptors defined throughout Annex U.

af_descr_tag – The af_descr_tag is an 8-bit field that identifies each AF descriptor.

Table U.2 provides the Rec. ITU-T H.222.0 | ISO/IEC 13818-1 defined, Rec. ITU-T H.222.0 | ISO/IEC 13818-1 reserved, and user available AF descriptor tag values.

af_descr_length – The af_descr_length is an 8-bit field specifying the number of bytes of the AF descriptor immediately following af_descr_length field.

Table U.2 – 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
0x07-0x7F	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Reserved
0x80-0xFF	User Private

AF descriptors may be carried in the adaptation field of TS packets of a media elementary stream, as defined in 2.4.3.5.

U.3.2 Location descriptor

The location descriptor is used to signal the location of external data that can be synchronized with the program. It conveys several locations and their type (optionally including MIME types), along with the ability to signal upcoming external data association though a countdown until activation of the external data. It is possible to signal splicing of external data, by signalling that the newly associated data is temporary and the previous association will be re-used later on.

Table U.3 – TEMI location descriptor

Syntax	No. of bits	Mnemonic
temi_location_descriptor {		
af_descr_tag	**8**	**uimsbf**
af_descr_length	**8**	**uimsbf**
force_reload	**1**	**bslbf**
is_announcement	**1**	**bslbf**
splicing_flag	**1**	**bslbf**
use_base_temi_url	**1**	**bslbf**
reserved	**5**	**bslbf**
timeline_id	**7**	**uimsbf**
if (is_announcement) {		
timescale	**32**	**uimsbf**
time_before_activation	**32**	**uimsbf**
}		
if (!use_base_temi_url) {		
url_scheme	**8**	**uimsbf**
url_path_length	**8**	**uimsbf**
for (i=0;i< url_path_length;i++) {		
url_path	**8**	**bslbf**
}		
}		
nb_addons	**8**	**uimsbf**
for (i=0;i < nb_addons ;i++) {		
service_type	**8**	**uimsbf**
if (service_type==0) {		
mime_length	**8**	**uimsbf**
for (j=0;j<mime_length;j++) {		
mime_type	**8**	**bslbf**
}		
}		
url_subpath_len	**8**	**uimsbf**
for (j=0;j< url_subpath_len;j++) {		
addon_location	**8**	**bslbf**
}		
}		
}		

U.3.3 Semantic definition of fields in location descriptor

`force_reload`: When set to 1, indicates that the add-on description shall be reloaded before attempting to map media times or locate media components. Reloading may typically happen for manifest-based add-on such as MPEG-DASH or MPEG-MMT.

`is_announcement`: When set to 1, indicates that the add-on described by this descriptor is not yet active.

`splicing_flag`: When set to 1, indicates that the new add-on indicated by this descriptor temporarily interrupts the last defined add-on for which `splicing_flag` was not set. It is possible to have a sequence of add-ons with `splicing_flag` set. This allows terminal to optimize loading of the add-on when splicing period ends. There shall not be two `temi_location_descriptor` pointing to the same add-on with different values for `splicing_flag`, unless another `temi_location_descriptor` pointing to different add-ons is sent in-between with a `splicing_flag` set to 0.

`url_scheme`: Indicates the URL scheme to use for the URL. The scheme identified shall be appended to the `url_path`, according to Table U.4

Table U.4 – TEMI URL scheme types

TEMI URL Scheme Type	Scheme value
0	Scheme URL is Included in url_path
1	"http://"
2	"https://"
3-0x7F	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Reserved
0x80-0xFF	User private

`timeline_id`: A unique identifier for this content location. If `force_reload` is set to '0' and another `temi_location_descriptor` with the same `timeline_id` and `splicing_flag` has already been received, the associated descriptions of the two descriptors shall be the same. If the `splicing_flag` differs for the same `timeline_id`, the `timeline_id` is reassigned to the new URL defined in the descriptor (i.e., redefinition of `timeline_id`).

`timescale`: Indicates the timescale used to express the `time_before_activation` field in this message.

`use_base_temi_url`: When set to 1, indicates that the URL defined in the last received `temi_base_url_descriptor` shall be used as a base URL; when set to 0, a base URL is provided in the payload of this descriptor for the location described in this descriptor and only this descriptor.

`time_before_activation`: Indicates the time in timescale units until the resource identified by `addon_location` becomes active; the ratio `time_before_activation/timescale` indicates a duration in seconds. An implementation may use this information to start prefetching content.

`url_path_length`: Indicates the length in bytes of the base URL path; when set to 0, indicates an empty URL path.

`url_path`: Base URL common to the different add-ons, if any; it shall be encoded without trailing zero character. This URL shall be a valid URL, as defined in clause 3 of IETF RFC 3986, and may contain a Fragment and or a Query part.

`nb_addons`: Indicates the number of add-ons that share this timeline. If 0, only one add-on is present at the location indicated by `url_path`, if this string is not empty. If `url_path` is empty and `nb_addons` is 0, this means that no service is associated with the current broadcast. If `url_path` is empty and `nb_addons` is not 0, `url_subpath_len` must be greater than 0.

`service_type`: Indicates the type of add-on present at the given URL, as described in table U-5. An implementation can decide to fetch or not the add-on based on this service type indication.

Table U.5 – TEMI service types

TEMI Service Type	Add-on type
0	Specified with MimeType
1	MPEG-DASH
2	ISO/IEC 14496-12 file
3	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Transport Stream
0x04-0x7E	Rec. ITU-T H.222.0 | ISO/IEC 13818-1 Reserved
0x7F	Unknown service type
0x80-0xFF	User Private

`mime_type`: Indicates the mime type of the add-on available at the indicated location, as defined in IETF RFC 2046. An implementation can decide to fetch or not the add-on based on this mime type indication.

`url_subpath_length`: Indicates the length in bytes of the URL sub path; when set to 0, indicates an empty URL subpath.

`url_subpath`: Indicates the URL sub path, without trailing zero character; this URL shall be a valid URL, as defined in clause 3 of IETF RFC 3986. The URL for this add-on is obtained by merging `url_subpath` with the base URL path, as defined in clause 5 of IETF RFC 3986.

U.3.4 Base URL Descriptor

The base URL descriptor is used to assign a default base URL to all location descriptors.

Table U.6 – TEMI base URL descriptor

Syntax	No. of bits	Mnemonic
temi_base_url_descriptor {		
af_descr_tag	**8**	**uimsbf**
af_descr_length	**8**	**uimsbf**
url_scheme	**8**	**uimsbf**
for (i=0;i< N;i++) {		
base_url_path	**8**	**bslbf**
}		
}		

U.3.5 Semantic definition of fields in location descriptor

`url_scheme`: Indicates the URL scheme to for the URL. The scheme identified shall be appended to the `base_url_path`, according to Table U.4.

`base_url_path`: Base URL common to all following location descriptors, if any; it shall be encoded without trailing zero character. This URL shall be a valid path, as defined in clause 3 of IETF RFC 3986, and may contain a Fragment and or a Query part.

U.3.6 Timeline descriptor

The Timeline descriptor is used to carry timing information that can be used to synchronize external data. When the descriptor is carried within a TEMI access unit, the included timing information is given for the PTS value of the TEMI access unit carrying the descriptor. When the descriptor is carried in the adaptation field of a media component, the included timing information is given for the PTS found in the PES header starting in the payload of this transport stream packet or in the first subsequent transport stream packet with `payload_unit_start_indicator` set to 1 on this component (same PID). This PES header shall have a PTS declared. For a given media access unit, there shall be at most one `temi_timeline_descriptor` for which the last `temi_location_descriptor` received had an `is_announcement` flag set to 0. A `temi_timeline_descriptor`, for which the last `temi_location_descriptor` received had an `is_announcement` flag set to 1, indicates the media time at which the timeline will start upon activation. This Recommendation | International Standard does not define any restrictions on `temi_timeline_descriptor` using `timeline_id` values in the range [0x80, 0xFF].

In this section, this media PES packet is called the associated PES packet and the media PTS value is called the associated PTS.

Table U.7 – TEMI timeline descriptor

Syntax	No. of bits	Mnemonic
temi_timeline_descriptor {		
af_descr_tag	8	uimsbf
af_descr_length	8	uimsbf
has_timestamp	2	uimsbf
has_ntp	1	bslbf
has_ptp	1	bslbf
has_timecode	2	uimsbf
force_reload	1	bslbf
paused	1	bslbf
discontinuity	1	bslbf
reserved	7	bslbf
timeline_id	8	uimsbf
if (has_timestamp) {		
timescale	32	uimsbf
if (has_timestamp==1) {		
media_timestamp	32	uimsbf
} else if (has_timestamp==2) {		
media_timestamp	64	uimsbf
}		
}		
if (has_ntp) {		
ntp_timestamp	64	uimsbf
}		
if (has_ptp) {		
ptp_timestamp	80	uimsbf
}		
if (has_timecode) {		
drop	1	bslbf
frames_per_tc_seconds	15	uimsbf
duration	16	uimsbf
if (has_timecode==1) {		
short_time_code	24	uimsbf
} else if (has_timecode==2) {		
long_time_code	64	uimsbf
}		
}		
}		

U.3.7 Semantic definition of fields in location descriptor

has_timestamp: Indicates a media timestamp will be carried in this descriptor, and indicates its type. Value 0 means no media timestamp is present, value 1 means a 32 bit media timestamp is present, value 2 means a 64 bit media timestamp is present, value 3 is reserved.

has_ntp: When set to 1, indicates that a NTP timestamp will be carried in this descriptor.

has_ptp: When set to 1, indicates that a PTP timestamp will be carried in this descriptor.

has_timecode: When set to '00', indicates that no frame timecode is present, when set to '01' indicates a short frame timecode is present, when set to '10' indicates a long frame timecode is present, value '11' is reserved.

force_reload: When set to 1, indicates that add-on description shall be reloaded before attempting to map media times or locate media components. Reloading typically happens for manifest-based add-on such as MPEG-DASH or MPEG-MMT.

paused: When set to 1, indicates that the timeline identified by timeline_id is currently paused; this typically happens when a timeline has to be paused but no splicing timeline is to be inserted during the pause. When a timeline is running, all other timelines defined are implicitly in pause mode.

discontinuity: When set to 1, indicates that a discontinuity has happened in the timeline. If set to 0, no discontinuity happened since the last received `temi_timeline_descriptor` with the same value of `timeline_id` and same value of `splicing_flag`, if defined.

NOTE – An implementation may use this information to optimize playback of add-on content.

timeline_id: Indicates the active timeline. timeline_id values in the range [0, 0x7F] are identified in a temi_location_descriptor; for such values of timeline_id, the content of this temi_timeline_descriptor shall be ignored if no temi_location_descriptor with the same timeline_id has been received. timeline_id values in the range [0x80, 0xFF] identify timelines defined by means beyond the scope of this Specification.

timescale: Indicates the timescale used to express the media_timestamp field in this message.

media_timestamp: Indicates the media time in timescale units corresponding to the PES PTS value of this packet for the timeline identified by the last temi_location_descriptor received. The timeline may be interpolated between two temi_timeline_descriptor: let PTS_0 be the associated PTS of the temi_timeline_descriptor carrying media time MTA_0; until a new media timeline packet is received, the PTS of subsequent PES packets of other PIDs in this program is mapped to the TEMI timeline as follows:

$$MT_i = (PTS_i - PTS_0) / 90000 + MTA_0/timescale$$

ntp_timestamp: A full 64 NTP timestamp as defined in clause 6 of IETF RFC 5905. The timeline may be interpolated between two temi_timeline_descriptor: let PTS_0 be the associated PTS of the temi_timeline_descriptor carrying NTP timestamp NTP_0; until a new media timeline packet is received, the PTS of subsequent PES packets of other PIDs in this program is mapped to the NTP time NTP_i as follows:

$$NTP_i = (PTS_i - PTS_0) / 90000.0 + NTP_0$$

ptp_timestamp: A full 80 bits PTP timestamp as defined in IEEE 1588v2. The timeline may be interpolated between two temi_timeline_descriptor: let PTS_0 be the associated PTS of the temi_timeline_descriptor carrying PTP timestamp PTP_0; until a new media timeline packet is received, the PTS of subsequent PES packets of other PIDs in this program is mapped to the PTP time PTP_i as follows:

$$PTP_i = (PTS_i - PTS_0) / 90000.0 + PTP_0$$

drop: Drop-frame indication, as defined in clause 5 of IETF RFC 5484.

frames_per_tc_second: The number of those frames that make a time-code second, as defined in clause 5 of IETF RFC 5484.

duration: The duration in ticks of a frame expressed in the timescale of 90000 ticks per seconds, as defined in clause 5 of IETF RFC 5484.

short_time_code: A short 32 time code as defined clause 6.2 of IETF RFC 5484.

long_time_code: A full 64 time code as defined clause 6.2 of IETF RFC 5484.

short_time_code and long_time_code indicate the media time of the first access unit starting in the payload of the associated PES. Using the information of drop, duration and frames_per_tc_seconds, it is possible to interpolate the timing between two temi_timeline_descriptor.