

| | |
|---|---|
|  <p>STANDARD</p> <p>Adaptive Bit Rate Content Encoding</p> | <p>MISB ST 1910</p> <p>25 June 2020</p> |
|---|---|

1 Scope

This standard defines a file format for adaptive bit rate content encoding of MISP Class 1 Motion Imagery and a subset of Class 2 Motion Imagery for applications within the NSG. With constraints as defined in this document this standard mandates the Common Media Application Format (CMAF) [1] container for encoding and packaging segmented media – Motion Imagery, Audio, and Metadata for delivery via server/client-based protocols.

This standard supports H.264/AVC [2] and H.265/HEVC [3] compressed imagery and Key-Length-Value (KLV) metadata consistent with the Motion Imagery Standards Profile (MISP) [4].

This standard provides guidelines to map media content from an MPEG-2 Transport Stream (TS) to the Common Media Application Format (CMAF) based on ISO/BMFF, fMP4 container.

This standard does not address the media delivery method from a server to an end-user client. Delivery options such as MPEG-DASH [5] and HLS [6] are specific to application and implementation requirements, and therefore, outside the scope of this document.

2 References

- [1] ISO/IEC 23000-19:2020 Information technology - Multimedia application format (MPEG-A) - Part 19: Common media application format (CMAF) for segmented media.
- [2] ISO/IEC 14496-10:2014 Information Technology - Coding of audio-visual objects - Part 10: Advanced Video Coding.
- [3] ISO/IEC 23008-2:2017 Information Technology - High efficiency coding and media delivery in heterogeneous environments - Part 2: High efficiency video coding.
- [4] MISB MISP-2020.3 Motion Imagery Standards Profile, Jun 2020.
- [5] ISO/IEC 23009-1 :2019 Information technology - Dynamic adaptive streaming over HTTP (DASH) - Part 1: Media presentation description and segment formats.
- [6] IETF RFC 8216 HTTP Live Streaming 2nd Edition, Apr 2020.
- [7] MISB ST 0601.16 UAS Datalink Local Set, Oct 2019.
- [8] MISB ST 1001.1 Audio Encoding, Feb 2014.
- [9] ISO/IEC 14496-12:2015 Information technology - Coding of audio-visual objects - Part 12: ISO base media file format.
- [10] ITU-R BT.709-6 Parameter values for the HDTV standards for production and international programme exchange, 06 2015.

- [11] MISB RP 0802.2 H.264/AVC Motion Imagery Coding, Feb 2014.
- [12] MISB ST 1402.2 MPEG-2 Transport Stream for Class 1/Class 2 Motion Imagery, Audio and Metadata, Oct 2016.
- [13] MISB ST 0604.6 Timestamps for Class 1/Class 2 Motion Imagery, Oct 2017.
- [14] ISO/IEC 13818-1:2018 Information technology - Generic coding of moving pictures and associated audio information: Systems.
- [15] MISB ST 0603.5 MISB Time System and Timestamps, Oct 2017.

3 Revision History

| Revision | Date | Summary of Changes |
|----------|------------|--------------------|
| ST 1910 | 06/25/2020 | • Initial release |

4 Terms and Definitions

Encoding Ladder Content encoded into a variety of spatial resolutions, temporal resolutions, and bitrates designed to facilitate adaptation to changing network conditions. Different applications typically will choose a different encoding ladder as a function of compression type, content, and client device (i.e., computer, smartphone, etc.)

5 Acronyms

| | |
|----------------|---|
| AAC | Advanced Audio Codec |
| AVC | Advanced Video Coding |
| ABR | Adaptive Bit Rate |
| CMAF | Common Media Application Format |
| CTE | Chunked Transfer Encoding |
| DASH | Dynamic Adaptive Streaming over HTTP |
| GOP | Group-of-Pictures |
| HEVC | High Efficiency Video Coding |
| HLS | HTTP Live Streaming |
| HTTP | Hypertext Transfer Protocol |
| IBMF | ISO Base Media File format |
| IDR | Instantaneous Decoding Refresh |
| IETF | Internet Engineering Task Force |
| ISO/IEC | International Standards Organization/ International Electrotechnical Commission |
| ITU-R | International Telecommunication Union Radiocommunication Sector |
| KLV | Key Length Value |
| MISB | Motion Imagery Standards Board |
| MISP | Motion Imagery Standards Profile |
| MPD | Media Presentation Description |
| MPEG | Moving Picture Experts Group |
| NSG | National System for Geospatial-Intelligence |

| | |
|------------|-------------------------|
| OTT | Over-The-Top |
| PID | Packet ID |
| PMT | Program Map Table |
| RP | Recommended Practice |
| ST | Standard |
| TS | MPEG-2 Transport Stream |
| ULL | Ultra-Low Latency |
| URN | Uniform Resource Name |

6 Introduction

Adaptive Bit Rate (ABR) streaming is a media-streaming model for delivery of media content controlled by the client. Commercial services adopted ABR to satisfy the growing demand for over-the-top (OTT) delivery of content to consumers using various device types for content playback (e.g., computers, mobile devices, televisions). Consumer demand for content on various device types has rapidly matured network infrastructures, and server/client capabilities allowing ABR technologies to become mainstream. The convergence in the industry to adopt CMAF as the container file for ABR content provides a unifying model to support delivery of multimedia presentations to a variety of devices, such as set-top boxes and web browsers, using various means for streaming delivery such as MPEG-DASH and HLS.

ABR operates within a server/client relationship as shown in Figure 1. All CMAF resources composing a service are accessible through multiple HTTP requests. Standard HTTP 1.1 – or higher servers and caching proxies host and distribute media content to ABR clients. Figure 1 shows the architecture for services in ABR streaming. *CMAF Content Preparation* packages media content into one or more CMAF files, which reside on a web server (i.e., *ABR Server*).

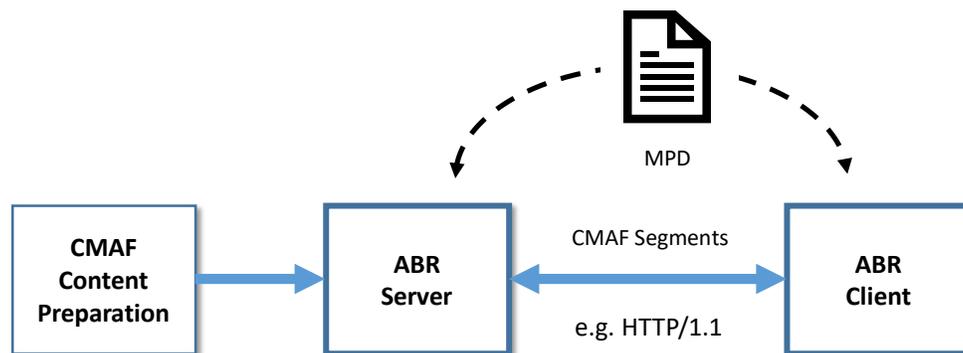


Figure 1: Architecture for ABR Delivery

A Media Presentation Description (*MPD*) manifest file (e.g., an XML formatted “instruction” file) provides the necessary information for an *ABR Client* to request, receive, process, and display the content. The client parses the *MPD* and makes further requests based on its environment (network bandwidth, client rendering capabilities, etc.). As the network conditions between a server and client may change over time, the client adapts its requests to meet new conditions.

Evolved from the MPEG-DASH specification [5], CMAF constrains the IBMF container further and provides for a unified MPD protocol to support both DASH and HLS applications. Developed principally for compressed video and audio, recent efforts have expanded its capabilities to carry additional types of media such as closed-captioning, subtitles, advertisements, etc. As these enhancements continue to mature, the NSG can leverage this technology to provide cloud-based processing and analytics to resource-constrained clients. This document leverages the baseline capabilities of CMAF with additional constraints to facilitate interoperability for NSG applications.

CMAF permits one media type per file. A video or audio encoding requires its own CMAF file. Adapting CMAF for government application requires considering in-band security information for the encoded content. MISB has long supported tight binding of security information with content. This document describes the packaging of metadata, to include security information, along with Motion Imagery unified within one CMAF file.

Figure 2 illustrates the source content-to-ABR delivery workflow and indicates the scope of the guidance provided by this standard. Section 8 details the organization of content in a CMAF file, while Section 8.1.2 provides detailed information on the Encoding Ladder.

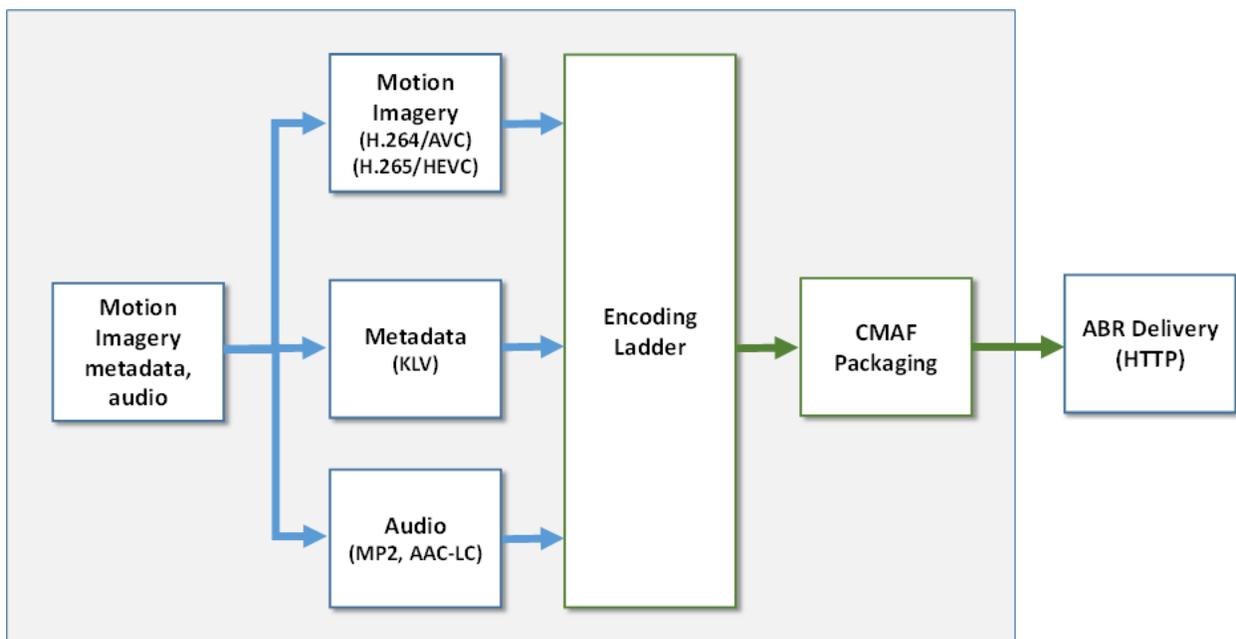


Figure 2: Source Content mapped to CMAF

This document specifies requirements on Motion Imagery, Audio, and Metadata encapsulated in a CMAF file as allowed by the CMAF specification and approved by the MISB. Specifically, this document indicates the supported profiles and levels for H.264/AVC and H.265/HEVC compressed Motion Imagery types. MISB ST 0601 [7] exemplifies the anticipated type of metadata in ABR applications. This document also indicates allowed compressed audio formats consistent with those supported in CMAF and those identified in MISB ST 1001 [8].

7 CMAF Overview

7.1 Content Encoding/Packaging

Figure 3 delineates the scope of the CMAF specification versus the ancillary streaming components which define the manifest files and resources for a CMAF presentation.

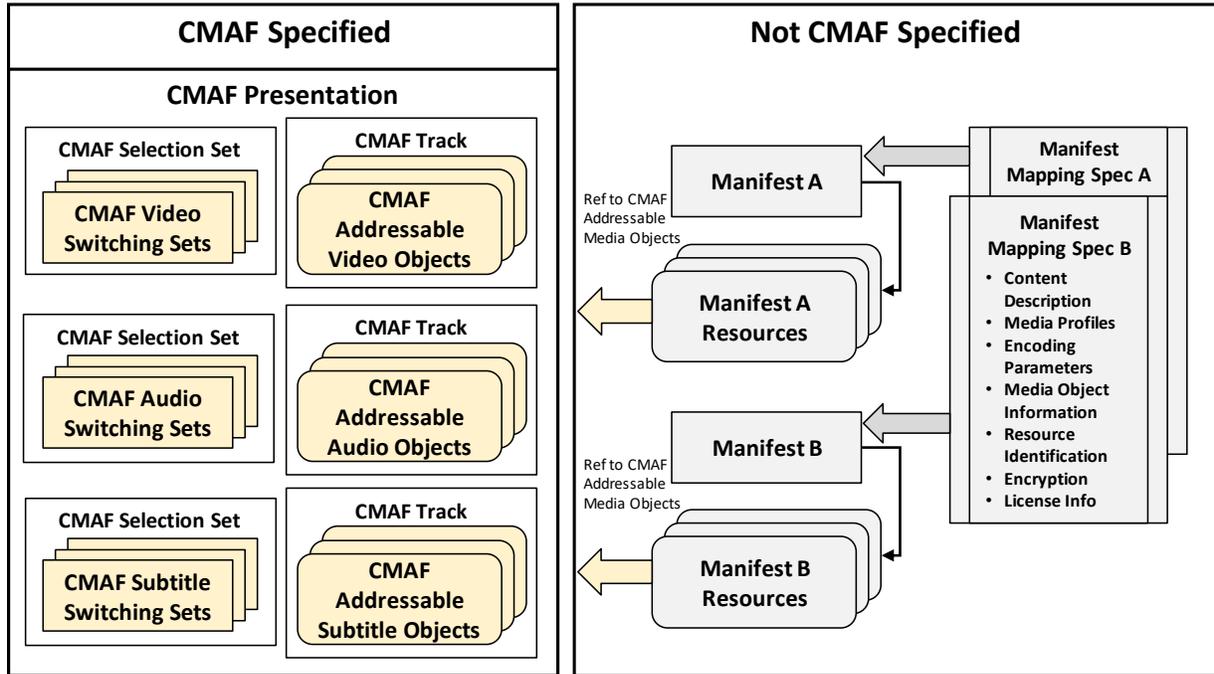


Figure 3: CMAF Application Model

To represent a CMAF presentation, a manifest file generated by the CMAF packager describes the CMAF track relationships, encoding constraints, and other parameters needed for content delivery. A client requests the manifest file to orchestrate a CMAF presentation by choosing various CMAF tracks of content encoded to different bitrates, spatial or temporal densities (i.e., a CMAF *Switching Set*) through *Addressable Media Objects* provided by the server. CMAF limits one media type per track; that is, one CMAF file per media essence type. CMAF switching sets (shown in Figure 4) aligned end-to-end provide seamless switching between the alternative bitrate and spatial/temporal encodings and constrained to enable seamless track switching

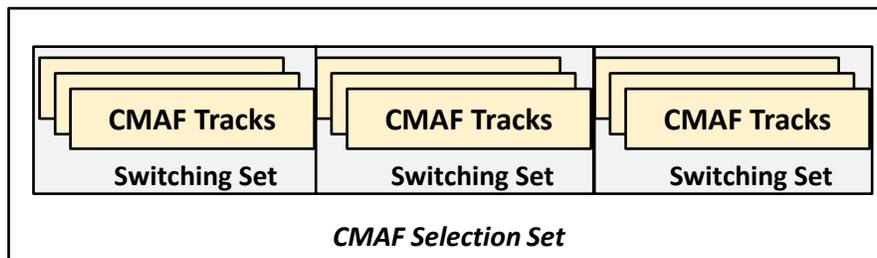


Figure 4: Multiple encodings of media content aligned as switching sets

A CMAF *track* as shown in Figure 5 is a continuous sequence of one or more CMAF *fragments* in presentation order conforming to a CMAF media profile and an associated CMAF *header*, which facilitates the decoding of the fragments. Each CMAF track represents encoding of one media stream; thus, three different encodings of the same content require three different CMAF tracks.

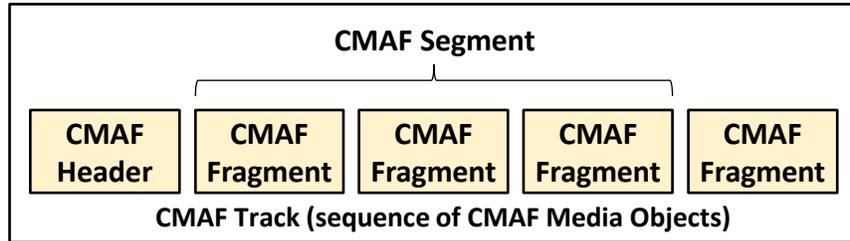


Figure 5: CMAF Track Composition

A CMAF *segment* is an addressable media object consisting of one or more CMAF fragments from the same CMAF track. Each CMAF media profile further constrains CMAF fragments and media sample data.

In common practice segmented video is “Closed GOP” encoded when generating switching sets. This ensures smooth switching between switching sets. A typical switch point is an I or IDR frame in compressed video. A CMAF *Selection Set* is one or more CMAF switching sets where each CMAF switching set encodes an alternative aspect of the same presentation over the same period, only one of which plays at a time. A switching set allows changes in frame size, frame rate (if it does not interfere with fragment temporal length and start times), encoding parameters, etc.

Figure 6 shows a more detailed organization of a CMAF file.

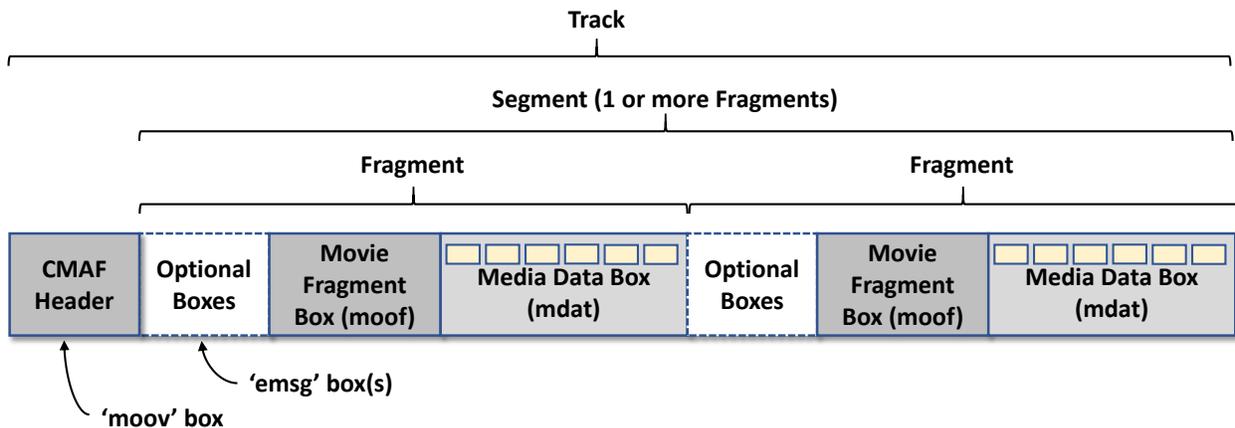


Figure 6: Organization of a CMAF file

A CMAF track starts with a CMAF header followed by a continuous sequence of one or more CMAF fragments stored in presentation order. The CMAF header contains a MovieBox or moov

for processing and presenting CMAF fragments in a CMAF track. Following the moov box are one or more “optional boxes” allowing additional functionality. Specific to this document is the emsg box (discussed later) as an optional box for containerizing metadata. The CMAF specification imposes additional constraints for the moov box described in the IBMF specification [9]. CMAF constrains an IBMF file to only one track per media essence type: thus, for example, one video encoding per CMAF file, one audio encoding per CMAF file, etc.

7.1.1 Low Latency CMAF

Typically, a CMAF fragment consists of at least a *Movie Fragment Box* (moof) and *Media Data Box* (mdat) pair. To support ultra-low-latency (ULL) delivery, such as live presentations, CMAF further allows a decomposition of a fragment into smaller CMAF *chunks* as shown in Figure 7. The CMAF specification requires a moof box per CMAF chunk. In this document the use of the emsg box requires each chunk to also contain an emsg box when supplying metadata for that media chunk. While CMAF chunks enable lower latency delivery, the additional moof boxes do introduce additional overhead to the file.

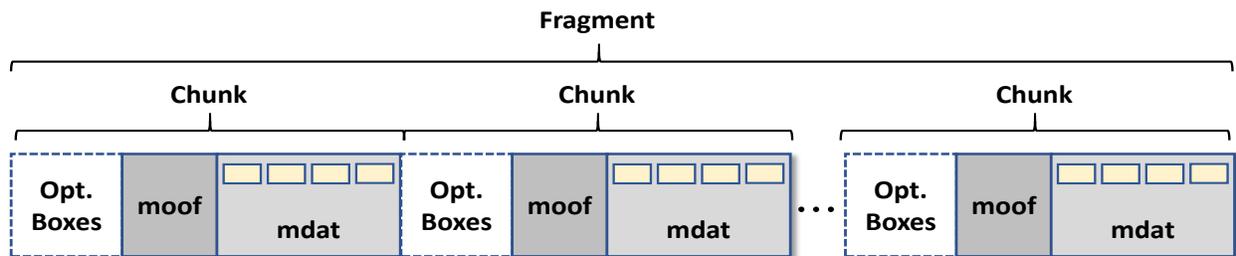


Figure 7: CMAF fragment decomposed into CMAF chunks

A CMAF fragment then consists of multiple CMAF chunks each of which is a pair of moof and mdat boxes with optional emsg box(s). Chunks can theoretically be very small (e.g., 200 milliseconds), but practically achieving low latency through the transfer of small data units requires optimized infrastructure across a network to realize this benefit. Table 1 summarizes the hierarchy of data units in CMAF.

Table 1: Hierarchy of Segments, Fragments and Chunks

| Type | Organization | Application |
|----------|---|-------------|
| segment | One or more fragments combine to form a segment | common use |
| fragment | One or more chunks combine to form a fragment | common use |
| chunk | The smallest referenceable media unit | low latency |

7.2 Metadata and Event Message Box

Introduced in the MPEG-DASH specification, the event message facilitates augmenting content with non-media type information like subtitles. Events are messages having type, timing, and optional payload. The Event Message (emsg) box structure holds event message information. All emsg boxes pertaining to a segment need to precede the segment; this minimizes the time for a

client to detect and parse them. The emsg box structure, shown in Figure 8, provides signaling for generic events related to the media presentation time.

Version 1 of the `DASHEventMessageBox` (shown in yellow) adds the field `presentation_time`; this makes event message timing independent of box location in the CMAF track, but more importantly allows for aligning information within an emsg to a frame of video in time.

```
aligned(8) class DASHEventMessageBox extends FullBox (emsg, version, flags = 0){
  if (version==0) {
    string      scheme_id_uri;
    string      value;
    unsigned int(32) timescale;
    unsigned int(32) presentation_time_delta;
    unsigned int(32) event_duration;
    unsigned int(32) id;
  } else if (version==1) {
    unsigned int(32) timescale;
    unsigned int(64) presentation_time;
    unsigned int(32) event_duration;
    unsigned int(32) id;
    string      scheme_id_uri;
    string      value;
  }
  unsigned int(8) message_data[];
}
```

Version 1
used for
CMAF
'emsg'

Figure 8: DASHEventMessageBox inband event message data structure

The descriptions for the `DASHEventMessageBox` fields for version 1 given in Table 2 are from the MPEG-DASH and CMAF standards; see [1] for additional requirements on the `DASHEventMessageBox`.

Table 2: Event Message (emsg) Fields

| emsg Field | Description |
|-------------------|--|
| scheme_id_uri | Identifies the message scheme. The owner defines the semantics and syntax of the <code>message_data[]</code> for the scheme identified. The string may use URN or URL syntax. When a URL is used, it is recommended to also contain a month-date in the form <code>mmyyyy</code> ; the assignment of the URL must have been authorized by the owner of the domain name in that URL on or very close to that date. A URL may resolve to an Internet location, and a location that does resolve may store a specification of the message scheme. |
| value | Specifies the value for the event. The owner of the scheme must define the value space and semantics identified in the <code>scheme_id_uri</code> field. |
| timescale | Provides the timescale indicated in the <code>MovieHeaderBox</code> , in ticks per second, for the <code>presentation_time</code> and <code>event_duration</code> fields within this box. |
| presentation_time | Provides the presentation time of the event measured on the CMAF track's presentation timeline, in the timescale declared in its <code>MovieHeaderBox</code> . |

ST 1910 Adaptive Bit Rate Content Encoding

| | |
|-----------------------|---|
| event_duration | Provides the duration of event in media presentation time. The value 0xFFFF indicates an unknown duration. |
| id | A field identifying this instance of the message. Messages with equivalent semantics shall have the same value, i.e. processing of any one event message box with the same id is sufficient. |
| message_data[] | Represents the body of the event message. The owner of the scheme defines the syntax and semantics of this field identified in the scheme_id_uri field. Specific applications and users may define message schemes. |

The CMAF specification recommends version 1. MISB ST 1910 requires version 1.

Figure 9 shows an example emsg box with parameters for **scheme_id_uri** and **value** defined by ST 1910 for MPEG-2 TS mapping (discussed in Section 9). In this example, metadata items from MISB ST 0601 form the contents of the **message_data[]** field.

| Property name | |
|-------------------|---|
| type | emsg |
| size | 363 |
| flags | 0 |
| version | 1 |
| start | 80 |
| timescale | 25000 |
| presentation_time | 0 |
| event_duration | 1000 |
| id | 0 |
| scheme_id_uri | urn:misb:KLV:bin:2019 |
| value | 01FC |
| message_data | 6,14,43,52,2,11,1,1,14,1,3,1,1,0,0,0,130,1,29,2,8,0,5,33, |

Figure 9: Example: ST 1910 emsg box

| Requirement | |
|-------------|--|
| ST 1910-01 | An Event Message shall use version 1 of the MPEG-DASH Event Message structure. |

7.2.1 Event Message Box Frequency

One or more Event Message boxes can represent a given segment, or media sample (note: a media sample defined in CMAF is media data associated with a single decode start time and duration. This definition applies to a frame of imagery, so more than one event message can link

to a frame of Motion Imagery). CMAF requires all Event Message boxes for a given CMAF Fragment to precede the moof box for that fragment.

7.2.2 Event Message Rate Partitioning (Informative)

Rate partitioning is a method for separating multiple sources of sample data into streams of “like” sample rates. For instance, various metadata sources sampled and captured at 10 Hz, 30 Hz, and 50 Hz can be “rate partitioned” into three separate streams with update rates of 10 Hz, 30 Hz and 50 Hz. In such cases, metadata sampling may be on a timeline different than the Motion Imagery. Within CMAF a timeline for the file governs all information in the file as defined by the timescale selected. Choosing a suitable timescale which supports all media and data within the file allows rate partitioning of metadata.

Time defined by the timescale in a CMAF file is in ticks per second. For video, a media sample is one frame. For an Event Message, a data (i.e., metadata) sample is one event message. Choice in timescale needs to account for the highest expected frequency of the media or data. Said differently, the timeline needs enough resolution to represent the smallest unit of time expected for a sample. Video typically runs at a near constant sample (i.e., frame) rate, whereas metadata may be less periodic, more frequent than the frame rate, or need more precise timing than a frame allows.

As an example, a timescale of 60 ticks per second is enough to specify video at 60 frames per second, where the duration for each media sample is one frame (i.e., $60/1 = 60$). Likewise, specifying a timeline of 96000 ticks per second but with a duration of 1600 produces the same result (i.e., $96000/1600 = 60$). A greater timescale value enables sampling data to a finer resolution in time. Choosing a timescale to meet the resolution requirements for the highest frequency signal in the file enables rate partitioning. The timescale does need however to be a multiple of the image frame rate to be evenly divisible.

A client can sort the different rates of metadata based on its Event Message `presentation_time`, the `timescale`, and the difference in `presentation_time` between similar event messages.

7.2.3 MISB Specific Considerations

KLV metadata is a supported data type using the `message_data[]` field. Although the `message_data[]` field is agnostic to the data type stored, commercial/consumer decoder/players do not parse and interpret KLV. This document describes how to add KLV metadata into an `emsg` box within the CMAF framework. However, parsing and rendering this data requires special player decoding logic.

The `presentation_time` value in an `emsg` enables synchronization and binding of the KLV metadata to its associated frame of Motion Imagery. Security information when carried with the metadata ensures Motion Imagery and metadata remain together in delivery to a client receiver.

The following sections define requirements for mapping Motion Imagery, Metadata and Audio to a CMAF file.

8 Source Content Mapping

The MISP mandates that security information be present within any file containing Motion Imagery, metadata, or audio content. The security information represents the highest level of security for the combined content within a file. As security information is KLV metadata this information ultimately maps to an emsg.

| Requirement | |
|-------------|--|
| ST 1910-02 | A CMAF file shall contain security information representing the highest level of security within the file. |

8.1 Motion Imagery Essence

The CMAF specification limits the compression type, profiles, and levels for ABR video essence. Additional formats are likely over time. Table 3 lists the intersection of supported video profiles/levels by CMAF and those approved for use within the MISP. Updates to this document will reflect support of higher compression levels, such as H.265/HEVC Level 6.1 (also approved within the MISP), as ABR technology evolves.

Table 3: Motion Imagery Profiles

| Motion Imagery Profile | Codec | Profile | Level | Color Primaries | Brand | Specification |
|------------------------|-------|---------|-------|-----------------|--------|----------------------------------|
| HD | AVC | High | 4.0 | BT.709 [10] | 'cfhd' | CMAF Media Profile |
| HDHF | AVC | High | 4.2 | BT.709 | 'chdf' | CMAF Media Profile |
| UHD | AVC | High | 5.1 | BT.709 | 'avc1' | Advanced Video Coding extensions |
| HDD8 | HEVC | Main10 | 5.0 | BT.709 | 'cud8' | CMAF Media Profile |
| UHD10 | HEVC | Main10 | 5.1 | BT.709 | 'cud1' | CMAF Media Profile |

| Requirement | |
|-------------|---|
| ST 1910-03 | A CMAF file shall conform to the Motion Imagery profiles listed in MISB ST 1910 Table 3: Motion Imagery Profiles. |

8.1.1 Encoding Guidelines (Informative)

The following recommendations for encoding Motion Imagery for a given source are to facilitate a consistent user experience. These guidelines apply to CMAF files with only one segmented Motion Imagery encoding profile.

- Fixed frame size, fixed aspect ratio, and fixed pixel density
- Constant temporal frame rate

These guidelines apply to CMAF files with multiple segmented Motion Imagery encoding profiles (e.g. Encoding Ladder):

- Constant temporal frame rate

- Key frame (I/IDR) present at least every GOP
- GOP of 2 seconds (equates to 60 frames at 30 frames per second)
- Closed-GOP structures for seamless switching between renditions
- One codec type per CMAF presentation (no codec changes)
- Coding structure (I-B-P) optimized for image quality and bandwidth based on application needs (see MISB RP 0802 [11] for additional guidelines). Current experimentation employs two B-frames and two reference frames per group-of-pictures (GOP).

8.1.2 Encoding Ladder (Informative)

Encoding parameters such as spatial/temporal resolutions and bitrates for the Encoding Ladder may vary based on application and workflow requirements. Table 4 illustrates an exemplar set of bitrates in kilobits per second (kb/s) for encoding H.264/AVC and H.265/HEVC assuming 16:9 aspect ratio imagery and a temporal frame rate of 30 frames per second (fps).

Table 4: Exemplar Encoding Ladder

| Spatial Density (samples) | Frame Rate (fps) | Aspect Ratio | H.264/AVC (kb/s) | H.265/HEVC (kb/s) |
|---------------------------|------------------|--------------|------------------|-------------------|
| 1920 x 1080 | 30 | 16:9 | 5500 | 3800 |
| 1280 x 720 | 30 | 16:9 | 3300 | 2300 |
| 1280 x 720 | 30 | 16:9 | 2000 | 1400 |
| 960 x 540 | 30 | 16:9 | 1200 | 850 |
| 640 x 360 | 30 | 16:9 | 750 | 500 |
| 640 x 360 | 30 | 16:9 | 450 | 300 |

Several commercial industry rules-of-thumb for building an Encoding Ladder are:

- Sufficient maximum and minimum bitrates to meet anticipated network bandwidth fluctuations
- Steps in the ladder given the selected range where the ratio between steps is 1.5 to 2. For example, the ratios of bitrates between the first two levels in Table 4 are:
 - 1080p AVC at 5500 kb/s and 720p at 3300 kb/s = $5500/3300 = 1.67$
 - 1080p HEVC at 3800 kb/s and 720p at 2300 kb/s = $3800/2300 = 1.65$
- Choose a reference a frame size which optimizes image quality for its given bitrate; then from this bitrate determine the remaining ladder encodings.

8.1.3 CMAF Packaging Recommendations (Informative)

A CMAF segment begins with an encoded imagery I/IDR frame to facilitate switching amongst representations. Although segment length may vary across applications, MISB recommends a segment length equal to two seconds, which provides a balance between access time and player requests. This means each GOP of encoded imagery will likewise be two seconds. Thus, at 30 frames per second one segment corresponds to 60 frames of imagery.

MISB recommends a fragment length equal to one second for delivery. Although this introduces additional overhead with an additional IDR frame, the shorter fragment affords finer control in stepping through content. For low latency applications, CMAF permits the use of Chunked Transfer Encoding (CTE), often referred to as ultra-low-latency (ULL) mode. This mode of stream delivery splits traditional segments into smaller addressable fragments called “chunks” ultimately reducing end-to-end streaming latency. A reasonable chunk length is 200 milliseconds; this equates to 6 frames of imagery at 30 fps. The values shown in Table 5 apply to all encoded spatial resolutions and bitrates within the encoding ladder.

Table 5: CMAF Packaging Guidelines: Imagery at 30 fps

| GOP Type | GOP Size | Segment Length | Fragment Length | Chunk Length (opt. Low Latency) |
|----------|-------------------|-------------------|-------------------|---------------------------------|
| Closed | 2 sec (60 frames) | 2 sec (60 frames) | 1 sec (30 frames) | 200 msec (6 frames) |

Design choices in Encoding Ladder, encoding structure, GOP size, etc. depend on the requirements for its application. In lieu of such requirements, the parameters indicated in the above sections should provide a reasonable solution.

8.2 KLV Metadata

As discussed in Section 7.2, the Event Message box (emsg) provides a structure to insert and carry KLV metadata along with its corresponding track of Motion Imagery as one unified file. This standard requires emsg box version 1, which supports a **presentation_time** value to align an emsg in time with the Motion Imagery along a CMAF segment timeline.

8.2.1 KLV Scheme: `scheme_id_uri`

The `scheme_id_uri` = “urn:misb:KLV:bin:2019” identifies the inband event message scheme for binary KLV. Each emsg box carries this information as well as within the MPD issued by a client receiver.

| Requirement | |
|-------------|---|
| ST 1910-04 | The <code>scheme_id_uri</code> for emsg carriage of KLV shall be “urn:misb:KLV:bin:2019”. |

CMAF requires the emsg (or group of emsg boxes) containing the data for an image frame to physically precede the segment containing the referred image frame. So for example, if there were one emsg per image frame, such as illustrated in Figure 10, then an equivalent number of emsg boxes consistent with the number of frames present within that segment need to precede the segment. In this 2-second segment there are 60 frames (i.e., F1, F2, ...F60) of Motion Imagery. Thus, 60 emsg boxes (i.e., e1, e2, ...e60) equating to the 60 image frames precede the segment.

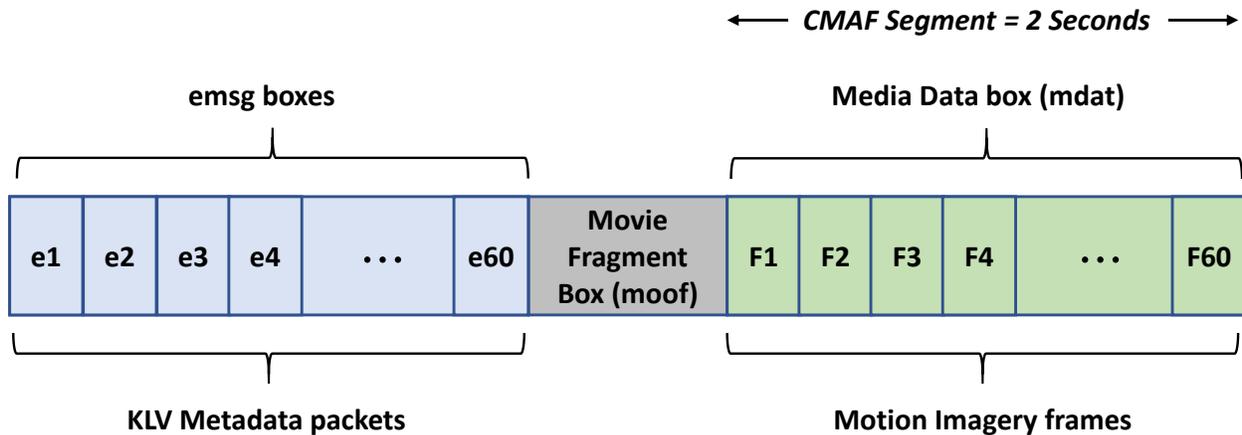


Figure 10: Example: KLV metadata as emsg and Motion Imagery in mdat box

Note: there is no requirement that there be one emsg for each image frame. In fact, an emsg can be present sporadically (i.e., not for every frame), or even a multiplicity of times for the same image frame.

8.2.2 KLV Track: emsg id field

A KLV track is a logical ensemble of KLV messages within a CMAF file. Adding an identifier to a KLV message carried in an emsg enables metadata to retain its relationship to other metadata within its own group; for example, packaging source content with five different metadata streams into five different emsg KLV tracks. The emsg id field provides this identifier for metadata affording the notion of a metadata track. The packager assigns a unique emsg id called a “stream index” to metadata belonging to the same metadata stream.

In this standard the stream index is a value from 0 to 255 inclusive, which maps into the least significant byte of the emsg id 32-bit field. As the purpose of an emsg id is unique identification of an emsg within a stream, in addition to the stream index the upper three bytes (i.e., 24 bits) of the emsg id is a count of the emsg belonging to the specific KLV track. The count wraps around to 0 after its maximum value.

Figure 11 shows the partitioning of bytes for the count and KLV track index. The KLV track index starts at zero for the first KLV track of a segment and increments by 1 for every track.

Note: If the number of KLV tracks changes during a live streaming session, the manifest should signal this change by introducing a new period.

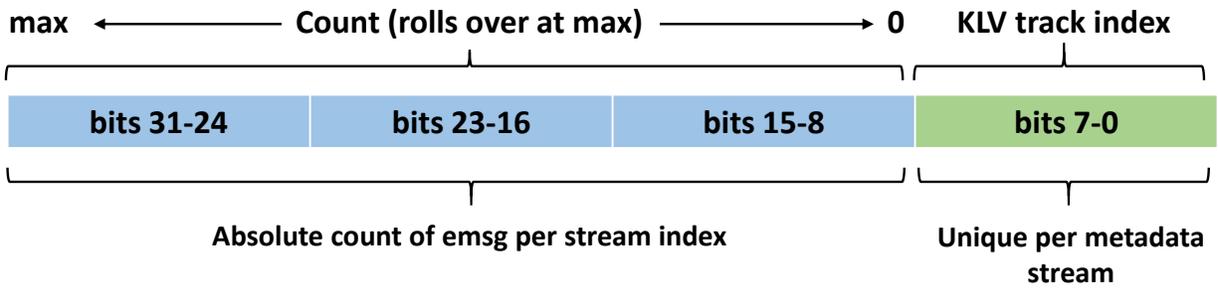


Figure 11: emsg id field structure

The count begins right justified in the least significant byte (bits 15-8) of the upper three bytes. Table 6 lists the allowed values for the KLV track Index.

Table 6: Valid KLV track index values

| KLV Track Index Values | |
|------------------------|----------|
| 0x00 – 0x0F | Reserved |
| 0x10 – 0xFF | Allowed |

| Requirement(s) | |
|----------------|--|
| ST 1910-05 | The emsg id shall be composed of the KLV track index in the least significant byte, completed with an absolute counter starting at zero and incrementing by 1 for every subsequent emsg box for that KLV track index in the upper 3 bytes. |
| ST 1910-06 | The KLV track index shall be between 0x10-0xFF. |

8.2.3 emsg: presentation_time field

The **presentation_time** field of the emsg enables structural alignment in time of the emsg information (i.e., KLV metadata) to its corresponding image frame. The **presentation_time** by itself does not imply the metadata is synchronous with the image frame; merely that the two data types are coincident in the incoming data stream. The **emsg value** field (discussed below) provides additional signaling for the presence of synchronous versus asynchronous metadata.

The **presentation_time** is a value that lies within the frame period in which metadata is coincident. That is, the occurrence in time of a frame and that of any metadata received during that time assumes the **presentation_time** for the metadata lies between the beginning and ending of that coincident image frame.

| Requirement | |
|-------------|--|
| ST 1910-07 | The emsg presentation_time shall be within the duration of an image frame. |

8.2.4 msg: event_duration field

The `event_duration` specifies the duration (i.e., pertaining to the length) of the metadata in time. As the length of metadata received in a stream may be unknown until reception of all metadata, the length defaults to the duration of one image frame.

| Requirement | |
|-------------|---|
| ST 1910-08 | The default value for the <code>msg event_duration</code> shall be the duration of one image frame. |

8.2.5 KLV Type: msg: value field

Source metadata is either time-aligned with a frame of Motion Imagery (i.e., synchronous metadata) or in proximity to a frame (i.e., asynchronous metadata). Regardless of the metadata synchronization type the metadata maps into an `msg` box and receives a `presentation_time` (reference Figure 8). The `msg value` field signals the data as synchronous or asynchronous with respect to the imagery. This signal informs a decoder on how to interpret the `msg` data and its respective timing to the Motion Imagery.

Each metadata type (i.e., synchronous, asynchronous) maps to its own respective set of `msg` boxes. Thus, an `msg` box has a stream type “personality” containing only that type of data. As illustrated in the example of Figure 12, content with one synchronous metadata and one asynchronous metadata source requires two sets of `msg` boxes, one for each type of metadata.

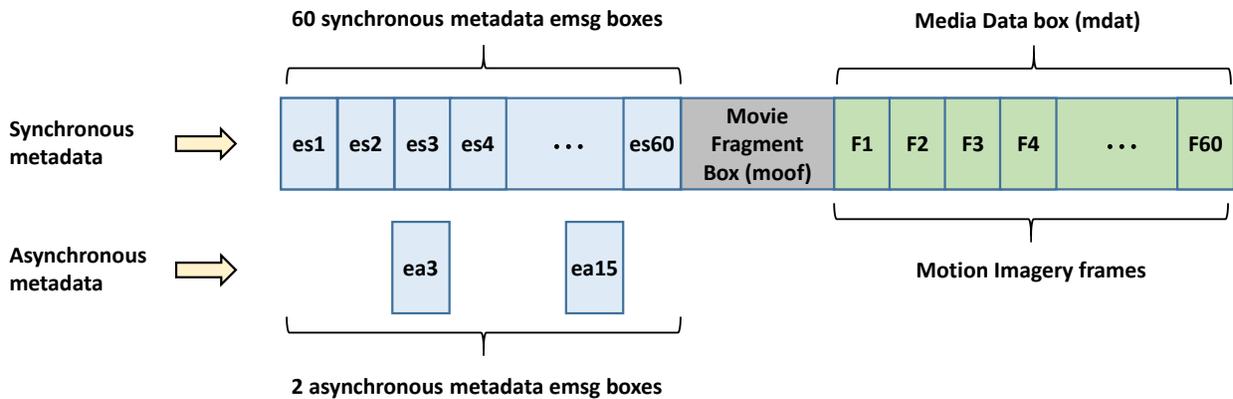


Figure 12: Example: synchronous (‘es’) and asynchronous (‘ea’) metadata mapped into respective msg boxes over a CMAF segment

The example shows one `msg` box per image frame (i.e., `es1` contains metadata for `F1`, `es2` contains metadata for `F2`, etc.) for the synchronous metadata, and only two `msg` boxes (i.e., `ea3` and `ea15`) for the asynchronous metadata over the 60 frame image sequence. The incidence of `ea3`, `ea15` falls along the media timeline in proximity to a frame of imagery it may relate.

There are different “goodness” levels of synchronization for the Motion Imagery with respect to its metadata depending on how a source constructs its content. When absolute time information

is present in both the Motion Imagery and metadata remediation in a post-process can improve the level. See Appendix B for information on remediation of MPEG-2 TS content.

Where a MISP timestamp is present in the SEI message of the compressed Motion Imagery and a MISP timestamp is present in the metadata, extremely accurate alignment of an image frame to its metadata is possible. This is true for both asynchronous and synchronous metadata. Because one prevalent use case is mapping MPEG-2 TS content to CMAF, this standard defines specific **emsg values** to 1) indicate the type of existing synchronization within a source TS, 2) allow for a remediated TS (discussed further in Appendix B), and 3) extend to other content sources. Table 7 lists the allowed values for the **emsg value** field. Section 9 provides a more in-depth discussion of these values.

Table 7: emsg value signaling per type of metadata synchronization

| MISP timestamp in Motion Imagery SEI | MISP timestamp in KLV metadata | Remediated: Aligned by MISP timestamp's | Level of Synchronization | emsg value (utf8) | Notes |
|---|--------------------------------|---|--------------------------|-------------------|--------|
| yes | yes | yes | Level 1 | 11FC | |
| yes | yes | yes | Level 2 | 12FC | |
| yes | yes | no | Level 2 | 01FC | Note 1 |
| no | yes | no effect | Level 3 | 01BD | Note 2 |
| Note 1: Legacy (non-remediated) MPEG-2 TS synchronous metadata | | | | | |
| Note 2: Legacy (non-remediated) MPEG-2 TS asynchronous metadata | | | | | |

8.2.6 KLV Data: emsg: message_data[] field

The **message_data[]** is the data information (i.e., KLV metadata). The KLV is a binary set of data as defined by various MISB standards, such as MISB ST 0601. The data inserts directly into the **message_data[]** field as a series of bytes.

Appendix A provides a sample of KLV metadata mapped into an event message emsg box.

8.3 Audio Essence

If encoding audio, the CMAF specification allows a variety of compression types for ABR audio. Table 8 lists the intersection of supported profiles by CMAF and those approved by the MISP in MISB ST 1001.

Table 8: Audio Media Profiles

| Audio Profile | Codec | Profile | Level | CMAF Brand | Normative Reference |
|---------------|-------|---------|-------|------------|---------------------|
| AAC | AAC | AAC-LC | 2 | 'aac' | Table A.2 [1] |

| Requirement | |
|-------------|---|
| ST 1910-09 | A CMAF file shall conform to the Audio profiles listed in MISB ST 1910 Table 8: Audio Media Profiles. |

9 Source-specific Packaging

Source content as identified within the MISP may come from Class 0 Motion Imagery, Class 1 Motion Imagery, Class 2 Motion Imagery or Class 3 Motion Imagery acquisition systems. CMAF packaging applies to any type of content that meets both the CMAF specification and the approved methods for compression within the MISP as indicated in Table 3 and Table 8 above in this document.

In this standard the packaging of content into CMAF and the signaling for metadata is the same regardless of the source content. This allows for extension of the technology into serving additional applications as they evolve. This document will update mappings for future sources in support of new applications as warranted. As the current demand is for mapping MPEG-2 Transport Stream content to CMAF the following section pertains to this specific application.

9.1 MPEG-2 Transport Stream Content

Motion Imagery encapsulated within MPEG-2 TS is either H.264/AVC or H.265/HEVC compressed video. MISB ST 1402 [12] provides guidelines for carriage of Motion Imagery, Audio and Metadata in MPEG-2 TS. MISB ST 0604 [13] specifies the format and where to insert a MISP timestamp into a Motion Imagery compressed stream. These two standards guide the requirements for mapping content from MPEG-2 TS to CMAF.

The top graphic in Figure 13 illustrates a conceptual example of an MPEG-2 TS with four essence streams: compressed Motion Imagery, synchronous KLV metadata, asynchronous KLV metadata, and audio. The compressed Motion Imagery (IPP frames), audio (A), synchronous metadata (M_S), and asynchronous metadata (M_A) data when multiplexed together might appear as shown in the center graphic. The sequence of multiplexed data continues completing one GOP.

The next GOP aligns with the next segment and begins at the next I-frame. One or more emsg boxes (i.e., first emsg box shown as e1) followed by a moof box precedes the imagery I-frame data in an mdat box.

Assuming no transcoding, the bottom graphic in Figure 13 shows the various essences within the TS mapped into a CMAF file assuming a 2-second segment. Metadata fills the emsg boxes according to the imagery frame it pertains; thus, in this example, the first M_S metadata applies to the first I-frame. The second M_S applies to the first P-frame, while the M_A metadata applies to the second P-frame. Although this example shows one set of metadata per image frame this is not necessary. Frames may have more than one emsg assigned, for instance, if both synchronous and asynchronous metadata are coincident to the same frame, there are multiple metadata streams, or a frame may have no metadata, and thus, no emsg assigned.

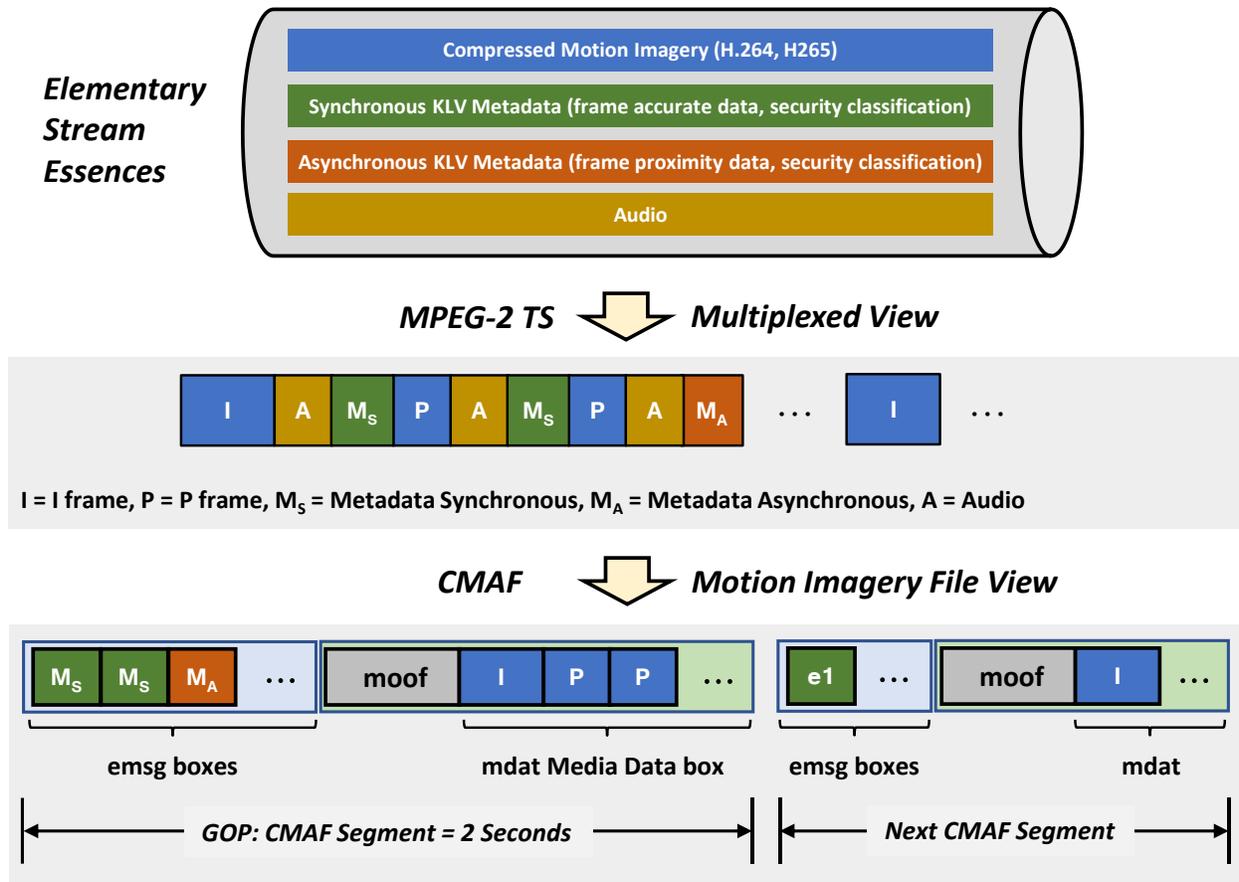


Figure 13: Example MPEG-2 Transport Stream essences mapped to CMAF file

9.1.1 Motion Imagery SEI Message: MISP Timestamp

Within the Supplemental Enhancement Information (SEI) message of the compressed Motion Imagery stream a MISP timestamp may be present (Note: current MISP guidance requires the MISP timestamp; however, some legacy systems do not supply one). On packaging the Motion Imagery into CMAF this information is to remain intact in any subsequent transcoding of the Motion Imagery.

| Requirement(s) | |
|----------------|---|
| ST 1910-10 | Where H.265/HEVC compressed Motion Imagery is transcoded, the information in the SEI Message user_data_unregistered field shall be preserved. |
| ST 1910-11 | Where H.264/AVC compressed Motion Imagery is transcoded, the information in the SEI Message user_data_unregistered field shall be preserved. |

9.1.2 Packaging KLV Metadata

MISB ST 1402 allows a source Motion Imagery MPEG-2 transport stream to contain one synchronous metadata elementary stream and one or more asynchronous metadata elementary streams. ISO/IEC 13818-1 [14] defines a `stream_id = 0xFC` for synchronous streams, and a `stream_id = 0xBD` for asynchronous streams. Table 9 summarizes this guidance.

Table 9: Number of stream types allowed (per MISB ST 1402)

| Multiplex Method | stream_id (ISO/IEC 13181-1) | Allowed in Transport Stream (MISB ST 1402) |
|------------------|-----------------------------|--|
| Synchronous | 0xFC | 0 or 1 (zero or one may be present) |
| Asynchronous | 0xBD | 0 or * (zero or more may be present) |

With only one synchronous stream type allowed this stream would typically map into an emsg with a KLV stream index (i.e., within the emsg id field) with a one fixed value throughout the file. In the asynchronous stream type case, there are more options: 1) combine (i.e., flatten) all asynchronous streams into one stream with one assigned KLV stream index, or 2) maintain each asynchronous stream with its own identity by assigning a unique KLV stream index (i.e., within the emsg id field) for each asynchronous stream. These are implementation choices which may have benefits in some applications.

9.1.3 Metadata Carriage Type Signaling

The time synchronization in a transport stream between a Motion Imagery frame and its corresponding metadata is governed in two ways: 1) in the Synchronous Metadata Multiplex Method (MISB ST 1402) metadata is prefixed with a Presentation Time Stamp (PTS) for multiplexing in similar fashion to the Motion Imagery; and 2) in the Asynchronous Metadata Multiplex Method metadata is multiplexed in *proximity* to the imagery, which could be displaced in time by several image frames. Synchronous carriage thus improves the alignment of the Motion Imagery with metadata.

ISO/IEC 13818-1 defines the **stream_id** in the PES (packetized elementary stream) header to indicate either synchronous data carriage (i.e., **stream_id** = 0xFC), or asynchronous data carriage (i.e., **stream_id** = 0xBD). Also specified are descriptors within the Program Map Table (PMT) for auxiliary information regarding an elementary stream. MISB ST 1402 defines values (see ST 1402 Appendix B) for these metadata descriptors. Within the **metadata_descriptor** field of a synchronous stream, the **metadata_application_format** default value prescribed by the MISB is 0x0100 (see ST 1402). In asynchronous carriage the **registration_descriptor** has no such corresponding field.

This standard supports two types of transport stream constructions: legacy and remediated. The **stream_id** together with the **metadata_application_format** provide the necessary information to map synchronous metadata to CMAF for either construction. Mapping asynchronous metadata to CMAF requires only the **stream_id**.

9.1.3.1 CMAF Packaging of a Legacy Transport Stream

Synchronous Metadata:

For CMAF packaging, synchronous metadata with **metadata_application_format** values in the range 0x0100-0x0103 and **stream_id** = 0xFC map to the **value** field of an emsg as utf8 string “01FC”. “FC” is readily identifiable as the TS synchronous stream type.

| Requirement | |
|-------------|--|
| ST 1910-12 | Where MPEG-2 Transport Stream synchronous metadata with a stream_id = 0xFC and metadata_application_format in the range 0x0100-0x0103 is mapped to a CMAF emsg box, the emsg value field shall be set to the utf8 string “01FC”. |

Asynchronous Metadata:

Unlike the synchronous stream an asynchronous elementary stream does not have a metadata_application_format descriptor. Signaling asynchronous metadata for CMAF packaging is through the TS stream_id value (which is 0xBD) alone. For CMAF packaging, asynchronous metadata with stream_id = 0xBD map to the value field of an emsg as utf8 string “01BD”. “BD” is readily identifiable as the TS asynchronous stream type.

| Requirement | |
|-------------|--|
| ST 1910-13 | Where MPEG-2 Transport Stream asynchronous metadata with a stream_id = 0xBD is mapped to a CMAF emsg box, the emsg value field shall be set to the utf8 string “01BD”. |

Table 10 restates the guidance for mapping a legacy MPEG-2 transport stream to a CMAF file.

Table 10: Signaling for legacy MPEG-2 TS CMAF packaging

| Type of KLV Synchronization | metadata_application_format | stream_id | emsg value (utf8) |
|-----------------------------|-----------------------------|-----------|-------------------|
| synchronous | 0x0100-0x0103 (uint) | 0xFC | 01FC |
| asynchronous | N/A | 0xBD | 01BD |

9.1.3.2 CMAF Packaging of a Remediated Transport Stream

Remediation is a process which can improve the synchronization of metadata to Motion Imagery. Remediation can occur either upstream prior to CMAF packaging or during the CMAF packaging process itself. Identifiers for remediated data signal the mapping within a CMAF file.

Synchronous Metadata:

As described in Section 9.1.3.1, the metadata_application_format field within the metadata descriptor of the PMT for a MPEG-2 TS synchronous elementary stream supports legacy stream metadata encoding. In a remediated stream, the metadata_application_format likewise supports remediated stream metadata; however, there are new indicators defined which signals the “quality” or “goodness” of metadata synchronization.

Table 11 shows three cases of timing for metadata to Motion Imagery. The “Level of Synchronization” has two cases (i.e., Level 1 and Level 2) for synchronous data and one (i.e., Level 3) for asynchronous data. At any one time only one level of synchronous metadata is present – either Level 1 or Level 2. The Level of Synchronization is a qualifier on the “goodness” of the timing for the metadata, which informs the decoder of the timing quality of the Motion Imagery with respect to its metadata.

Table 11: Signaling for remediated MPEG-2 TS CMAF packaging

| Type of KLV Synchronization | Level of Synchronization* | metadata_application_format | stream_id | emsg value (utf8) |
|-----------------------------|---------------------------|-----------------------------|-----------|-------------------|
| synchronous | Level 1 | 0x11FC (uint) | 0xFC | 11FC |
| synchronous | Level 2 | 0x12FC (uint) | 0xFC | 12FC |
| asynchronous | Level 3 | N/A | 0xBD | 01BD |

* defined in Appendix B Section 12.1.3

| Requirement(s) | |
|----------------|--|
| ST 1910-14 | Where MPEG-2 Transport Stream synchronous metadata with a stream_id = 0xFC and metadata_application_format = 0x11FC is mapped to a CMAF emsg box, the emsg value field shall be set to the utf8 string "11FC". |
| ST 1910-15 | Where MPEG-2 Transport Stream synchronous metadata with a stream_id = 0xFC and metadata_application_format = 0x12FC is mapped to a CMAF emsg box, the emsg value field shall be set to the utf8 string "12FC". |

Asynchronous Metadata:

The identifier of remediated MPEG-2 TS asynchronous metadata is the same as a legacy stream. Appendix B provides additional information on remediation and what these new identifiers mean.

9.1.4 Decoder Backward Compatibility

CMAF decoders need to support both legacy and remediated stream signaling. Examining the first two characters of an emsg value indicates whether the metadata is of synchronous or asynchronous origin. In both legacy and remediated cases, a utf8 value of "FC" for the first two characters indicates synchronous metadata regardless, while a utf8 value of "BD" for the second two characters indicates asynchronous metadata. Thus, a non-remediated-aware decoder will process both legacy and remediated streams as "seeing" just synchronous and asynchronous. There is no distinction made on the synchronous timing quality in this case; the decoder receives all synchronous metadata in the same way as a legacy stream.

A remediated-capable decoder can examine the leading two characters of the emsg value; remediated synchronous data has the leading two characters equal to the utf8 string "11" or "12"; remediated asynchronous data has the leading two characters equal to the utf string "01". All legacy data has the leading two characters equal to "01".

10 Player Functionality (Informative)

10.1 *Trick Play*

The term "trick play" refers to playback other than forward playback at the recorded speed of the video/audio content ("1x"). Examples include fast forward, slow motion, reverse, single step, and random access.

These modes of playback fall to the responsibility of the player design. However, impacting the degree of control is the choice in CMAF segment length and download speed limitations. Encoded content I/IDR frame periodicity determines random access points. Player buffer size is a factor in limiting rewinding to earlier content. Decoding slower than normal and repeating frames can simulate slow motion.

Other options to implement trick play include converting I/IDR frames to thumbnail images which require far less storage at the player. Playing through the thumbnail images creates a "film strip" of the content for determining when to begin playback at a specific point. Another option is to request I/IDR frame only segments at a lower bitrate to speed delivery. Higher bitrates then resume once normal playback continues.

10.2 *Video / Metadata Synchronization*

Although synchronization of Motion Imagery and metadata within a CMAF file is predictable and reliable, web browsers such as Chrome, Firefox, and Edge exhibit different behavior in rendering video with other timed media like metadata. As the use cases for these technologies are typically video with audio (and subtitles with loose timing with respect to a video frame), synchronizing metadata to be frame accurate continues to be works in progress. Thus, applications should be cognizant of these differences. The MISB continues to evaluate delivery performance.

11 Appendix A – Sample KLV Mapped to Event Message

Figure 14 shows a sample of KLV mapped into an emsg version 1 box. The Motion Imagery framerate is 25 Hz. A timescale of 25000 with a duration of 1000 provides the framerate timing (i.e., 25000/1000 = 25). Since this is frame 1 the `presentation_time` is 0 with an `id` = 0. Note that the next event message follows the first. In the second event message the `presentation_time` = 0x03e8 (i.e., 1000) indicating a time of 1/25 of 25000 and an `id` = 1.

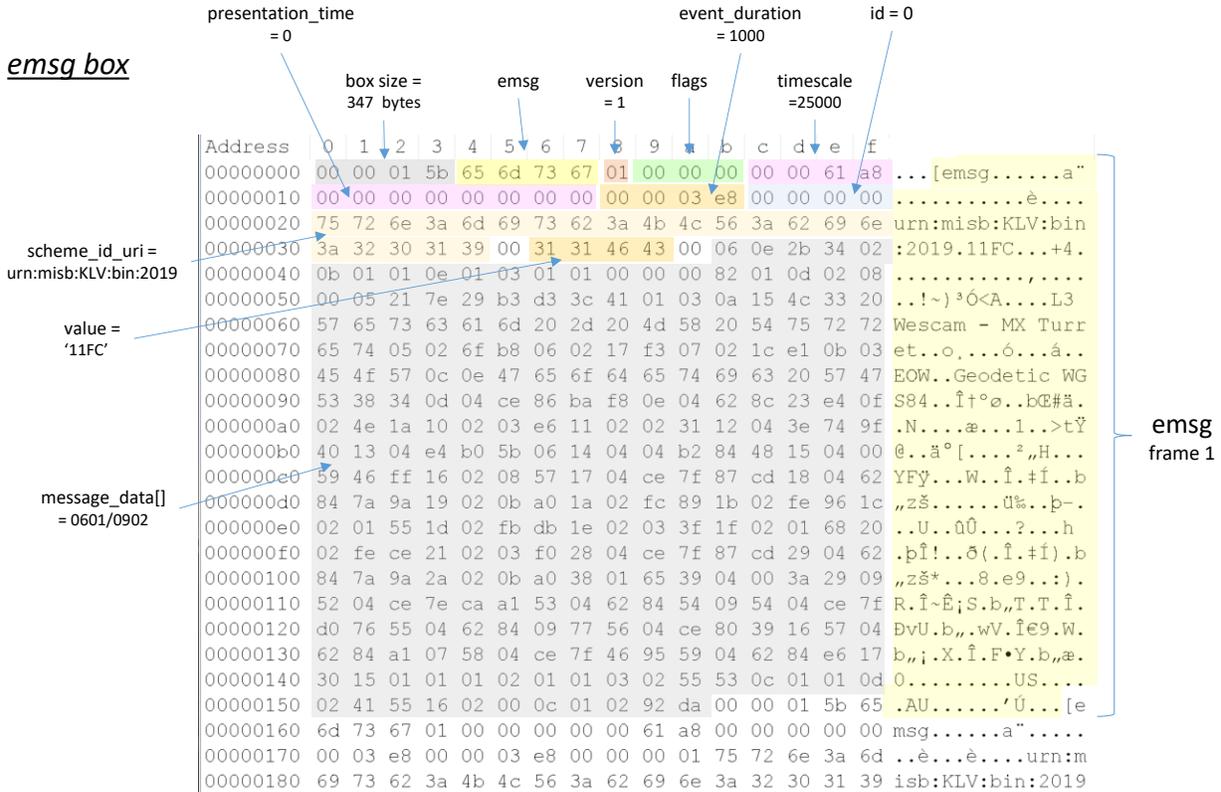


Figure 14: Sample of KLV metadata mapped to an event message (emsg box)

12 Appendix B – MPEG-2 TS Motion Imagery/metadata Synchronization – Informative

Class 1/Class 2 Motion Imagery content encapsulated within MPEG-2 Transport Stream typically carries additional content such as metadata and audio. The synchronization of metadata to a Motion Imagery frame at the source depends on how accurately the implementation inserts metadata with respect to the imagery. In an asynchronous metadata stream metadata may lose its timing with respect to the imagery because there is no facility to time the multiplexing of metadata into the transport stream – the only inherent timing is locality or proximity of the metadata to its corresponding related frame. In synchronous metadata streams the “goodness” or accuracy of the metadata to its respective image frame relies on the implementation supplying the metadata to the transport stream multiplexer at the correct time to its associated imagery.

In both cases improved accuracy of the synchronization is possible. This is the function of the post-collection remediation process described here. Note: the MISB is developing guidance for remediation so this is an informative overview only.

Figure 15 illustrates where in the workflow the remediation process occurs. An asset (e.g., airborne platform) delivers source content in a MPEG-2 Transport Stream (TS) with compressed Motion Imagery and metadata and audio to a receiver (e.g., ground station). The remediation process accepts this TS and outputs a remediated TS, which is an improved input to further downstream processes, such as Adaptive Bitrate (ABR) delivery or exploitation tools.

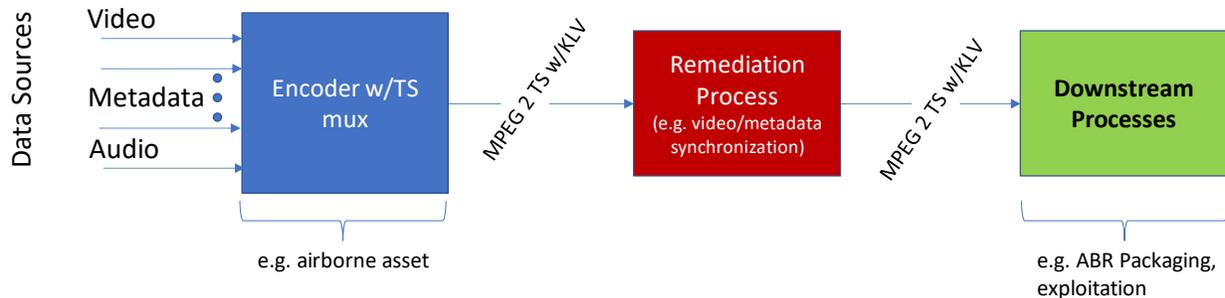


Figure 15: Remediation process of MPEG-2 TS content

12.1 Functions of the Remediation Process

Improving the time synchronization between Motion Imagery and metadata depends on whether a MISP timestamp is present in the Motion Imagery. MISB ST 0604 describes the format and the location for a MISP timestamp within the Supplemental Enhancement Information (SEI) message in compressed Motion Imagery when supplied. The first step in the remediation process is determining the presence of this timestamp; if present, accurate synchronization is possible. Leveraging the MISP timestamp in the Motion Imagery along with the MISP timestamp in the metadata enables accurate synchronization between the two.

In the event a timestamp is not present in either the Motion Imagery or the metadata, the MPEG-2 TS Presentation Time Stamp (PTS) in the header for synchronous metadata defaults as the most optimal timing available. For asynchronous metadata, the relative proximity of the metadata to its image frame serves the synchronization function; this method produces the least accurate synchronization.

Levels of Synchronization – Timing “goodness”

The time synchronization between Motion Imagery and metadata depends on the method of metadata carriage selected and whether MISP timestamps are present in the Motion Imagery and the metadata. Three levels of synchronization assigned based on these conditions indicate to a client receiver the “goodness” of the synchronization for exploitation purposes. In legacy streams these assigned levels are absent; in these systems a user has little knowledge of the accuracy in the synchronization. A remediated stream introduces these levels as signaling carried forward for end-user awareness. The following sections describe the conditions which result in an assigned level of synchronization.

12.1.1 Motion Imagery with a MISP Timestamp

MISB mandates metadata incorporate a MISP timestamp as defined in MISB ST 0603 [15]. The MISP likewise mandates a MISP timestamp in the Motion Imagery. Because both timestamps derive from the same absolute time reference (see ST 0603), they provide a very accurate means for registering metadata to a frame of Motion Imagery when present. Unfortunately, systems built which precede these requirements do not include the timestamp in the Motion Imagery.

Case 1: Synchronous Metadata with MISP Timestamp

Synchronous metadata within a MPEG-2 TS is “synchronous” because the header for the data contains a Presentation Time Stamp, or PTS, assigned by the transport stream multiplexer much like that done for video and audio. Thus, synchronization of metadata to the imagery occurs at the input to the TS multiplexer. Assuming the implementation provides the data to the multiplexer consistent with the availability of its referenced image this is an accurate synchronization method. However, receivers of this data do not have enough information regarding the implementation to know the accuracy of the multiplexed timing.

If the MISP timestamp is present in the Motion Imagery and the metadata, correct synchronization of the two can be both guaranteed and known. With this information adjustments made to the PTS of the metadata with respect to the Motion Imagery form new inputs for re-multiplexing of the two.

Case 2: Asynchronous Metadata with MISP Timestamp

Asynchronous metadata carries no PTS information; multiplexing occurs in proximity to Motion Imagery when presented to the multiplexer. This produces uncertainty in the time synchronization between the imagery and the metadata. However, when the metadata and the Motion Imagery both contain a MISP timestamp, retiming is possible as in Case 1. Thus, asynchronous metadata post remediation becomes accurately synchronized metadata.

In both Case 1 and 2 the MISP timestamp guides increased accuracy in Motion Imagery / metadata synchronization. This is the optimal situation and one in which systems should adhere. **This is Level 1 synchronization.**

Case 3: Synchronous Metadata without MISP Timestamp

Assuming the metadata aligns coincident with its corresponding image frame at the multiplexer, this produces reasonably accurate synchronization. The PTS for both the Motion Imagery and the metadata provide the best timing information available. Unfortunately, without information on the implementation which constructed the stream timing, it is not possible to know the degree of accuracy. **This is Level 2 synchronization.**

Case 4: Asynchronous Metadata without MISP Timestamp

Where a MISP timestamp is not present in the metadata, remediation of the timing is not possible. **This is Level 3 synchronization.**

12.1.2 Motion Imagery without a MISP timestamp

Not all source content contains a MISP timestamp in the Motion Imagery. In these cases, remediation cannot improve the accuracy in synchronization of the two. Although remediation of

the timing is not possible, the inherent timing provided by the MPEG-2 TS PTS can indicate that the Motion Imagery and metadata are in close, if not complete, alignment. In the following two cases, presence of a MISP timestamp in the metadata but not the Motion Imagery does not impact remediation, and therefore, this produces only these two cases.

Case 5: Synchronous Metadata

Assuming the metadata aligns coincident with its corresponding image frame at the multiplexer, this produces reasonably accurate synchronization. However, note this is an assumption. The PTS for both the Motion Imagery and the metadata provide the best timing information available. Unfortunately, without information on the implementation which constructed the stream timing, it is not possible to know the degree of accuracy. For this reason, the grading of the quality of synchronization is like Case 3 and is a **Level 2 synchronization**.

Case 6: Asynchronous Metadata

Without a MISP timestamp in the Motion Imagery and without the PTS synchronizing mechanism of the transport stream this situation provides the lowest level of synchronization timing – that is, a **Level 3 synchronization**.

12.1.3 Levels of timing synchronization “goodness”

Given the cases described, the rating of the “goodness” or accuracy in the synchronization of metadata to Motion Imagery results in three levels: Level 1, Level 2, Level 3 listed in Table 12.

Table 12: Levels of KLV Metadata Synchronization

| Case | Stream Metadata Type | MISP Timestamp | | Level of Synchronization |
|------|----------------------|----------------|----------|--------------------------|
| | | Motion Imagery | Metadata | |
| 1 | synchronous | Yes | Yes | Level 1 (best) |
| 2 | asynchronous | Yes | Yes | Level 1 |
| 3 | synchronous | Yes | No | Level 2 |
| 4 | asynchronous | Yes | No | Level 3 |
| 5 | synchronous | No | X | Level 2 |
| 6 | asynchronous | No | X | Level 3 |

12.1.4 Conversion to synchronous elementary stream

In a remediated stream the Level of Synchronization in Table 12 is coded with the value given in Table 11 into the `metadata_application_format` of the metadata descriptor in the Program Map Table (PMT) of the MPEG-2 Transport Stream. This signal provides an end user with information to improve their understanding of the Motion Imagery-to-metadata synchronization in the exploitation process. The levels correspond to identifiers for CMAF packaging according to Section 0.