INTERNATIONAL ORGANISATION FOR STANDARDISATION

ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/IEC JTC1/SC29/WG11

CODING OF MOVING PICTURES AND AUDIO

ISO/IEC JTC1/SC29/WG11 N16186

June 24 2016

Source	Editors
Status	CD TEXT
Title	Common Media Application Format for Segmented Media
Author	Kilroy Hughes, David Singer

Common Media Application Format for Segmented Media

Media Application Format optimized for large scale delivery of a single encrypted, adaptable multimedia presentation to a wide range of devices; compatible with a variety of adaptive streaming, broadcast, download, and storage delivery methods

Table of Contents

Figures

Introduction

Several MPEG technologies have been adopted for the majority of video delivered over the Internet and other IP networks (cellular, cable, broadcast, etc.). Various organizations have taken MPEG’s core coding, file format and system standards, and combined them into their own specifications for their specific applications. While these specifications share major common parts, their differences result in both unnecessary duplication of engineering effort, and duplication of identical content in slightly different formats that increases storage and delivery costs. The industry would benefit if application consortia could reference a single MPEG specification (a “common media format”) that would allow a single media encoding to use across many applications and devices.

Common Media Application Format for Presentation of Segmented Media

1 Scope

The scope of CMAF is the encoding and packaging of segmented media objects for delivery and decoding on end user devices in adaptive multimedia presentations. Three Resource packages are defined for storage, identification, and delivery of encoded media objects. Delivery and presentation are abstracted by a hypothetical application model that allows a wide range of implementations without specifying any.

CMAF constrains media encoding and packaging to allow adaptive delivery of alternative tracks of segmented media to different devices, over different networks. CMAF defines a CMAF Presentation Profile, ISO Base Media File format file constraints and brand, CMAF Media Profiles that specify track formats, media encoding/decoding, and brands, and constraints on sets of CMAF Tracks that can be adaptively streamed. This enables most Internet devices today to play a CMAF Presentation conforming to the specified CMAF Presentation Profile.

A manifest and player are assumed in the application model. The manifest describes a CMAF presentation and media resources, and the player selects, decodes, synchronizes, and presents CMAF media resources described by the manifest in a continuous multimedia presentation consistent with its encoding.

CMAF does not specify a manifest, player, or delivery protocol, with the intent that any that meet the functional requirements can be used.

See Section 6 for more detail.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

These normative references are intended to include corrigenda and amendments available at the time of use.

[AAC] ISO/IEC 14496-3, Information technology — Coding of audio-visual objects — Part 3: Advanced Audio Coding, including Amendment 4

[ABNF] RFC 5234 — Augmented BNF for Syntax Specifications: ABNF https://tools.ietf.org/html/rfc5234

[AVC] ISO/IEC 14496-10, Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding

[CENC] ISO/IEC 23001-7: 2016, Third Edition, “Information technology — MPEG systems technologies — Part 7: Common encryption in ISO base media file format files”

[DASH] ISO/IEC 23009-1, Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats

[HEVC] ISO/IEC 23008-2, Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 2: High efficiency video coding

[ISOBMFF] ISO/IEC 14496-12:2015 “Information technology — Coding of audio-visual objects — Part 12: ISO Base Media File Format

[ISOTXT] ISO/IEC 14496-30, "Timed Text and Other Visual Overlays in ISO Base Media File Format"

[ISOVIDEO] ISO/IEC 14496-15, Third Edition, “Information technology -- Coding of audio-visual objects -- Carriage of NAL unit structured video in the ISO Base Media File Format”

[R1886] ITU-R Recommendation BT.1886, Reference electro-optical transfer function for flat panel displays used in HDTV studio production, March 2011.

[R2035] ITU-R Recommendation BT.2035, A reference viewing environment for evaluation of HDTV program material or completed programmes, July 2013.

[BT709] ITU-R Recommendation BT.709, Parameter values for the HDTV standards for production and international programme exchange, June 2015.

[X667] ITU-T Recommendation X.667, Information technology — Open Systems Interconnection — Procedures for the operation of OSI Registration Authorities: Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier components. https://www.itu.int/rec/T-REC-X.667

[RFC6381] IETF RFC 6381, The 'Codecs' and 'Profiles' Parameters for "Bucket" Media Types, August 2011.

[RFC7230] IETF RFC 7230, Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing

[RFC7231] IETF RFC 7231, Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content

[SCTE128] ANSI/SCTE 128–1 2013: AVC Video Constraints for Cable Television, Part 1 — Coding, http://www.scte.org/documents/pdf/Standards/ANSI_SCTE%20128–1%202013.pdf

[BT2020] ITU–R Recommendation BT.2020, Parameter values for ultra–high definition television systems for production and international programme exchange, October 2015, http://www.itu.int/rec/R–REC–BT.2020/en

[ST2084] SMPTE ST 2084:2014, Dynamic Range Electro–Optical Transfer Function of Mastering Reference Displays, August 16, 2014.

[ST2086] SMPTE ST 2086:2014, Mastering Display Color Volume Metadata Supporting High Luminance and Wide Color Gamut Images, October 13, 2014.

[CEA608] CTA–608–E R–2014, Line 21 Data Services, April 1, 2008, http://www.ce.org/Standards/Standard–Listings/R4–3–Television–Data–Systems–Subcommittee/Line–21–Data–Service.aspx

[CEA708] CTA–708–E, Digital Television (DTV) Closed Captioning, August 23, 2013, http://www.ce.org/Standards/Standard–Listings/R4–3–Television–Data–Systems–Subcommittee/CEA–708–D.aspx

[MP4SYS] ISO/IEC 14496–1:2010, Information technology –– Coding of audio–visual objects –– Part 1: Systems,

[MP4FILE] ISO/IEC 14496–14, Information technology –– Coding of audio–visual objects –– Part 14, The MP4 File Format

[IMSC] W3C, TTML Profiles for Internet Media Subtitles and Captions 1.0 (IMSC1), http://www.w3.org/TR/ttml–imsc1/

[TTMLREG] W3C, TTML Media Type Definition and Profile Registry, W3C Working Group Note, 10 May 2016, https://www.w3.org/TR/ttml-profile-registry

[EBUTTD] TECH 3380, EBU-TT-D Subtitling Distribution Format, https://tech.ebu.ch/docs/tech/tech3380.pdf

[VTT] W3C, WebVTT: The Web Video Text Tracks Format, Draft Community Group Report, 8 December 2015, http://www.w3.org/TR/webvtt1/

3 Terms, definitions, and abbreviated terms

3.1 Terms and definitions

CMAF File an ISOBMFF file conforming to the `‘cmfc’` brand specified in Section 7, and the Sections it references. (Section 7)

Adaptive Switching Process process defined by a Media Profile for seamlessly switching between CMAF Tracks in a Switching Set of that Media Profile. (Section 6.4.6)

CMAF Header sequence of ISO Base Media File Format boxes starting with a file type box and including a movie box that includes initialization information for a CMAF Track. (Section 7.3.3).

CMAF Presentation set of one or more synchronized CMAF Selection Sets, each containing CMAF Tracks of one media type that can be sequentially decoded to produce a multimedia user experience, potentially including synchronized audio, video, and subtitles. (Section 6.4.8)

CMAF Presentation Timeline
the timeline shared by all CMAF Selection Sets in a CMAF Presentation, starting at time zero coincident with the start of the earliest intended for presentation, continuing through the duration of the last samples intended for presentation. (Section 6.4.8)

CMAF Fragment
ISO Base Media File Format segment, as defined by ISO/IEC 14496-12 section 8.16, conforming to CMAF encoding and packaging constraints. (Section 6.4.1)

CMAF Segment
resource consisting of one or more consecutive CMAF Fragments from the same CMAF Track. (Section 6.4.4)

Note: CMAF Segments are distinct from ISOBMFF segments and DASH segments.

CMAF Chunk
resource that contains a single ISO Base Media File Format segment, as defined by ISO/IEC 14496-12 section 8.16, that contains a sequential and contiguous subset of the samples of a CMAF Fragment (Section 6.4.5).

CMAF Resource
addressable CMAF media object, including CMAF Headers, CMAF Track Files, CMAF Segments, and CMAF Chunks. (Section 6.4)

CMAF Resource Identifier
identifier such as a URI, or URL and byte range, or other object identifier, that uniquely identifies a CMAF Resource within its scope of use.

CMAF Selection Set
set of CMAF Switching Sets, where each Switching Set encodes an alternative aspect of the same Presentation over the same time period, only one of which is intended to be played at a time, e.g. a different language or codec (Section 6.4.7).

CMAF Switching Set
set of CMAF Tracks, each of which is an alternative encoding of the same source content constrained to enable seamless Track switching and decoding (Section 6.4.6).

CMAF Track
sequence of CMAF Fragments of the same media stream in presentation order, and an associated CMAF Header sufficient to initialize playback. (Section 6.4.2)

CMAF Track File
CMAF Track stored in a single ISOBMFF file containing a CMAF Header and all CMAF Segments in sequence, starting from decode time zero. (Section 6.4.3

Manifest
document that describes one or more CMAF Presentations (Section 6.4.9)

Player
component of the CMAF application model responsible for interpreting a Manifest, requesting CMAF Segments, and rendering a CMAF Presentation

Presentation Time Offset
a CMAF Track's track presentation time at the start of a CMAF Presentation.

Required Media Profile
CMAF Track and Media Profile required in all CMAF Presentations conforming to that CMAF Presentation Profile.

Note: A CMAF Presentation Profile can conditionally require Media Profiles, e.g. subtitles can be optional, but if present, will be offered in the Required Media Profile(s).

Sample
all of the media data in a CMAF Fragment associated with a single timestamp.

The term “sample” is also used in the context of video to refer to the spatial samples of an image, and in the context of audio to refer to higher frequency temporal waveform samples. Unless qualified, the term “Sample” refers a file format media sample.

Selection
the choice, possibly by the user or using user preferences, of an alternative from a Selection Set (e.g. selecting an audio track by language).

Switching
dynamic choice by automated algorithm of an alternative Resource from a Switching Set, e.g. choosing the next Resource at a different bit-rate to adapt to the currently available network throughput.

Seamless Switching
switching without interrupting presentation of the media content i.e. decoding samples at the same time and quality as though their containing CMAF Track was decoded without Switching.

Single Initialization Switching Set
CMAF Switching Set constrained to allow initialization once by processing a CMAF Header, then Seamless Switching without additional CMAF Header processing. (Section 6.4.6)

Sub-sampling
video encoding using an exact fraction of the number of spatial samples in the source video, i.e. the scaling ratio yields a spatial sample count that is an integer or even integer, as specified in Section 9.

CMAF Presentation Profile
constraints on the CMAF Tracks and CMAF Media Profiles that are required to be included in a CMAF Presentation conforming to that Profile, as specified in Annex A.1

CMAF Media Profile: encoding constraints on a CMAF Track and its contained media samples indicated by a compatibility brand in the CMAF Header that is registered at MP4RA.org, and references a CMAF Track and Media Profile specification

3.2 Abbreviations and Acronyms

For the purposes of this Media Application Format, the following abbreviations apply.

AU	Access Unit
CDN	Content Delivery Network
CMAF	Common Media Application Format
CVS	Coded Video Sequence, as defined by the video format Note: typically, a sequence of samples (coded video frames), starting with a SAP type 1 or 2, and including all samples prior to the next SAP type 1 or 2
DASH	MPEG Dynamic Adaptive Streaming over HTTP (ISO/IEC 23009-1)
KID	MPEG Common Encryption Key Identifier
NAL	Network Adaptation Layer
PPS	Picture Parameter Set
SAP	Stream Access Point
SEI	Supplemental Enhancement Information
SPS	Sequence Parameter Set

VCL	Video Coding Layer
VPS	Video Parameter Set

4 Document Organization

Document sections are ordered from general to specific, starting with the scope of CMAF and basic definitions, then the media file specification, Common Encryption for all Track types, then specific Track formats.

Section 6 — CMAF Scope, Definition and Media Object Model describes the segmented media playback model and the associated objects defined by the CMAF.

Section 7 — The Common Media Application Format — describes the use of ISO file format for the Common Media Format brand.

Section 8 — Common Encryption of Tracks — details how digital rights management information and encryption is applied to the Common Media Format.

Section 9 — CMAF Video Tracks — describes the general video track format, specifics for NAL Structured Video tracks, and the AVC video format.

Section 10 CMAF Audio Tracks — describes the general audio track format, and specifics for AAC Media Profiles.

Section 11 Subtitles and Captions — describes the subtitle track format, and specifics Media Profiles for WebVTT and IMSC1 TTML subtitles, and signaling of CEA 608/708 captions embedded in video streams.

Annex A. CMAF Presentation and Media Profiles

Annex B. HEVC Media Profile and Track Format

Annex C. (Informative) Subsampling of Tracks in Track Switching Sets

Annex D. (Informative) Example Encoding Parameters for CMAF Switching Sets

5 Document Notation and Conventions

The following terms are used to specify conformance elements of this specification. These are adopted from the ISO/IEC Directives, Part 2, Annex H [ISO-P2H ISO-P2H]. For more information, please refer to those directives.

- SHALL and SHALL NOT indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.
- SHOULD and SHOULD NOT indicate that among several possibilities one is recommended as particularly suitable, without mentioning or excluding others, or that a certain course of action is preferred but not necessarily required, or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.
- MAY and NEED NOT indicate a course of action permissible within the limits of the document.

Terms defined to have a specific meaning within this specification will be capitalized, e.g. “Track”, and should be interpreted with their general meaning if not capitalized.

6 CMAF Scope, Definition and Media Object Model

6.1 Overview of Media Object Model

CMAF defines a logical media object model that provides the context to generate CMAF Resources. Figure 1 provides a high-level overview of the logical model and the mapping of the logical model to CMAF Resources. Resources provide the interfaces to storage and delivery systems by adding a CMAF Resource Identifier to CMAF external objects, namely CMAF Header, CMAF Track Files, CMAF Segments and CMAF Chunks.

The logical data model as well as the physical instantiations are defined in more details in the following.

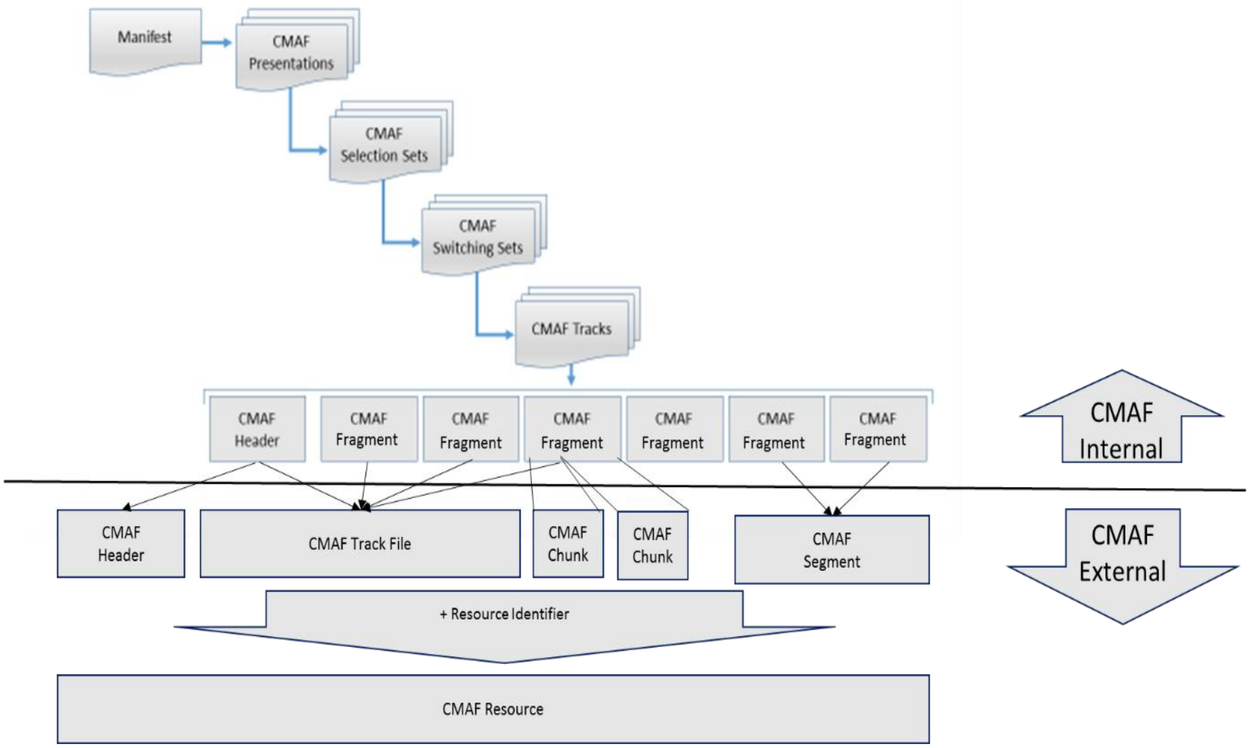


Figure 1 – CMAF Media Object Model

The CMAF media objects that are encoded and decoded are CMAF Headers and CMAF Fragments.

A presentation ordered sequence of CMAF Fragments and associated CMAF Header is a CMAF Track.

CMAF Tracks are grouped based on their content and encoding constraints in Switching Sets and Selection Sets.

One or more CMAF Selections Sets can be grouped and synchronized in a CMAF Presentation.

CMAF Fragments can be packaged, stored, delivered, and synchronized as CMAF Resources. Each CMAF Resource can be identified by a CMAF Resource Identifier in servers, networks, players, and Manifests.

CMAF defines CMAF Resources of four different types, which are the objects that define the external interface of CMAF:

1. CMAF Headers
2. CMAF Segments (containing one or more CMAF Fragments)
3. CMAF Chunks (each containing a portion of the samples of one CMAF Fragment) CMAF
4. CMAF Track Files (containing the CMAF Header and all CMAF Fragments of one Track)

A Manifest in the media object model is expected to describe the logical data structures in included in a CMAF Presentation as well CMAF Resources, and enables playback of a CMAF Presentation. The specific format of the Manifest is beyond the scope of this specification.

The same CMAF Resources and CMAF Presentation may be described by multiple Manifests and manifest formats, and may be combined with different CMAF Resources in different CMAF Presentations using different Manifests. The ability to reuse of the same CMAF Resources in different CMAF Presentations and Manifests is expected to improve encoding, storage, and delivery efficiency.

It is possible for Manifests to organize CMAF Presentations in sequences and other relationships. Examples include a sequence of Presentations, each containing a program or advertisement, in a TV channel, an ad Presentation that overlays a program

Presentation, a programmatic selection of a sequence of independent Manifests, manual selection of Presentations from a menu, etc.

6.2 CMAF Content Generation Process

The CMAF content generation processing model is shown in Figure 2. Each CMAF Presentation is composed of one or more media content components, for example audio components in various languages, different video components providing different views of the same program, subtitles in different languages, etc. Each media content component has assigned one of the following media content component type: audio, video or subtitle.

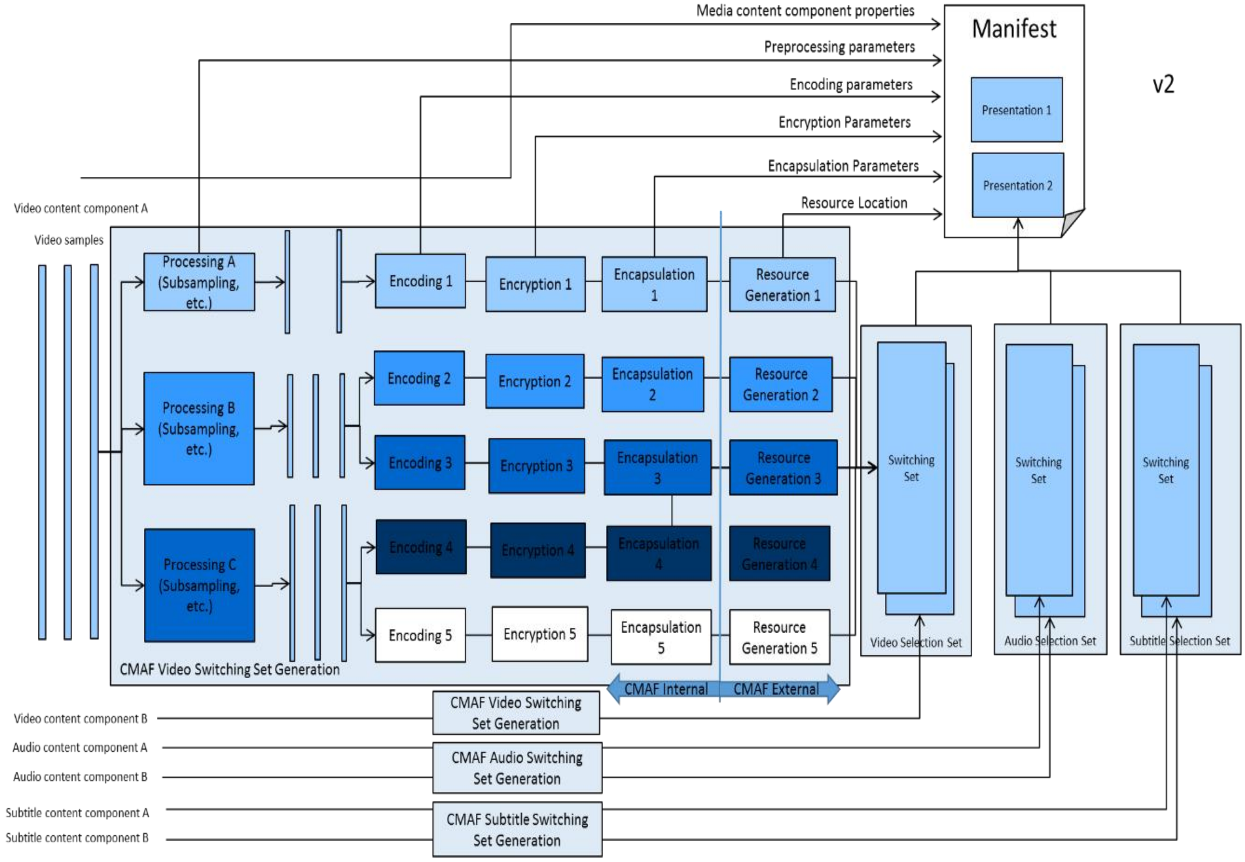


Figure 2— CMAF Content and Encoding Model

Each media content component may have been preprocessed (e.g. sub-sampling) and encoded in different versions of media streams, typically different in bitrate and resulting quality. Media streams are possibly encrypted and then encapsulated into CMAF Tracks. CMAF Tracks are finally made available as CMAF Resources. For live streaming, a CMAF Fragment in each CMAF Track can be encoded simultaneously from each multi-media content component, then each CMAF Fragment immediately encapsulated and made available as a CMAF Resource.

By definition, all CMAF Tracks of *all* media content components in one multi-media Presentation share the same media presentation time. This permits synchronization of different CMAF Tracks at the CMAF Player based on the common media presentation timeline. The feature of synchronizing separately stored CMAF Tracks is referred to as *late-binding*.

CMAF Tracks are arranged in Switching Sets. Only CMAF Tracks of the same media content component are added to a Switching Set, but CMAF Tracks of one media content component may be split into different Switching Sets.

By combining CMAF Tracks in one Switching Set conforming to a CMAF Media Profile, the CMAF Presentation author informs the CMAF Player that

- Any single CMAF Track within the Switching Set is sufficient to render the contained media content component.
- CMAF Tracks collected in the Switching Set are interchangeable encoded versions of one media content component
- The CMAF Tracks collected in the Switching Set represent perceptually equivalent content
- CMAF Tracks within a Switching Set are seamlessly switchable with encoding constraints and Adaptive Switching Process defined for each CMAF Media Profile

CMAF Track switching refers to the presentation of decoded data of one CMAF Track up to a media presentation time *t*, and presentation of decoded data of another CMAF Track from time *t* onwards. If CMAF Tracks are included in one Switching Set, and the Player applies the CMAF Track Switching Process defined by the CMAF Media Profile, then the presentation is perceived *seamless* across the switch for the media content component represented by the Switching Set. The CMAF Media Profiles and their track format specifications define the process required to seamlessly switch from one CMAF Fragment to another, without for example requiring parallel decoding or parallel downloading.

CMAF Tracks may be offered as CMAF Resources together with the metadata necessary to describe the content in a Manifest. A Manifest is expected to include the CMAF Resource Identifiers and Media Profile information to refer to CMAF Resources. The manifest is outside of scope of this specification, but Annexes provides examples how a CMAF Presentation can be described by different streaming manifests. The Manifest is expected to enable a CMAF Player to map the CMAF media object model and access CMAF Resources when they are available, determine the type of the CMAF Resource, synchronize the CMAF Resource in the CMAF Presentation, and access information necessary for decoding, decryption, and display of a multi-media presentation.

6.3 Specification Scope

CMAF specifies the following to support interoperability:

1. **CMAF Presentation Profiles** identify Presentations containing CMAF Tracks conforming to CMAF Media Profiles required by the Presentation Profile. Players can rely on the required Media Profiles being available for each media type (audio, video, or subtitles) contained in a Presentation identified by the Presentation Profile Identifier. Other CMAF Switching Sets and Media Profiles can also be included in the CMAF Presentation to enable additional features that might not be supported by all players conforming to the CMAF Presentation Profile. A CMAF Presentation may conform to and signal multiple Presentation Profiles if all the required CMAF Tracks are available.
2. **CMAF Media Profiles and ISOBMFF [ISOBMFF] compatibility brands** identify CMAF media sample and track format constraints for a CMAF Track that are specific to a codec operating point. Each Media Profile defines codec-specific properties required by all CMAF Tracks; such as the CMAF Header sample entry,

media sample format, CMAF Fragment constraints, sample presentation synchronization, scaling, mixing, layering, random access, and encoding constraints such as profiles and levels.

3. **CMAF Switching Set Adaptive Switching Processing and Constraints** for CMAF Switching Sets that conform to a CMAF Media Profile. Switching Set constraints between CMAF Tracks of the same Media Profile enable seamless switching using a specified Adaptive Switching Process. One or more Adaptive Switching Processes can optionally be defined by a CMAF Media Profile, and signaled in a Manifest so that Players can apply the correct Adaptive Switching Process to the Switching Set. A common example is a Switching Set identified as “bitstream switchable” by a manifest, which is constrained to continue decoding without processing a CMAF Header each time it is adaptively switched. However, the Adaptive Switching Process does require adaptive scaling per CMAF Fragment when a video codec such as AVC is used.
4. **Compatibility Brands for CMAF Resources:** CMAF Resources for CMAF Tracks provide the ability to offer CMAF Tracks as physical resources for delivery and storage applications. The specification defines how to package CMAF Fragments in CMAF Resources, and provides ISOBMFF segment compatibility brands for different CMAF Resources.

A CMAF Presentation Profile is specified in Annex A.1.

CMAF Media Profiles are specified in Annexes A.2, A.3, and A.4.

CMAF track formats and Adaptive Switching Processing are specified in Sections on AAC Track format, and AVC Track Format.

Additional CMAF Presentation Profiles can be specified that are identified by URN independent of the CMAF specification, if they conform to the general CMAF requirements.

Additional Media Profiles can be specified and identified by compatibility brands independent of the CMAF specification, if they conform to the general CMAF Track requirements, and define codec-specific CMAF Track formats, codec constraints, and optional Adaptive Switching Processes.

6.4 The Media Object Data Model

6.4.1 CMAF Fragments

CMAF Fragments are ISO Base Media File Format movie fragments, additionally constrained to be ISOBMFF [ISOBMFF] segments additionally constrained by the CMAF specification. CMAF Fragments are the CMAF media objects that are encoded and decoded.

CMAF Fragments are additionally constrained in order to conform to a CMAF Media Profile, and its CMAF Track and CMAF Switching Set constraints, and the general constraints of CMAF Selection Sets, and CMAF Presentations; as described below and specified in the normative sections of this specification.

The CMAF Media Profiles in Annex A.2 and their related track formats constrain CMAF video Fragments to contain one or more complete Codec Video Sequences to make them independently decodable. A CMAF Fragment typically consists of one ISOBMFF segment, but can contain more than one media data box, and can be chunked into more than one ISOBMFF segment.

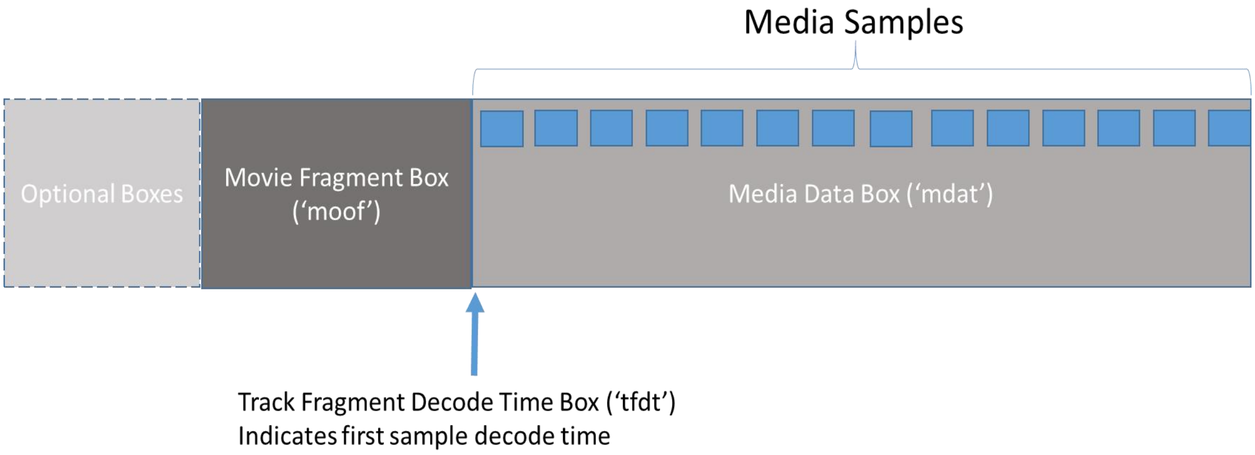

Media Samples
Optional Boxes
Movie Fragment Box ('moof')
Media Data Box ('mdat')
Track Fragment Decode Time Box ('tfdt')
Indicates first sample decode time

Figure 3 — Example CMAF Fragment data structure containing a single ISOBMFF segment

6.4.2 CMAF Tracks

A CMAF Track is a continuous sequence of CMAF Fragments of one Media Profile in presentation order and its associated CMAF Header. The CMAF Header contains a Movie Box (`'moov'`) sufficient to process and present all CMAF Segments in the CMAF Track.

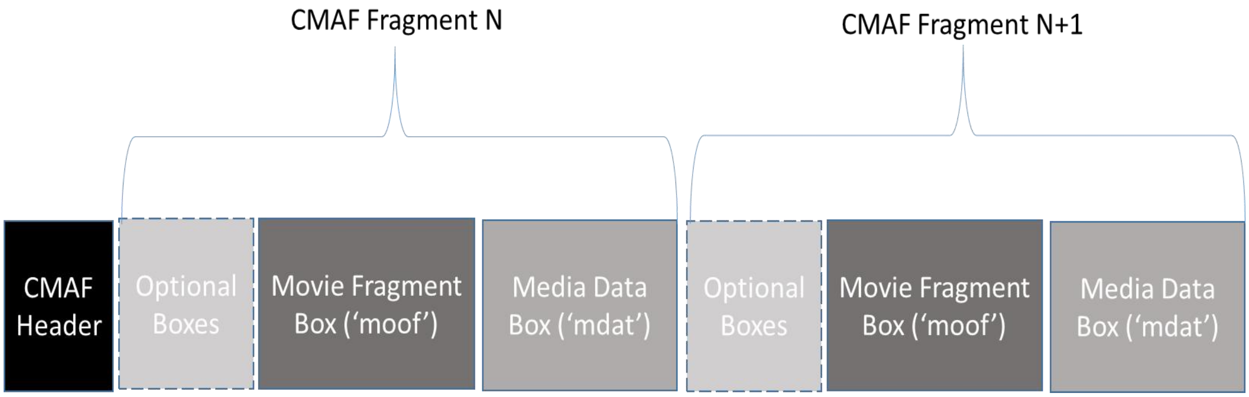


Figure 4 — CMAF Track Data Structure

See Sections 7 through 11 for additional details on the construction of CMAF Fragments and CMAF Tracks of different media types.

6.4.3 CMAF Track Files

A CMAF Track File is a complete CMAF Track stored in a single ISOBMFF file. A CMAF Track File starts with a CMAF Header followed by a continuous sequence of CMAF Fragments in decode order, and the first CMAF Fragment has a BaseMediaDecodeTime of zero. Additional boxes, such as Segment Index Boxes (`‘sidx’`), may be present between the CMAF Header and the first CMAF Fragment.

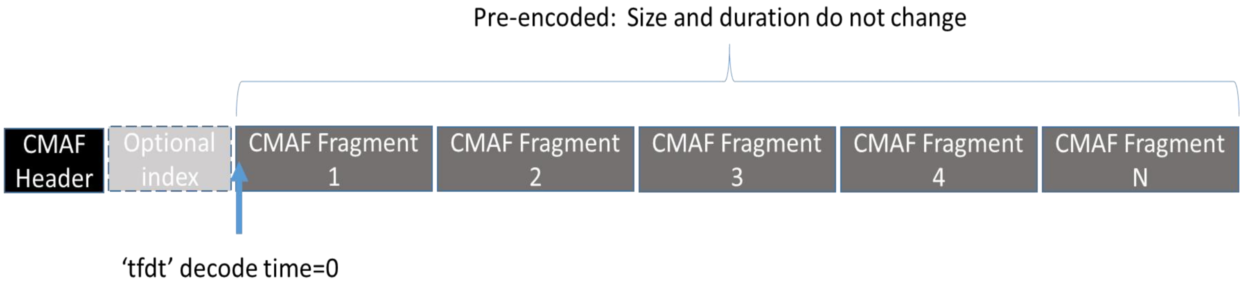


Figure 5 — CMAF Track File Structure

A sequence of one or more CMAF Fragments can be requested from a Track File Resource using HTTP 1.1 and a Resource Identifier consisting of the file URL and a byte range. CMAF Resource Identifiers can be listed in a manifest or determined by some other delivery format defined method, such as downloading a Segment Index Box (`'sidx'`) to determine the byte ranges of Fragments. The manifest and request method is out of scope of CMAF.

6.4.4 CMAF Segments

One or more CMAF Fragments from a CMAF Track can be packaged in a CMAF Segment, which is then typically identified by a Resource Identifier by servers and Manifests to reference, deliver, and synchronize each Segment in a CMAF Presentation.

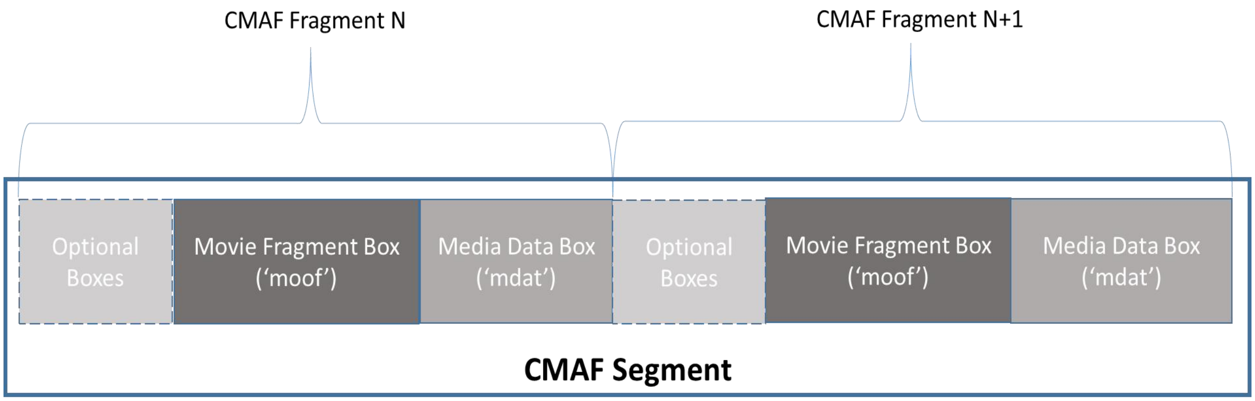


Figure 6 — Example CMAF Segment Data Structure

CMAF video Fragment durations are typically 2–6 seconds for coding efficiency, and CMAF Segment durations are typically not greater than 10–12 seconds to limit delivery latency in live streaming and bitrate rate adaptation response time.

Subtitle Segment durations are usually similar to video Segment durations in live Presentations to avoid increasing presentation delay. In a prerecorded CMAF Presentation, a single Subtitle CMAF Segment can have a duration up to the duration of the Track that contains it.

6.4.5 CMAF Chunks

CMAF Chunks are Resources primarily used to deliver media samples before they can be packaged in a complete CMAF Segment during live encoding and streaming. CMAF Chunks enable the progressive encoding, delivery, and decoding of each CMAF Fragment. Broadcast and multicast protocols can deliver and identify CMAF Chunks through other methods.

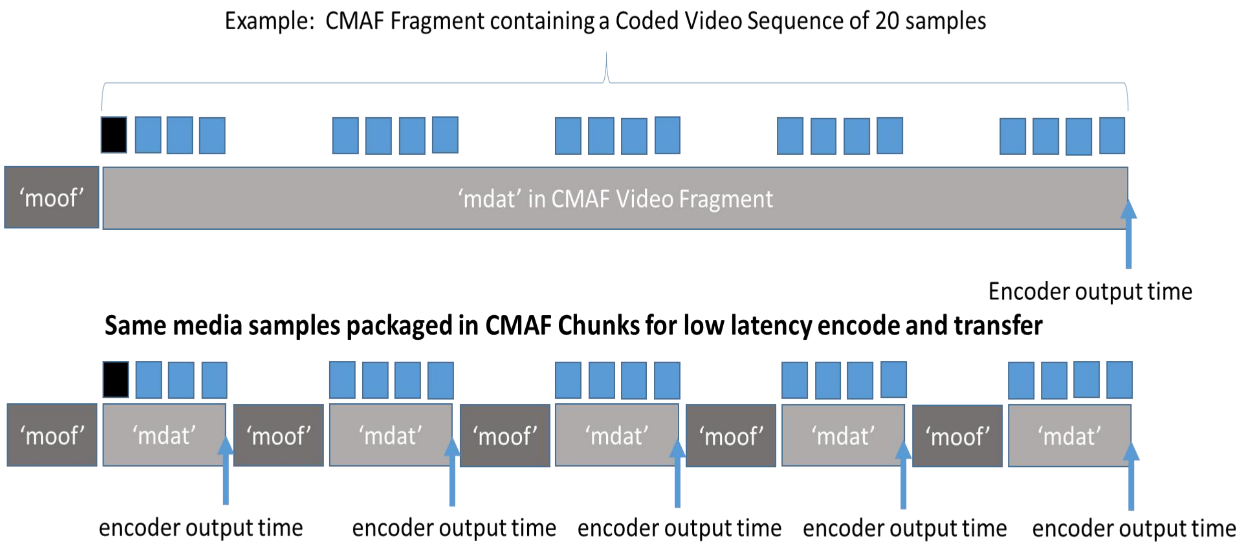


Figure 7 — CMAF Chunk Data Structure

6.4.6 CMAF Track Switching Sets and Adaptive Switching

6.4.6.1 Introduction

A CMAF Switching Set is a collection of CMAF Tracks, where each Track is a different encoding of the same source content. Switching Sets contain time aligned CMAF Fragments start with stream access points to simplify switching between Tracks by sequencing CMAF Segments from different CMAF Tracks during playback. Seamless Switching allows Players to adapt to network and other conditions.

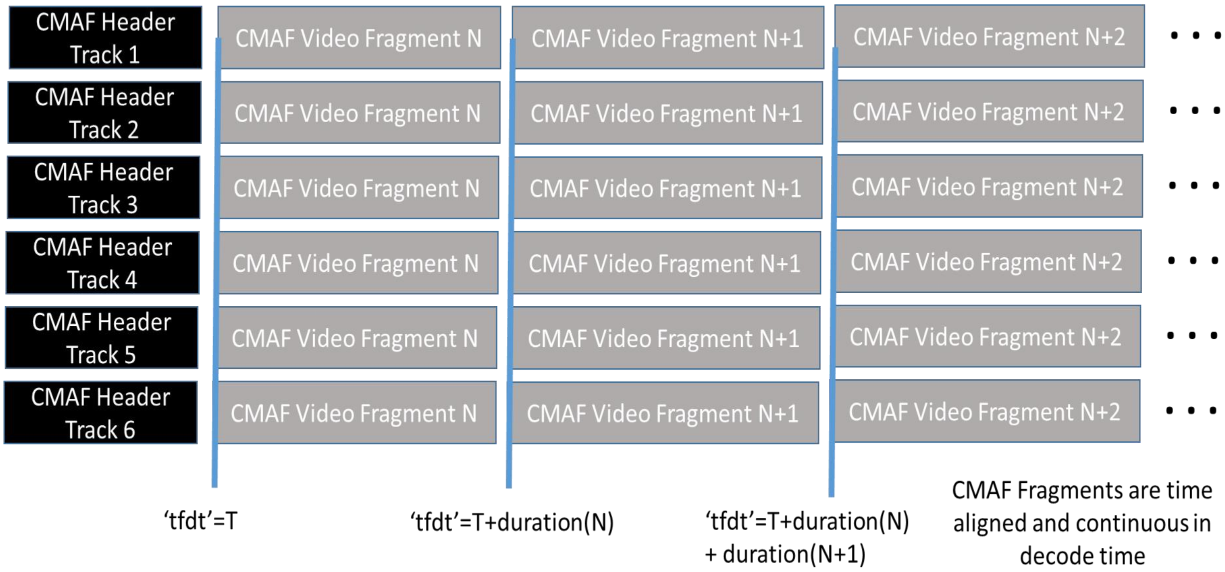


Figure 8 — CMAF track switching set

The Manifest description of CMAF Switching Sets and the CMAF Tracks they contain is out of scope of the CMAF specification, but Manifests are expected to contain enough CMAF Track and Segment information to enable automatic selection and adaptive switching by Players.

6.4.6.2 Decoding Adaptively Switched Segments

Each CMAF Media Profile can specify one or more Adaptive Switching Processes that define how Players are expected to adaptively switch between alternative CMAF Fragments in a CMAF Switching Set. The Adaptive Switching Process defines operations such as decoder and decryptor initialization and re-initialization, video scaling, audio mixing, sample presentation timing, etc. that are intended to make adaptive switching as perceptually seamless as possible.

The Media Profiles in the CMAF specification define Adaptive Switching Processes that constrain Switching Sets and switching points to produce a stream of CMAF Headers and CMAF Fragments that will decode on most decoders and not require the download of duplicate content in overlapping CMAF Fragments. It is possible for other Media Profiles to specify Adaptive Switching Processes that rely on multiple decoders, layering of multiple

CMAF Tracks, etc. to adapt bitrate, frame rate, resolution, point of view, audio scenes and objects, etc.

An adaptively delivered CMAF Track conforming to a Media Profile in Annex A. is a timed sequence of CMAF Headers and Fragments selected from multiple CMAF Tracks in a Switching Set. The encoding constraints of the Switching Set and the associated Adaptive Switching Process determine the bitstream delivered to a parser and decoder. Typical ISOBMFF [ISOBMFF] file parsers and decoders can decode the concatenated sequence of CMAF Fragments the same as a fragmented movie file, with the possible exception of needing to re-initialize the decoder at switch points.

An adaptive streaming Player typically switches CMAF Tracks to adapt the selected bitrate to the current throughput of the network to maintain a safe level in the Player’s buffers in order to maximize bitrate and quality, but prevent an interruption in the presentation.

The encoding constraints of the CMAF Tracks in a Switching Set and associated Adaptive Switching Process determine if the Switching Set can be initialized once with a CMAF Header, or if it needs to process a CMAF Header for each CMAF Track switch. A Switching Set Process can be specified by a Media Profile and its track format to identify Switching Set encoding constraints required for these two processing models, and potentially others that depend on the codec, track formats features such as layering, etc.

See section [7.3.9] for details on CMAF Switching Sets.

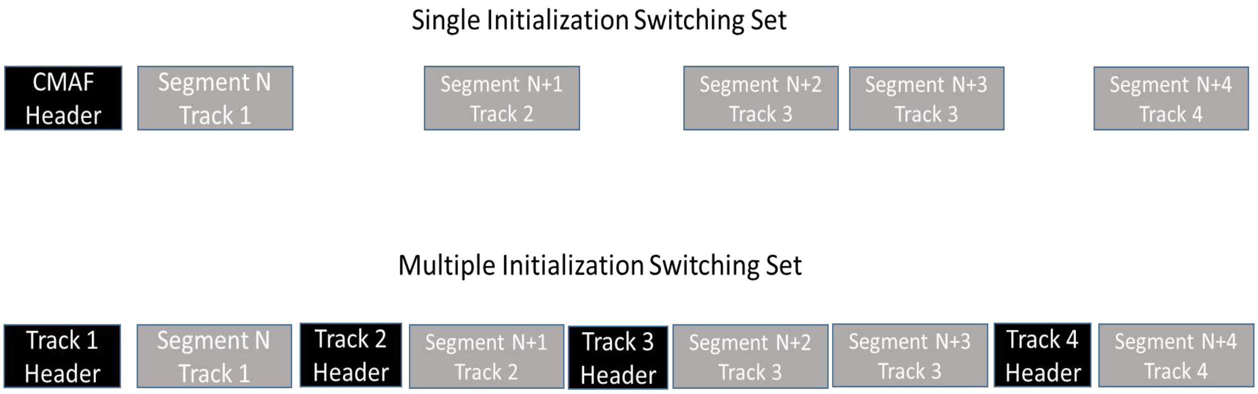


Figure 9 — Adaptive Switching Processing models for different Switching Sets

The normative behavior of video decoders can be well-specified, but display processing considered out of scope, i.e. conversion by a display processor of YCbCr 4:2:0 subsamples from the decoder to some number of pixels in a color space appropriate for a particular display. Adaptive Switching Processes defined by CMAF video Media Profiles expect Players to scale different video spatial sampling used to encode different CMAF Fragments to the same display size and position. The encoding of CMAF Tracks in a Switching Set is constrained so that CMAF Fragments can be accurately rescaled to the same presentation height and width, and that will result in precise registration and appearance of the video content for visual continuity, including its size, shape, location, color, brightness, etc. The constraints specified for CMAF Tracks and Switching Sets determine the scaling behavior needed in display processors, but display processing and other Player requirements are not directly specified by CMAF.

The CMAF Header of a CMAF video Track typically needs to be processed on each CMAF Track switch if each CMAF Track only stores the video decoding parameter sets for that CMAF Track in the CMAF Header. A decoder and display processor can then index the correct decoding, cropping, and scaling parameters in the sample entry using the index values in the video slice headers. This is the multiple initialization Adaptive Switching Processing model.

Video CMAF Fragments from the same Switching Set, encoded with sequence and picture parameter sets stored in each CMAF Fragment can be decoded without processing a CMAF Header on CMAF Track switches because each CMAF Fragment contains the necessary decoding, cropping, and scaling parameters referenced by video slices in that CMAF Fragment. Other constraints are required for a single initialization Switching Set, such as no edit list or the same edit list in all CMAF Headers in the Switching Set. This is the single initialization Adaptive Switching Processing model. These two Adaptive Switching Processes can be used by many codecs and Media Profiles.

6.4.7 CMAF Track Selection Sets and Late Binding

A CMAF Selection Set is a set of CMAF Switching Sets, where each Switching Set encodes an alternative aspect of the same program over the same time period — for example, different audio languages, video camera angles, video formats, or codecs.

CMAF Players are expected to select one Switching Set from each Selection Set at the start of playback based on Player compatibility and user preferences. Users or playback applications may switch between Switching Sets in a Selection Set during playback, but seamless presentation is not expected, either because the content differs (e.g. language or camera), or because CMAF Segment time alignment and decoding are not constrained to decode seamlessly. The process of selecting independently stored CMAF Tracks for synchronized presentation is called “late binding”. Late binding allows CMAF Tracks to be encoded once and used in many different combinations.

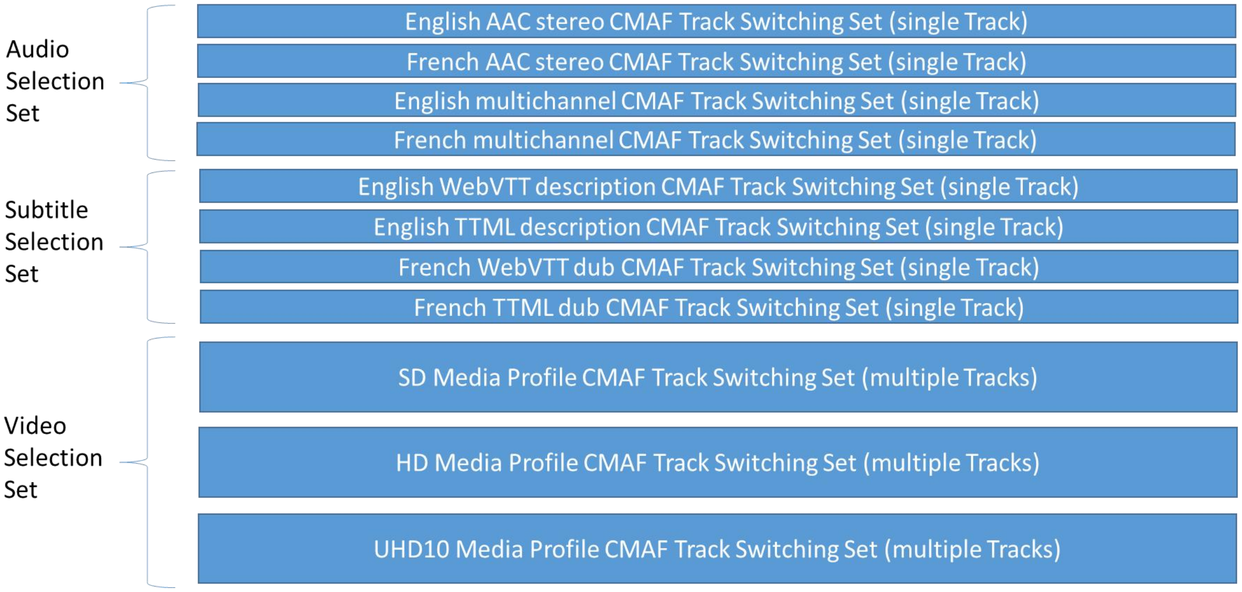

Audio Selection Set
English AAC stereo CMAF Track Switching Set (single Track)
French AAC stereo CMAF Track Switching Set (single Track)
English multichannel CMAF Track Switching Set (single Track)
French multichannel CMAF Track Switching Set (single Track)
Subtitle Selection Set
English WebVTT description CMAF Track Switching Set (single Track)
English TTML description CMAF Track Switching Set (single Track)
French WebVTT dub CMAF Track Switching Set (single Track)
French TTML dub CMAF Track Switching Set (single Track)
Video Selection Set
SD Media Profile CMAF Track Switching Set (multiple Tracks)
HD Media Profile CMAF Track Switching Set (multiple Tracks)
UHD10 Media Profile CMAF Track Switching Set (multiple Tracks)

Figure 10 — Example of CMAF Selection Sets

The description of Selection Sets in a Manifest is not specified by the CMAF specification, however each Manifest format is expected to provide sufficient information to enable Players to automatically select optimal CMAF Track Switching Sets from CMAF Selection Sets.

When there are multiple tracks in a Selection Set that differ by language, the language of each track needs to be identified to at least the precision needed to differentiate the tracks, using the language field and optionally the Extended Language Box (`'elng'`) as defined in [ISOBMFF] 8.4.6. See also 7.5.5.

6.4.8 CMAF Presentation and Timing Model

A multimedia CMAF Presentation is a group of one or more synchronized Selection Sets that are intended for simultaneous presentation. There is typically one Selection Set for each media type (e.g. audio, video, subtitles, or metadata), which may contain one or more Switching Sets with one or more CMAF Tracks. All CMAF Tracks in a CMAF Presentation are encoded and decoded synchronized to a common presentation timeline.

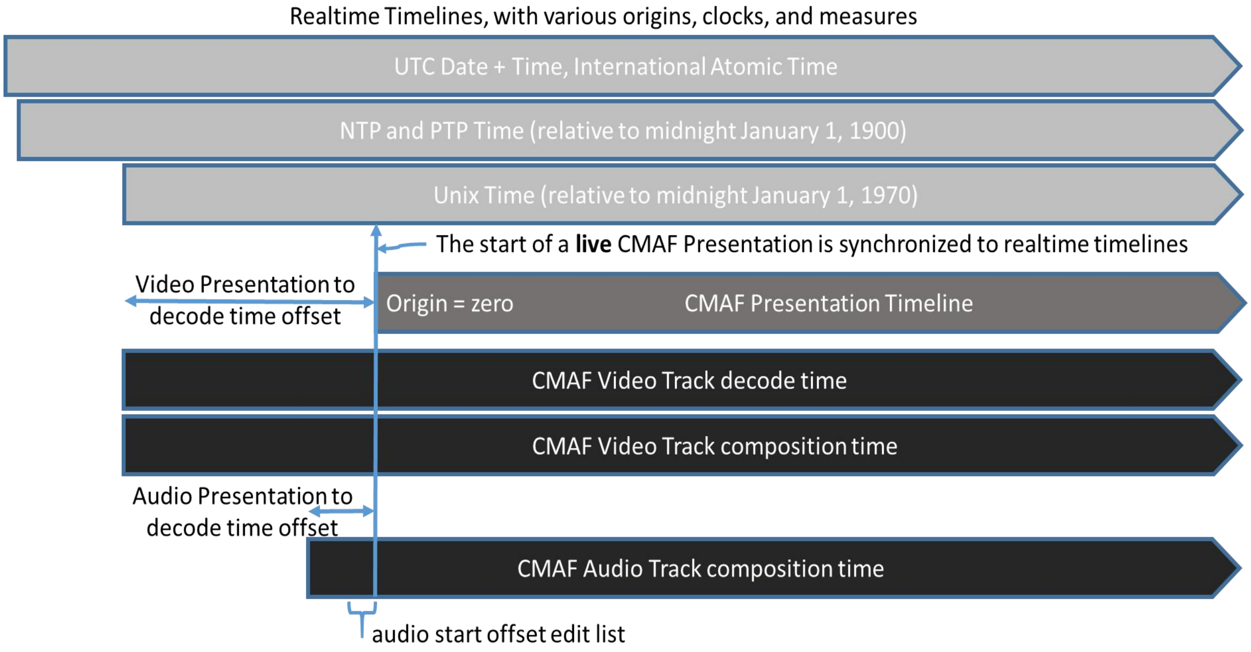

Realtime Timelines, with various origins, clocks, and measures
UTC Date + Time, International Atomic Time
NTP and PTP Time (relative to midnight January 1, 1900)
Unix Time (relative to midnight January 1, 1970)
The start of a **live** CMAF Presentation is synchronized to realtime timelines
Video Presentation to decode time offset
Origin = zero
CMAF Presentation Timeline
CMAF Video Track decode time
CMAF Video Track composition time
Audio Presentation to decode time offset
CMAF Audio Track composition time
audio start offset edit list

Figure 11 — CMAF timeline and synchronization model

There are multiple timelines involved in synchronizing a CMAF Presentation. Each timeline has a timescale in units per second, increases over time, and has an origin where the measure equals zero.

The timelines are:

1. Track decode time (determined by the storage sequence and duration of each media sample, equal to prior sample durations added to the `baseMediaDecodeTime` of each CMAF Track, Fragment, or sample)

2. Track composition time (determined by each sample's composition offset in the track run box relative to its decode time, which reorders video samples to their presentation order, and can introduce a delay relative to audio and the presentation timeline)

3. Track presentation time (determined by applying any offset edit list in the CMAF Header to the track composition timeline, as shown for a CMAF audio Track in Figure 11)

4. CMAF Presentation Timeline (determined by synchronizing all CMAF Tracks in a CMAF Presentation to CMAF Presentation time zero, and typically representing this in a manifest)
5. Clock time, e.g. UTC time, either at the time of encoding, or at the time of playback (this timeline may not be relevant for VOD Presentations, may be stored in a Producer Reference Time Box for a CMAF Track, or in a manifest linked to the CMAF Presentation Time)

CMAF abstracts the timing model of ISOBMFF tracks and files to apply to late binding, live streaming, and different combinations of CMAF Tracks in different CMAF Presentations.

Separately stored CMAF Tracks can be encoded and packaged with an edit list in the CMAF Header to remove composition delays added by positive video composition offsets and/or trim leading samples from CMAF Tracks (typically audio), so that the first presented audio, video, and subtitle samples will start simultaneously as well as present synchronously over the CMAF Presentation Timeline. Accurately capturing synchronization information in the CMAF Tracks in a CMAF Presentation determines the presentation synchronization that manifests need to describe to accurately place CMAF Presentations on their presentation timelines.

The ISOBMFF primarily defines files, where the decode time of each sample (`BaseMediaDecodeTime)` is the sum of prior sample durations in that track in stored order. The first sample in each track in a file has a decode time of zero. Movie fragmentation is optional.

In the case of CMAF Tracks, movie fragments are required, and the first CMAF Fragment may have a non-zero `baseMediaDecodeTime` in the Track Fragment Decode Time Box (`'tfdt'`). The decode time of each sample equals the sum of prior sample durations in the CMAF Track added to the `baseMediaDecodeTime` of the first CMAF Fragment in the CMAF Track. And, the decode time of each sample also equals the sum of prior sample durations in the CMAF Fragment that contains it, added to its `BaseMediaDecodeTime`.

Decode time is continuous by ISOBMFF definition in each CMAF Track, and that enables synchronization between independently stored CMAF Tracks on a common presentation timeline determined by a Manifest.

Manifests may specify a manifest presentation time to CMAF Track presentation time offset for each CMAF Switching Set consistent across tracks with the CMAF Presentation Timeline to determine the first sample in each CMAF Track that will be presented at the start of the CMAF Presentation, and the duration of the CMAF Presentation on the manifest presentation timeline. A manifest or player can select a subset of the CMAF Presentation Timeline for playback, but has to apply the track presentation time offsets valid at CMAF Presentation Timeline zero to maintain the synchronization encoded in the CMAF Tracks.

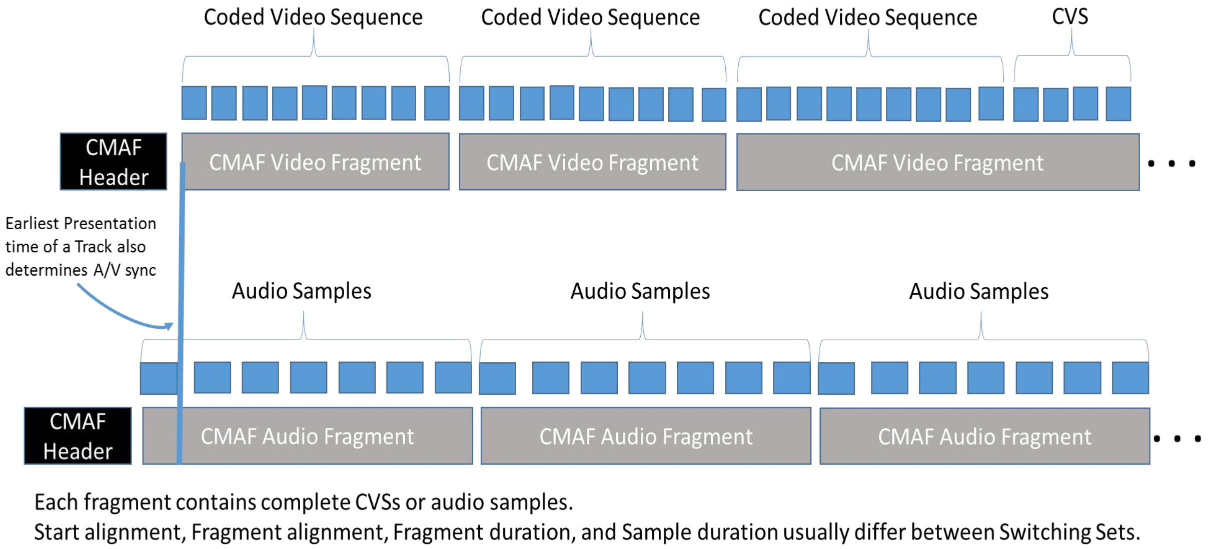


Figure 12 — Audio Video Synchronization between Selection Sets

CMAF Tracks containing video can use negative composition offsets where necessary to remove composition delay, so that the composition time interval of each CMAF Fragment will equal its decode time interval, and no edit list is necessary to remove composition timeline delay. Video Switching Sets normally align to the start of a CMAF Presentation on a CMAF Fragment boundary, so no edit list is necessary to adjust the earliest track presentation time to match the CMAF Presentation Timeline zero. In that case, the baseMediaDecodeTime in the Track Fragment Decode Time Box ('tfdt') is the earliest sample composition and presentation time of the CMAF Fragment. Because CMAF Fragments contain complete coded video sequences (often one), sample reordering happens within the CMAF Fragment and does not change the earliest sample presentation time or CMAF Fragment duration. If alternative CMAF Tracks in a CMAF Switching Set have different picture removal delays and composition offsets due to different encoded picture size and number of reference frames used, it will not result in different composition timeline delays and edit lists between CMAF Tracks.

CMAF Switching Sets, especially audio, might contain CMAF Fragments that are not aligned to the CMAF Presentation Timeline zero. An Audio CMAF Fragment usually overlaps the start of a video CMAF Fragment, except at the start of a synchronized encoding. An edit list can be used to exclude the earlier samples in the CMAF Fragment from presentation. A Manifest can also link CMAF Presentation Timeline zero to a UTC time to indicate the earliest availability of CMAF Segments that are being encoded or made available in realtime.

During random access and “trick play” (fast forward, reverse, slow motion, etc.), the Track Fragment Decode Time Box (‘tfdt’) can be used to calculate the presentation time from the CMAF Fragment baseMediaDecodeTime, taking any edit lists into account.

Decode time discontinuities in CMAF Track timelines are considered a new CMAF Track in a new CMAF Presentation. A Manifest can locate the new CMAF Tracks on its presentation timeline by applying a new track presentation time offset for each CMAF Track at the start of a new CMAF Presentation (e.g. a DASH Period).

6.4.9 Manifest

Although CMAF does not define the form or the content of the Manifest, it does define its role. A Manifest is a document that describes one or more CMAF Presentations; e.g. the MPEG DASH MPD.

A Manifest provides the Player with information to select, initialize, and synchronize the CMAF Track(s) to be played, and identify CMAF Headers and Resources, and possibly download them synchronously. CMAF Tracks and Fragments contain sufficient information to enable decryption, decoding, synchronization, and rendering once CMAF Fragments are located in CMAF Resources. A Manifest can also provide information on delivery protocol, network management, authorization, license acquisition, etc. in addition to Resource identification and Presentation description, but those are optional for the hypothetical presentation model. The manifest can also signal whether tracks are CMAF conformant.

A Manifest is responsible for describing the combination and synchronization of independently encoded CMAF Switching Sets grouped in Selection Set to form a synchronized multimedia presentation. A common presentation timeline, similar to an ISOBMFF movie timeline, can be used to synchronize the timestamps of each CMAF Track so that a CMAF Player can “late bind” the selected audio, video, and subtitle Switching Sets. In cases where there are multiple CMAF Tracks in a Switching Set, a Manifest can indicate to a Player how it can seamlessly switch between CMAF Tracks to optimize for the network bandwidth and device capability.

Manifests are expected to include any recorded audio offset edit list duration in Manifest synchronization information so that audio Switching Set synchronization is precisely described relative to video to the accuracy of the media timescales. A Manifest may specify a presentation start time referencing wall clock time, e.g. UTC time; typically for a live Presentation.

A Manifest can specify a growing Presentation duration for each Selection Set and its contained Switching Sets and Tracks during live encoding and playback.

A Manifest can remove earlier Segments from a Presentation. A Manifest may specify an availability start and end time (in wall clock time) for an entire Presentation.

A Manifests can include additional information to indicate start and end times within CMAF Presentations to begin and end media playback at presentation times different from the start and end times of the CMAF Resources described. For example, edit information can limit the portion of a CMAF Presentation that is presented to CMAF Fragment or sample accuracy.

6.5 Hypothetical Player Model

6.5.1 Overview

Player implementers should note that CMAF provides the following affordances:

- A CMAF Segment is well-suited to network transfer because it is a compact, self-contained set of media samples that covers a single, short period of time, and can be sequenced with other CMAF Segments in a single track parser/decoder and browser media source buffer without additional bitstream splicing and editing.
- Each CMAF Segment contains a media timestamp in the form of BaseMediaDecodeTime in the Track Fragment Decode Time Box (`'tfdt'`), which allows individual CMAF Segments and CMAF Tracks that start with arbitrary timestamps to be synchronized to a presentation timeline
- The movie fragment box (‘`moof'`) and the sample data contain everything necessary to render the samples at the correct place on the playback timeline after a Track has been initialized with a CMAF Header.
- CMAF Tracks can be separately selected and delivered by a player, then synchronized at presentation time, thus allowing each player to customize the presentation for the device, network, and user.
- The picture at the start of each video CMAF Segment can be obtained with minimal transfer activity for fast forward or fast reverse streaming.
- Selection Sets and single track CMAF Segments allow alternative content to be offered for playback without requiring the Player to transfer content that it does not intend to play.

- Switching Sets allows alternative bitrate and resolution encodings of the same content to be seamlessly presented through a single video decoder.
- The use of Common Encryption supports access to the same content by multiple decryption key delivery systems.
- The `'emsg'` box allows Segments to signal application-defined Track events with low latency during live presentations without requiring frequent Manifest downloads.

6.5.2 Adaptive Streaming Playback (Informative)

In order to play an adaptive streaming Presentation, a Player typically:

- parses the Manifest and selects Selection Sets of media types it can present on that device (e.g. audio only, or audio/video/subtitles/Picture in Picture, etc.).
- compares Switching Set information to Player, decoder, display, DRM, etc. capabilities to determine the compatible Switching Sets in those Selection Sets it can play.
- selects the most preferred and compatible Switching Set in each Selection Set, sometimes based on stored user preferences (language, accessibility, rating, stereo or multichannel audio, etc.).
- selects an initial Track from each selected Switching Set, usually based on estimated network bandwidth, rapid start heuristics, display size, etc.
- initializes, decodes, synchronizes, and presents the selected Tracks, and automatically requests each CMAF Segment in sequence from that Switching Set, adapting requested bitrates to maintain continuous playback within the limitations of network throughput.

Live Presentation playback is typically optimized for low latency, so a player only buffers a few seconds of each selected CMAF Track in the player to minimize presentation delay. A different bitrate could be selected for each CMAF Segment requested from a Switching Set in order to prevent buffer underflow in the player while maximizing media quality. Once a live presentation delay and buffer duration is selected, the delay can’t be changed without halting playback and rebuffering, or decoding at a speed faster or slower than normal, which is usually not acceptable. Measurements of network latency, jitter, throughput rate, throughput variation, Segment duration variation, and server/client clock synchronization can help a player select an optimal presentation delay and next Segment bitrates to request. To minimize visible changes between CMAF Segments encoded at different bitrates, live CMAF Switching Sets may include more Tracks with smaller bitrate differences, for instance a decrease of 30% to the next lowest bitrate Track.

Recorded (video on demand) Presentation playback is typically optimized for infrequent switching and infrequent rebuffering. Players can accumulate several minutes of buffer time during playback by selecting lower bitrate Segments that can be downloaded faster than they are decoded. Once sufficient buffer time has accumulated; an optimum bitrate can be selected. Tracks may be rarely switched because the long player buffer duration averages short term changes in network throughput. Because bitrate changes are less frequent and therefore less noticeable, fewer Tracks with larger bitrate differences may be considered acceptable in order to reduce encoding and storage, for instance a decrease of 50% to the next lowest bitrate Track.

When Players adaptively switch CMAF video Tracks, they typically rescale the decoded and cropped image to the Player selected display aperture. Lower bitrate CMAF Tracks in a Switching Set are typically encoded at lower resolutions (subsampled) in corresponding to their bitrate. If video samples are stored with all decoding parameters in the CMAF header, the CMAF Header has to be inspected or processed before the first CMAF Segment from a new Track so that the decoding, cropping, and scaling will apply the correct parameters for the following CMAF Fragments from that Track. If video samples are stored inline, inspecting or processing a CMAF Header is not necessary, except for the first CMAF Fragment of a Switching Set because each CMAF Fragment contains the necessary decoding parameters in the first video sample.

See Annex C. Annex D. for recommendations on encoding adaptive Switching Sets.

7 The Common Media Application Format File

7.1 Introduction

The Common Media Application Format is derived from the ISO Base Media File Format, and is primarily a profile of that format.

7.2 CMAF Brands

This section defines the requirements for CMAF tracks containing any coded media. The media coding requirements mandatory in a presentation are expressed in Annex A.

The CMAF file-type brand indicating conformance with this section (file format, but not coding) is `'cmfc'`. The requirements of this brand include the requirements of the brand `'iso9'` [ISOBMFF]. It also includes boxes specified in Common Encryption [CENC], [DASH], MPEG-4 Part 15 for storage of NAL Structure video [ISOVIDEO], and those boxes defined by referenced audio specifications, as defined in Section 10.

If the `'cmfc'` brand is listed in `compatible_brands`, the file SHALL conform to the requirements in section 7. An ISO brand (`'iso9'` or lower) SHALL be listed in `compatible_brands`.

> Note: File readers should read possible future versions of CMAF that increment the `minor_version` number.

A CMAF track SHALL include the `'cmfc'` file brand, whether it contains an externally specified Media Profile brand that conforms to Track requirements for it media type (audio, video, or subtitles), or the CMAF Media Profile brands specified in Annex A.

If `'cmfc'` is the `major_brand`, the `minor_version` SHALL be set to 0, and file names SHOULD use the file extensions in Table 1. Otherwise, file names SHOULD use the file extension and Internet Media Type specified to match the major brand brand, e.g. `*.mp4`, `*.3gp`, `*.uvu`, etc.

Table 1 — Common Media Application Format File Extensions

Track type	File extension	Internet Media Type (MIME type)
Video	`.cmfv`	`video/mp4`
Audio	`.cmfa`	`audio/mp4`
Text (Subtitle)	`.cmft`	`application/mp4`

If `'cmfs'` is listed in the ISO Media segment-type `compatible_brands`, it SHALL conform to the requirements in section 7.3.6 on CMAF Segments. If `'cmfl'` is listed in the ISO Media segment-type `compatible_brands`, it SHALL conform to the requirements in section 7.3.7 on CMAF Chunks.

Table 2 — Common Media Application Format Brands

Brand	Location	Conformance Requirements
`'cmfc'`	File Type Box and Segment Type Box	Section 7
`'cmfs'`	Segment Type Box	Section 7.3.6 CMAF Segments
`'cmfl'`	Segment Type Box	Section 7.3.7 CMAF Chunks

The primary constraints of the `'cmfc'` file type brand are described in Sections 7.3.4 and 7.3.5.

Each Track has a CMAF Header associated with it, although the CMAF Header and CMAF Segments might not be stored as an ISO Media file, if they are stored at all.

7.3 CMAF File Syntax

7.3.1 CMAF Boxes

CMAF Tracks SHALL include the following boxes with nesting, optionality, and ordinality specified in the following tables. Other boxes MAY be included.

Table 3 — Common Media Application Format Track Header boxes

NL 0	NL 1	NL 2	NL 3	NL 4	NL 5	Format Req.	Specification	Requirements	Description
ftyp						1	[ISOBMFF] 4.3	Section 7.2	File Type and Compatibility
moov						1	[ISOBMFF] 8.2.1		Container for functional metadata
	mvhd					1	[ISOBMFF] 8.2.2		Movie header
	trak					+	[ISOBMFF] 8.3.1		Container for each track
		tkhd				1	[ISOBMFF] 8.3.2		Track header
		edts				CM	[ISOBMFF] 8.6.5		Edit Box
			elst			CM	[ISOBMFF] 8.6.6		Edit List Box
		mdia				1	[ISOBMFF] 8.4		Track Media Information
			mdhd			1	[ISOBMFF] 8.4.2	Section 7.5.5	Media Header
			hdlr			1	[ISOBMFF] 8.4.3		Declares the media handler type

NL 0	NL 1	NL 2	NL 3	NL 4	NL 5	Format Req.	Specification	Requirements	Description
			elng			0/1	[ISOBMFF] 8.4.6		Extended Language Tag
			minf			1	[ISOBMFF] 8.4.4		Media Information container
				vmhd		CM	[ISOBMFF] 12.1.2	Section 7.5.6	Video Media Header
				smhd		CM	[ISOBMFF] 12.2.2	Section 7.5.7	Sound Media Header
				sthd		CM	[ISOBMFF] 12.6.2		Subtitle Media Header
				dinf		1	[ISOBMFF] 8.7.1		Data Information Box
					dref	1	[ISOBMFF] 8.7.2	Section 7.5.8	Data Reference Box, declares source of media data in track
				stbl		1	[ISOBMFF] 8.5		Sample Table Box, container for the time/space map
					stsd	1	[ISOBMFF] 8.5.2	Section 7.5.9	Sample Descriptions (See Table 2-2 for additional detail.)
					stts	1	[ISOBMFF] 8.6.1.2	Section 7.5.11	Decoding, Time to Sample
					stsc	1	[ISOBMFF] 8.7.4	Section 7.5.11	Sample–to–Chunk
					stsz / stz2	1	[ISOBMFF] 8.7.3	Section 7.5.11	Sample Size Box
					stco	1	[ISOBMFF] 8.7.5	Section 7.5.11	Chunk Offset

NL 0	NL 1	NL 2	NL 3	NL 4	NL 5	Format Req.	Specification	Requirements	Description
					sgpd	*	[ISOBMFF] 8.9.3	Section 7.5.17	Sample Group Description Box
		udta				0/1	[ISOBMFF] 8.10.1		User Data Box
			cprt			+	[ISOBMFF] 8.10.2		Copyright Box
			kind			+	[ISOBMFF] 8..10.4		Track Kind Box
	mvex					1	[ISOBMFF] 8.8.1		Movie Extends Box
		mehd				0/1	[ISOBMFF] 8.8.2		Movie Extends Header
		trex				+ (1 per track)	[ISOBMFF] 8.8.3	Section 7.5.13	Track Extends Box
	pssh					*	[CENC] 8.1		Protection System Specific Header Box

Table 4 – Common Media Application Format Track Segment boxes

NL 0	NL 1	NL 2	NL 3	NL 4	NL 5	Format Req.	Specification	Requirements	Description
styp							[ISOBMFF] 8.16.2		Segment Type
prft							[ISOBMFF] 8.16.5		Producer Reference Time
emsg							[DASH]	Section 7.4.5	Event Message
moof						+	[ISOBMFF] 8.8.4		Movie Fragment
	mfhd					1	[ISOBMFF] 8.8.5	Section 7.5.14	Movie Fragment Header
	traf					1	[ISOBMFF] 8.8.6		Track Fragment

NL 0	NL 1	NL 2	NL 3	NL 4	NL 5	Format Req.	Specification	Requirements	Description
		tfhd				1	[ISOBMFF] 8.8.7	Section 7.5.15	Track Fragment Header
		tfdt				1	[ISOBMFF] 8.8.12		Track Fragment Base Media Decode Time
		trun				1	[ISOBMFF] 8.8.8	Section 7.5.16	Track Fragment Run Box
		senc				0/1	[CENC]		Sample Encryption Box
		saio				* see note 1	[ISOBMFF] 8.7.13		Sample Auxiliary Information Offsets Box
		saiz				* see note 1	[ISOBMFF] 8.7.12		Sample Auxiliary Information Sizes Box
		sbgp				*	[ISOBMFF] 8.9.2		Sample to Group Box
		sgpd				*	[ISOBMFF] 8.9.3	Section 7.5.17	Sample Group Description Box
mdat						+	[ISOBMFF] 8.2.2	Section 7.5.18	Media Data container for media samples

Format Req.: indicates the number of boxes that are required to be present in the file, where:

‘*’ means “zero or more” and

‘+’ means “one or more”

"0/1" indicates only that a box may be present, but it may also be conditionally required as specified in the CMAF format or a specific Profile.

Note 1: The Sample Auxiliary Information Box ('saio') and Sample Auxiliary Size Box ('saiz') are conditionally required for encrypted content as specified in section 8.7.12 of [CENC].

Table 2 is a continuation of Table 1 showing nesting levels 5 to 8 separately to reduce table width.

Table 5 — CMAF Protected Sample Entry Box structure

NL 5	NL 6	NL 7	NL 8	Format Req.	Source	Requirements	Description
stsd				1	Section 7.5.9		Sample Description Box
	sinf			*	[ISOBMFF] 8.12.1		Protection Scheme Information Box
		frma		1	[ISOBMFF] 8.12.2		Original Format Box
		schm		1	[ISOBMFF] 8.12.5		Scheme Type Box
		schi		1	[ISOBMFF] 8.12.6		Scheme Information Box
			tenc	1	[CENC] 8.2		Track Encryption Box

7.3.2 CMAF Track Structure

The following sections define the constraints of the CMAF Tracks.

7.3.3 CMAF Header

Each CMAF Track SHALL have an associated CMAF Header that can be processed to configure CMAF Track parsing, decoding, and display.

A CMAF Header SHALL contain the set of boxes and their sequence shown in

Table 3.

A CMAF header SHALL be conformant with [ISOBMFF], with the following additional constraints and requirements:

- The CMAF Header SHALL start with a File Type Box ('ftyp')
- The CMAF Header SHALL include one Movie Box ('moov').

- The Movie Box SHALL start with a Movie Header Box ('mvhd'), as defined in Section 7.5.1.
- The Movie Box SHALL contain exactly one track containing media data as specified in Section 7.3.4, which defines the Track Box ('trak') requirements for the Common Media Format.

  Note: Timed metadata tracks can be provided as separate CMAF Tracks in a separate Selection Set.
- The Movie Box SHALL contain a Movie Extends Box ('mvex'), as defined in Section 8.8.1 of [ISOBMFF], to indicate that the file contains Movie Fragment Boxes. The Movie Extends Box ('mvex') MAY contain a Movie Extends Header Box ('mehd'), as defined in [ISOBMFF] Section 8.8.2, and if so SHALL provide the overall duration of a fragmented movie. If the duration is unknown, this box SHALL be omitted.
  - If an Edit List Box (`elst') is included in a CMAF Track, the value of entry_count SHALL be 1, and all fields SHALL be set to the values specified in Section 7.5.12.

7.3.4 General Constraints on CMAF Tracks

A CMAF Track SHALL include a CMAF Header and a sequence of CMAF Fragments that are continuous and increasing in decode time.

The CMAF Header SHALL be sufficient to ensure that the processing of any CMAF Fragment in the CMAF Track is performed correctly by a system initialized with the CMAF header.

A CMAF Track SHALL conform to a fragmented ISOBMFF with the exception that the first CMAF Fragment in a CMAF Track MAY have a non-zero baseMediaDecodeTime, and sequence_number in Movie Fragment header Boxes SHALL be ignored.

Subsequent CMAF Fragments in a CMAF Track SHALL have baseMediaDecodeTime equal to the sum of all prior CMAF Fragment durations added to the first Fragment's BaseMediaDecodeTime.

Note: Valid CMAF Tracks do not have media time discontinuities resulting from missing samples or Fragments. Gaps in decode time would result in audio video synchronization errors. For recommendations on handling missing media samples and CMAF Fragments, see Annex G.

Each CMAF Fragment in a CMAF Track SHOULD have a duration of at least one second, with the possible exception of the first and last Fragments of the Track.

Additional general constraints specific to encryption, audio, video, and subtitle CMAF Tracks are in corresponding Sections 8 to 11.

if a CMAF Media Profile specifies CMAF Switching Sets and their constraints, it SHALL specify additional CMAF Track constraints specific to the codec, track format, and Media Profile, necessary for one or more Adaptive Switching Processes. See Sections 9.2.3 for example CMAF Switching Set constraints and Adaptive Switching Processes for NAL Structured Video Media Profiles.

7.3.5 CMAF Fragments

Each CMAF Fragment SHALL consist of one or more ISO Base Media segments [ISOBMFF, section 8.16] that contains one Movie Fragment Box (‘moof') followed by one or more Media Data Boxes ('mdat'). These SHALL conform to the following constraints:

1. A CMAF Fragment SHALL contain one or more Movie Fragment Box ('moof'), and each Movie Fragment Box SHALL be immediately followed by one or more Media Data Box ('mdat') containing the samples it references.
2. The Movie Fragment Box ('moof') SHALL contain a single Movie Fragment Header Box ('mfhd'). See Section 7.5.14 for more detail.
3. The Movie Fragment Box ('moof') SHALL contain a single Track Fragment Box ('traf').

4. Each Track Fragment Box ('traf') SHALL contain one Track Fragment Decode Time Box ('tfdt').
5. A CMAF Fragment SHALL contain those media samples necessary to render the duration of the track fragment box, and no others. For audio and video tracks, each sample SHALL occur exactly once in a CMAF Track. For text and subtitle tracks, the specific CMAF Media Profile defines the permissible duplication of tags at CMAF Fragment boundaries.
6. All media samples in a CMAF Fragment SHALL be addressed by byte offsets in the Track Run Box ('trun') that are relative to the first byte of the Movie Fragment Box ('moof'). (see [ISOBMFF] Section 8.8.4).
7. CMAF Fragments containing encrypted samples SHALL conform to the constraints in Section 8.
8. Each CMAF Fragment, in association with its associated CMAF Header, SHALL contain sufficient metadata to be decoded, decrypted, and displayed when it is independently accessed.

   Note: For instance, if sample groups and sample group descriptions are used to signal optional features, such as encryption key changes, then a Sample Group Description Box ('sgpd'), and Sample to Group Box ('sbgp') that reference that description has to be present in each in each Track Fragment Box ('traf').
9. The CMAF Fragment Movie Fragment Box (‘moof’) MAY be preceded by file level boxes including one or more Event Message Boxes ('emsg'). See Section 7.4.5 and Annex F. for more information on Event Messages.

The following diagram illustrates the box sequence and containment of a CMAF Fragment.

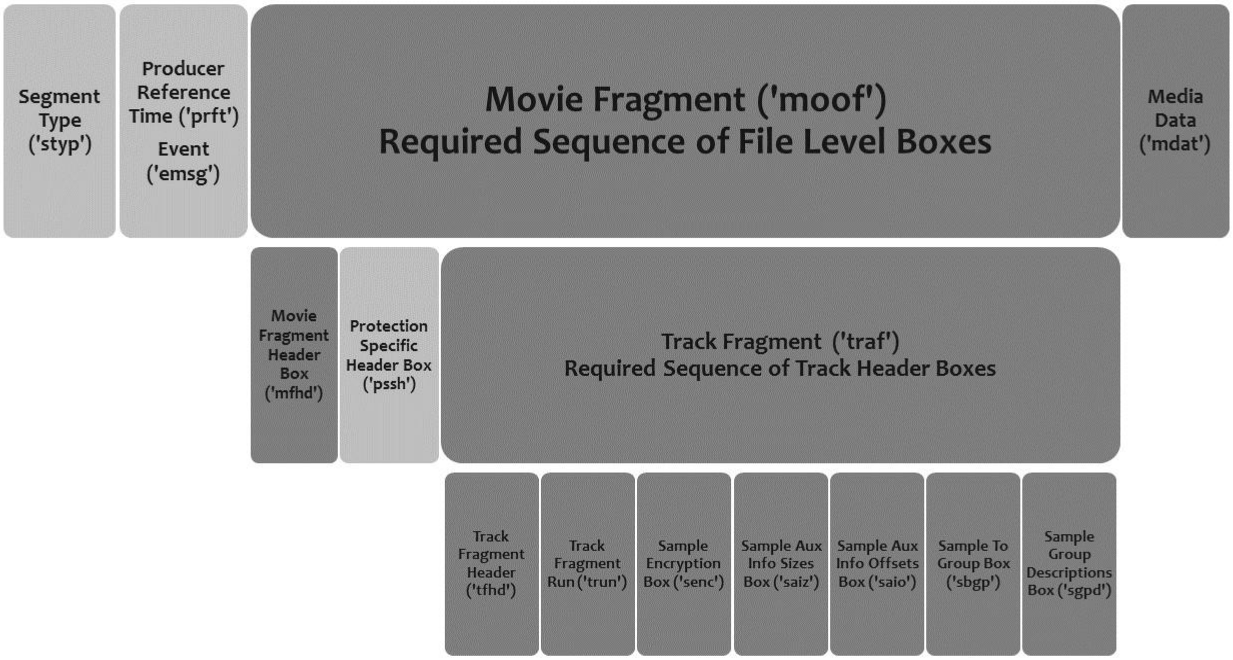

Segment Type ('styp')
Producer Reference Time ('prft')
Event ('emsg')
Movie Fragment ('moof')
Required Sequence of File Level Boxes
Media Data ('mdat')
Movie Fragment Header Box ('mfhd')
Protection Specific Header Box ('pssh')
Track Fragment ('traf')
Required Sequence of Track Header Boxes
Track Fragment Header ('tfhd')
Track Fragment Run ('trun')
Sample Encryption Box ('senc')
Sample Aux Info Sizes Box ('saiz')
Sample Aux Info Offsets Box ('saio')
Sample To Group Box ('sbgp')
Sample Group Descriptions Box ('sgpd')

Figure 13 — (Informational) CMAF Fragment box sequence and containment

NOTE Lower boxes indicate containment in the box above. The sequence of boxes contained in the ‘traf’ box is only recommended. The presence of the Protection Specific Header Box in the Movie Fragment is optional, and typically only used for the delivery of chained licenses for subscriptions, “key rotation” entitlement validation, etc. The Producer Reference Time Box ('prft'), Segment Type Box ('styp'), and Event Message Box ('emsg') are optional.

7.3.6 CMAF Segments

Each CMAF Segment SHALL include one or more complete CMAF Fragments.

A CMAF Segment MAY contain a SegmentTypeBox including the compatible brand 'cmfs', and the compatible_brands in the File Type Box of the CMAF Header.

Each CMAF Resource SHOULD have an associated CMAF Resource Identifier by which it can be addressed by servers, CDNs, and Manifests.

Note: Use of the same URI per CMAF Segment is recommended for Segment identification and efficient reuse in generic content distribution networks and caches, even when different manifests or delivery protocols are used.

A CMAF Segment MAY be identified by a CMAF Player by its CMAF Resource Identifier independent of the delivery system.

Note: For example, CMAF Segments can be delivered over broadcast or multicast, while the same CMAF Segments can be delivered over unicast, but the CMAF Player can identify any CMAF Segment by Resource Identifier in a manifest and schedule it for playback regardless of which network path delivered it.

7.3.7 CMAF Chunks for low latency delivery of media samples

A CMAF Chunk is an ISOBMFF segment related to a CMAF Fragment and CMAF Segment by the following constraints:

A CMAF Chunk –

1. SHALL contain only one ISOBMF segment, as defined in [ISOBMFF]), with one Media Data Box (`'mdat'`).
2. SHALL contain a sequential subset of the media samples of an associated CMAF Segment that contains one CMAF Fragment.

   Note: Each CMAF video Chunk will contain a subset of the samples of a coded video sequence.

3. The CMAF Chunk that corresponds to the beginning of the CMAF Segment shall have the same `BaseMediaDecodeTime` as that segment.
4. For any sample that occurs in both a CMAF Segment and a CMAF Chunk, the computed presentation times SHALL be identical.

A complete CMAF Segment response consisting of CMAF Chunks contains all of the Samples in the requested CMAF Segment in decode order. Seamless Track Switching is only possible on the first CMAF Chunk containing the first samples of the equivalent CMAF Segment.

7.3.8 CMAF Track File

A CMAF Track File SHALL be a single, complete CMAF Track stored in an ISOBMFF file with the first CMAF Fragment BaseMediaDecodeTime equal zero.

Additional boxes, such as Segment Index Boxes (‘`sidx`’), MAY be present between the CMAF Header and the first CMAF Fragment.

If Segment Index boxes exist, each subsegment referenced in the Segment Index Box SHALL be a single CMAF Fragment contained in the CMAF Track File.

If a CMAF Track is stored as a CMAF Track File, CMAF Resource Identifiers SHOULD be a file URL plus a byte range in each HTTP request for a CMAF Segment.

7.3.9 Constraints on CMAF Switching Sets

7.3.9.1 General Constraints on CMAF Switching Sets

A CMAF Switching Set SHALL contain one or more CMAF Tracks, each of which is an encoding of the same source content.

Each CMAF Switching Set SHALL contain one media type, i.e. audio or video or subtitles.

Each Track in a Switching Set SHALL have the same ISOBMFF defined decoding time duration.

Each Track in a Switching Set SHALL have the same number of CMAF Fragments.

CMAF Fragments in a CMAF Switching Set encoded from the same source media samples SHALL start at the same decode time, as specified by the `BaseMediaDecodeTime` in the Track Fragment Decode Time Box (`'tfdt'`).

7.3.9.2 Constraints on CMAF Switching Sets for Single Initialization Adaptive Switching Process

1. Decoding parameters and their indexes SHALL by identical in each CMAF Header, or signaled in each CMAF Fragment.
2. Sample default parameters in CMAF Headers SHALL be identical, or signaled in each CMAF Fragment, e.g. default_sample_duration and dependency flags signaled in the Track Fragment Header Box (`‘tfhd’`) in each CMAF Fragment instead of in the Track Extends Box (`'trex'`) in the CMAF Header.
3. Edit lists SHALL be Identical in each CMAF Header, if present.

In the same CMAF Switching Set, one or more encoding parameters MAY vary as specified by a Media Profile and Adaptive Switching Process, e.g. bit rate, frame rate, and resolution.

In the same CMAF Switching Set, the boxes identified in Table 6 SHALL be identical except for the identified permitted variations, and additional constraints required for a CMAF Switching Set specified by a CMAF Media Profile and Adaptive Switching Process.

Note: Boxes that contain other boxes can have different content, e.g. free-space or other boxes.

Table 6 — General Restrictions on header boxes in switching sets

Box	Switching Set Permitted Variation
`ftyp`	*identical*
`mvhd`	`creation_time`, and `modification_time` MAY differ
`tkhd`	`width`, `height`, `creation_time`, and `modification_time` MAY differ. `track_ID` *identical*
`elst`	May differ if an offset edit list is used in each CMAF Track to remove different composition offset delays, unless constrained to be *identical* by Single Initialization Switching Set constraints in 7.3.9.2

Box	Switching Set Permitted Variation
mdhd	creation_time, and modification_time MAY differ. timescale MAY differ only when CMAF audio Tracks have 2x sampling rates
hdlr	*identical*
elng	*identical*
vmhd	*identical*
smhd	*identical*
sthd	*identical*
dref	*identical*
stsd	Sample entries SHALL use the same codec identifier (four-character code). Visual sample entry constraints for CMAF Switching Sets are defined by each CMAF Media Profile, track format, and Adaptive Switching Process. Visual sample entries MAY indicate different codec profile, level, and constraints; and different encoded width, height, sample aspect ratio, and cropping dimensions, as specified by the CMAF Media Profile and compatibility brand. Audio sample entry constraints for CMAF Switching Sets can optionally be defined by each CMAF Media Profile. Text and Subtitle CMAF Switching Sets are not defined in current CMAF Media Profiles.
stts	*identical*
stsc	*identical*
stsz / stz2	*identical*
stco	*identical*
sgpd	*identical*
kind	*identical*
mehd	*identical*
trex	*identical*
pssh	*identical*

CMAF Tracks in a CMAF Switching Set MAY have different frame rates, but the number of samples per Segment and their duration SHALL differ inversely so that alternative Segments have equal duration.

7.3.10 General Constraints and Requirements for Video Switching Sets

CMAF video Switching Sets MAY contain multiple CMAF Tracks, and SHALL be constrained by CMAF video Media Profile specifications to allow players to seamlessly switch between tracks using one or more specified Adaptive Switching Processes.

An Adaptive Switching Process SHALL specify the necessary Switching Process, such as when CMAF Headers need to be processed, what video scaling or audio mixing is required, what switch points are allowed, etc. to result in a seamless switch. The Adaptive Switching Process depends on the codec and track format constraints also specified by the CMAF Media Profile.

Video Media Profiles specified in Annex A.2 with reference to video track formats specified in Section 9 define two Adaptive Switching Processes, both constrained to allow seamless switching between CMAF Fragments on a single video decoder. Adaptive Switching can be done on CMAF Fragment boundaries and the resulting sequence of CMAF Fragments can be seamlessly decoded using widely supported ISOBMFF parsers and decoders in playback environments such as Web browsers. The Adaptive Switching Process requires cropping and scaling each CMAF video Fragment to the same display aperture, the same refresh rate, and the same presentation timeline.

7.3.11 CMAF Selection Sets

All CMAF Switching Sets within a CMAF Selection Set SHALL be of the same media type, e.g. audio, video, metadata, or subtitles.

All Switching Sets within a Selection Set SHALL be of approximately the same duration, and the difference less than the maximum duration of a Coded Video Sequence.

A Selection Set may contain only one Switching Set if no alternative content is available.

Different Switching Sets that encode the same content with different codecs or video formats may be contained within a Selection Set to enable a player to preselect the most compatible codec or video format for that content, and automatically adaptively switch within the selected Switching Set

One CMAF Track SHOULD be presented from each Selection Set in a Presentation, e.g. one audio and one video Selection Set.

Note: A Subtitle Track can be selected, but not displayed by user control, and “forced” titles may be displayed even through regular subtitles are not displayed.

7.3.12 CMAF Presentations

A CMAF Presentation SHALL contain one or more synchronized Selection Sets, each Selection Set potentially synchronizing a different media type (audio, video, or subtitles).

A CMAF Presentation SHALL conform to section A.1.

Each CMAF Track in a Presentation SHALL be synchronized to the same presentation start time equal to zero on the CMAF Presentation Timeline, and have approximately the same Track duration, within a tolerance of the longest CMAF Segment duration of any Track in a Selection Set. CMAF Tracks of different media types in a synchronized multimedia CMAF Presentation that have synchronized media samples in the source content SHALL have presentation time synchronized media samples in encoded CMAF Fragments.

7.4 Additional Boxes, not defined in the ISO Base Media File Format

7.4.1 Track Encryption Box (`‘tenc’`)

The Track Encryption Box (‘tenc’) specified in [CENC] 8.2 indicates that samples in the track might be encrypted using Common Encryption. The Track Encryption Box contains parameters and default values that apply to an entire track. See Section 8.2.2.2 for additional information.

7.4.2 Sample Encryption Box ('senc')

A Sample Encryption Box ('senc') specified in [CENC], or a UUID box in the Movie Fragment Box ('moof') or a box within it, SHALL be present in each CMAF Segment that stores Sample Auxiliary Information for Common Encryption (e.g. subsample information for NAL structured video, or per-sample initialization vectors).

The information in the 'saio' and 'saiz' byte offsets locating Sample Auxiliary Information SHALL be correct, even when it is stored in 'senc', in order to conform to [CENC].

NOTE Common Encryption allows storage of Sample Auxiliary Information in any location, but content conformance testing and player parsing for CMAF are optimized by constraining the location in each CMAF Segment so that it is easily accessible to the file parser.

7.4.3 Protection System Specific Header Box (‵pssh′)

The Protection System Specific Header Box is specified in the Common Encryption Specification for the storage of content protection system information such as license acquisition information and DRM licenses. Protection System Specific Header Boxes are primarily designed to store information in download files. CMAF streaming applications SHOULD signal license acquisition information in the Manifest, and SHOULD NOT duplicate the information in Protection System Specific Header Boxes in CMAF Headers. See Section 8 for more information.

7.4.4 Media Profile Specific Boxes

Audio and video track formats typically derive sample entries and define decoder configuration boxes that can be required by one or more Media Profiles. For example, the 'avcC' and 'hvcC' boxes are defined in MPEG-4 Part 15 and required by the NAL Structured Video Track Format specified in Section 9 and referenced by the CMAF video Media Profiles specified in Annex A. Other CMAF Media Profiles MAY specify boxes that are added to the ‵cmfc′ file by the Media Profile brand.

7.4.5 Event Message Box ('emsg')

The Event Message Box ('emsg') is specified in section 5.10.3.3 of [DASH].

One or more Event Message Boxes MAY be present in a CMAF Fragment, and if present SHALL precede the first Movie Fragment Box in the CMAF Fragment.

Event Message boxes in a CMAF Track SHALL have a value in the timescale field equal to the value of the timescale field in the Media Header Box (‘mdhd’) of that track.

DASH defines the timing of an Event Message relative to the earliest sample presentation time of a DASH Segment using the field presentation_time_delta, which “provides the Media Presentation time delta of the media presentation time of the event and the earliest presentation time in this segment.”

For CMAF Fragments, the presentation_time_delta SHALL equal the media presentation time of the event minus the earliest presentation time of the following CMAF Fragment.

In the case where a CMAF Track is defragmented (during processing, etc.), all Event Message Boxes SHALL be stored prior to the Movie Box, and the presentation_time_delta SHALL equal the media presentation time of each Event Message.

Note: The earliest media presentation time in a CMAF Fragment is equal to the earliest sample composition time plus an edit offset, if an edit list is present in the track. The earliest decode time of a defragmented ISOBMFF track is zero.

See Annex F. for more information on the use of event messages.

7.5 Constraints on ISO Base Media File Format Boxes

7.5.1 Movie Header Box (`'mvhd'`)

The value of the duration field SHALL be set to zero to indicate that the Movie Box (`'moov'`) contains no media samples and therefore has no duration.

NOTE The `duration` field in the Media Header Box (`'mdhd'`) applies to the Track Box (`'trak'`), which contains no media samples in CMAF. The duration of an entire fragmented movie can optionally be stored in the `fragment_duration` field of the Movie Extends Header Box (`'mehd'`), which is equal to the sum of all track Fragment durations in the longest track in the movie. If the duration is unknown, this box is omitted.

The fields `rate`, `volume`, and `matrix` SHALL be set to their default values.

7.5.2 Metadata

Metadata, carried in either user data or metadata boxes MAY be present. When present they SHALL NOT occur at file level, i.e. they can only be contained in another box, as permitted by [ISOBMFF].

7.5.3 Kind Box (`'kind'`)

The Kind Box (`'kind'`) MAY be used to store the role of a CMAF Track. The Kind Box (`'kind'`) box is stored in the User Data Box (`'udta'`) of the Track Box (`'trak'`), as documented in the ISO Base Media File Format [ISOBMFF].

Any track can be labeled with role information describing the intended purpose of the track. This information can be captured at the time of encoding, and later copied to a Manifest describing the CMAF Tracks in a Selection Set so that a user or an automatic algorithm can make an appropriate selection.

The Kind Box (`'kind'`) can contain one or more tags from a variety of places, including:

- the DASH specification [DASH]section 5.8.5.5, as identified by the schemeURI "`urn:mpeg:dash:role:2011`" (without the quotation marks);

- The W3C HTML5 specification of track 'kind', as identified by the schemeURI [[ED — ? https://www.w3.org/TR/html5/embedded-content-0.html#the-track-element]]

Where multiple schemes define the same concepts, the DASH scheme SHOULD be used. In particular, where captions or descriptions need to be identified, or that the text be marked as easy to read, the following values from DASH SHOULD be used:

- "caption"
- "description"
- [[ED: "public.easy-to-read" — no matching DASH value? ATSC3?]]

7.5.4 Track Header Box ('tkhd')

CMAF Track Header Boxes SHALL conform to [ISOBMFF] Section 8.3.1 with the following additional constraints:

- The field `duration` SHALL be set to a value of zero ('0'), indicating no media samples are present in the Track Box ('trak').
- The field `matrix` SHALL be set to their default values as defined in [ISOBMFF], except to indicate video orientation (i.e. portrait or landscape orientation relative to the captured scene).
- The following fields SHALL be set to default values as defined in [ISOBMFF], unless specified otherwise in this specification:

  The `layer` field SHOULD equal 0 for video tracks and `-1` for subtitle tracks (i.e. in front of the video).

- The `width` and `height` fields for a non-visual track (e.g. audio) SHALL be 0.
- The `width` and `height` fields for a CMAF video track SHALL specify the track's normalized presentation size as fixed-point 16.16 values expressed in square pixels after decoder cropping, and in the case of video encoded with a non-square spatial sample shape, after horizontal scaling has been applied. See Section 9.2.4.1 for normalized `width` and `height` calculation.

Note: Normalized `width` and `height` are primarily useful to determine the picture aspect ratio, and for device selection of Tracks that approximate a player's display aperture size, when bandwidth and decoding capacity allow. Adaptively switched Segments can be scaled to a device determined display aperture by applying scaling ratios equal to the display aperture's width and height in square pixels, divided by the Segment's decoded and cropped horizontal and vertical spatial sample counts. The spatial sample counts can be derived from the Track's visual sample entry for `'avc1'` or `'hvc1'` video samples (see Section 9.2.4.2), or from a Sequence Parameter Set NAL in each Segment and Coded Video Sequence for `'avc3'` or `'hev1'` video samples. See Section 9.2.4.3 for the storage and semantics of video Sequence Parameter Sets.

- Subtitle Tracks MAY set `width` and `height` to an intended layout size, in which case the text layout engine or graphics engine can scale the width and height to match the video display aperture (player implementation dependent).
- As defined in ISO/IEC 14496-30 [ISOTXT], Subtitle Tracks encoded as text MAY use relative position coordinates and font sizes so that the text layout engine can adjust glyph and layout size to match the final video display aperture without relying on image scaling. For such tracks, the value of zero width and height SHOULD be used to indicate that the data can be rendered at any size, and the layout size may be determined by matching the size of the video display aperture.
- For scalable text and subtitle tracks, the flag `track_size_is_aspect_ratio` may also be used.
- The `track_size_is_aspect_ratio` flag indicates that the width and height fields are not expressed in pixel units, but indicate the intended aspect ratio. If the aspect ratios of this track and related video tracks are not identical, then the respective positioning of the tracks is undefined, possibly defined by external context. This flag value is `0x000008`.

7.5.5 Media Header Box ('mdhd')

The CMAF Media Header Boxes SHALL conform to [ISOBMFF] Section 8.4.2 with the following additional constraints:

- The value of the `duration` field SHALL be set to a value of zero ('0');

  NOTE The `duration` field in the Media Header Box ('mdhd') applies to the Track Box ('trak'), which contains no media samples in CMAF. The duration of an entire fragmented Track can optionally be stored in the `fragment_duration` field of the Movie Extends Header Box ('mehd'), which is equal to the sum of all track Fragment durations.

- Where possible, the value of the `timescale` field SHOULD be chosen such that when the frame rate is constant, the value of the sample duration may also be constant.
- All tracks that are language-specific SHOULD identify the language as precisely as possible (e.g. a text track whose language can be written in different scripts should identify which script is used.). When the language is not relevant or not known, the 'und' (undetermined) language tag SHOULD be used.

7.5.6 Video Media Header ('vmhd')

Video Media Header Boxes in a CMAF SHALL conform to [ISOBMFF] Section 8.4.5 with the following additional constraints:

- The following fields SHALL be set to their default values as defined in [ISOBMFF] Section 8.4.5:
  - `version=0`
  - `graphicsmode=0`
  - `opcolor={0, 0, 0}`

7.5.7 Sound Media Header ('smhd')

The balance value in the sound media header SHOULD be zero (centered).

7.5.8 Data Reference Box (`'dref'`)

Data Reference Boxes in a CMAF Track SHALL conform to [ISOBMFF] Section 8.7.2 with the following additional constraints:

- The Data Reference Box (`'dref'`) SHALL contain a single entry with the `entry_flags` field set to `0x000001` (which means that the media data is in the same file as the Movie Box containing this data reference).

7.5.9 Sample Description Box (`'stsd'`)

Sample Description Boxes in a CMAF Track SHALL conform to version 0 as defined in [ISOBMFF] Section 8.5.2 with the following additional constraints:

- Sample entries for encrypted tracks (those containing any encrypted sample data) SHALL encapsulate the existing sample entry with a Protection Scheme Information Box (`'sinf'`) that conforms to [ISOBMFF] section 8.12.1.
- For video tracks, a visual sample entry SHALL be used. Design rules are specified in Section 9.2.4.2.
- For audio tracks, an audio Sample entry SHALL be used. Design rules are specified in Section 10.3.6.
- For subtitle tracks a subtitle sample entry SHALL be used. Design rules are specified in [ISOTXT].

7.5.10 Protection Scheme Information Box (`'sinf'`)

CMAF SHALL use Common Encryption for Tracks containing one or more encrypted CMAF Segments as defined in [CENC], and use Scheme Signaling as defined in [CENC] Section 4. An encrypted CMAF Track SHALL include at least one Protection Scheme Information Box (`'sinf'`) identifying a scheme specified in [CENC] Section 10.

7.5.11 Track Sample Boxes

All boxes in the SampleTableBox (`'stbl'`) SHALL have or compute a sample count of 0, because CMAF does not reference media samples from the Track Box (`'trak'`). For example, the following boxes therefore have an entry_count of zero:

- Decoding Time to Sample Box (`'stts'`)
- Sample to Chunk Box (`'stsc'`)
- Chunk Offset Box (`'stco'`)
- Sample Size Boxes (`'stsz'` or `'stz2'`)
- Sync Sample Box ('sync')

The presence of an empty Sync Sample Box in video or other CMAF Header indicates that not all samples in the CMAF Track are sync samples.

The `sample_size` field of the `'stsz'` box SHALL be set to zero ('0'). (Sample size and duration information can be found in the Track Fragment Run Box (`'trun'`) in each CMAF Segment.)

The mandatory boxes of ISO/IEC 14496-12 are mandatory, even though they document no samples.

7.5.12 Track Edit List Box (`'elst'`)

A single Edit List Box with the following constraints MAY be present the CMF Header of a CMAF Track to adjust the earliest sample presented at CMAF Presentation Timeline zero, and its track presentation time. In this case, the edit SHALL conform to [ISOBMFF] section 8.6.6, which specifies a "non-empty edit" that "provides the offset from media composition time to movie presentation time". The term "movie presentation time" is equivalent to "track presentation time", and refers to sample presentation time after the application of edit list and composition offsets.

A single Edit List Box with the following constraints SHOULD be recorded in the CMF Header to document the earliest presented sample's track presentation time in an audio CMAF Track when the composition time of the first sample in the CMAF Fragment does not equal zero on the CMAF Presentation Timeline.

An offset edit list SHOULD be recorded in a CMAF header when encoding an audio track that contains a CMAF Fragment that overlaps the presentation start time of the first video sample, in order to skip audio that precedes the first video sample at the start of CMAF Presentation playback.

A start offset edit list can also be used to skip presentation of leading audio that is included to "prime" the decoder for predictive audio codecs, such as AAC, so the decoder can output the intended audio sample at the intended presentation start time by decoding and discarding the extra audio.

Each time a CMAF Header is processed, any offset edit value present SHALL be applied to the composition time of each sample in the CMAF Track to compute the track presentation time and synchronize all CMAF Tracks to a common CMAF Presentation Timeline.

Audio/video synchronization and start time can be expressed as a presentation time offset in a Manifest relative to the track presentation time. The track presentation time of each sample can be computed as:

track presentation time = `sample decode time + sample composition offset – Media-Time`

A start offset edit list SHALL be defined as a single Edit List Box (`'elst'`) in an Edit Box (`'edts'`) in a Track Box (`'trak'`) with the following values:

- `Segment-duration = 0`
- `Media-Time` = offset from the start of the first Fragment measured in the Track `timescale`
- `Media-Rate = 1`

As documented in [MP4FILE] an audio track compressed with AAC SHOULD have a pre-roll sample group ('roll'), and an edit list to omit output of the priming audio data and other audio data present in the track prior to the intended presentation start time.

7.5.13 Track Extends Box ('trex')

Track Extends Boxes ('trex') SHALL be present in a CMAF Track since it is a fragmented file as defined in [ISOBMFF] Section 8.8.3.

7.5.14 Movie Fragment Header Box ('mfhd')

Movie Fragment Header Boxes ('mfhd') in a CMAF Track SHALL conform to [ISOBMFF] Section 8.8.5.

Note: The sequence_number integer value is not required to be unique within a CMAF Track nor to increase with decode time.

7.5.15 Track Fragment Header Box ('tfhd')

Track Fragment Header Boxes ('tfhd') in a CMAF Track SHALL conform to [ISOBMFF] Section 8.8.7 with the following additional constraints:

- The track_ID field SHALL contain the same value as the track_ID in the matching CMAF Header
- the base-data-offset-present flag (in the tf_flags field) SHALL be set to zero in order to indicate that media samples are addressed using byte offsets relative to the the Movie Fragment Box ('moof'); and
- the default-base-is-moof flag (in the tf_flags field) SHALL be set to one in order to indicate that the data_offset field in the Track Fragment Run Box ('trun') is always calculated relative to the first byte of the enclosing Movie Fragment Box ('moof').

- Every Track Fragment Box ('traf') SHALL contain a Track Fragment Decode Time Box ('tfdt'), as defined in [ISOBMFF] Section 8.8.12, to provide the first Sample decode time in the Fragment.
- In CMAF Segments, the field baseMediaDecodeTime SHALL also equal the first Sample composition time in the Fragment.

The baseMediaDecodeTime of the first available CMAF Segment in a CMAF Track MAY be non-zero.

Note: A CMAF Track may be thought of as a portion of a hypothetical ISO Media track. In the event that the baseMediaDecodeTime of each CMAF Segment is set to its NTP encode time, for example, that would imply it was a portion of an ISO Media track that began January 1, 1900. For CMAF, 'tfdt' baseMediaDecodeTime can be considered an arbitrary timeline for CMAF Tracks, which follows ISO Media decode time rules within the available Segments, i.e. decode time is the sum of prior sample durations in the track, and 'tfdt' contains the decode time of the first sample in each Segment.

7.5.16 Track Fragment Run Box ('trun')

Track Fragment Run Boxes ('trun') in a CMAF Track SHALL conform to [ISOBMFF] Section 8.8.8 with the following additional constraints:

- IF the version field is set to '1', the sample_composition_time_offset is a signed 32-bit integer measured by the timescale of the track, which reorders video Samples from decode order to presentation order. The first presented Sample in a CMAF Segment SHALL be offset to coincide with the first Sample decode time (BaseMediaDecodeTime in 'tfdt'), and subsequent Samples follow continuously in presentation order.
- the data-offset-present flag (in the tf_flags field) SHALL be set to true in order to indicate that the data_offset field is present and contains the byte offset from the start of this Fragment's Movie Fragment Box ('moof') to the first

sample of media data in the following Media Data Box ('mdat'). Note, this is called movie-fragment relative addressing in [ISOBMFF].

- Within a video CMAF Track, any Track Run Box ('trun') that describes any non-sync pictures SHALL identify pictures using a combination of the sample_flags and first_sample_flags fields:
  - sample_is_non_sync_sample SHALL be 0 for SAP type 1 or 2, and 1 if not;
  - sample_depends_on SHOULD be 2 for I pictures;
  - sample_is_depended_on SHOULD be 2 for disposable pictures.

Note: At present, there is no generally supported semantic to represent "missing" media in a track. The Track Run Box may assign a single image sample the duration of the track run to fill the time interval. Audio samples have internal time structure, so a single sample's duration cannot be extended. See Annex G.

7.5.17 Sample Group Description Box ('sgpd')

When sample group information can change within a CMAF Track, a Sample Group Description Box SHALL be stored in each CMAF Fragment that references that sample group description. If sample group information is the same for all Fragments in a Switching Set, it MAY be stored in a Sample Group Description Box in the CMAF Header Sample Table Box ('stbl').

For example: when Common Encryption is used and KID values can change per CMAF Fragment, a Sample to Group Box ('sbgp') stored in each Track Fragment Box ('traf') will reference a Sample Group Description Box containing the KID, which is also stored in the Track Fragment Box in order to support random access.

Pre-roll sample groups are used for some Audio as defined in 7.5.12.

7.5.18 Media Data Box ('mdat').

Each CMAF Fragment SHALL contain one or more Media Data Box(es) ('mdat') containing media Samples. The Media Data Box conforms to the definition in [ISOBMFF] Section 8.1.1 with the following additional constraints:

- The only media Samples in each instance of this box SHALL be those referenced by the single Track Fragment Box that precedes it (i.e. only audio, video, or subtitles from the track fragment time interval of one track). In other words, all samples within an instance of this box belong to the same CMAF Track and Fragment.

8 Common Encryption of Tracks

8.1 Multiple DRM Support (Informative)

Multiple DRM systems can provide a license for an encrypted Track using Common Encryption [CENC]. The default_KID identifies the key and license required, and a registered SystemID identifies a Common Encryption capable DRM system. License acquisition information can be provided to identify which DRM systems can provide licenses. License acquisition information usually includes the URL of an authorization and license server, DRM SystemID, DRM client identification, type of license requested, media key identifier, etc., and may be stored in a 'pssh' box, a Manifest, or an application in order to assist players in requesting a license to decrypt a Track.

A single key and license may be sufficient to access all tracks in a presentation, or HD and other high value content may need different keys for audio and video tracks, since the audio path may be less secure. A content provider may also use different keys and licenses for different qualities, such as SD, HD, and UHD.

When streaming, it is recommended that any license acquisition information used to acquire CMAF decrypt key(s) be signaled in a Manifest or application and not in a Protection System Specific Header Box ('pssh') in the CMAF header. This will enable a

CMAF Player application to parse license acquisition information in advance of playback and download the required license(s). License acquisition information SHOULD NOT be duplicated in a 'pssh' version 0 box in the CMAF header since this may trigger licensing events that require player handling each time a CMAF Header is parsed, e.g. an HTML5 browser using media source extensions and the encrypted media extension interfaces.

Manifest signaling makes it easier to add or change license information without editing media files. That makes it easier to offer different types of licenses, e.g. subscription, rental, ownership, SD, HD, etc., without multiple media copies; and support different distribution channels and license servers with the same media. For a live streaming presentation, it is advantageous for players to request a license before the live media becomes available, authorize or purchase playback rights, and download a license in advance, rather than experience playback delay when thousands of players receive the first live Segment, which would be the case with license acquisition information stored in 'pssh' in the CMAF header.

8.2 Track Encryption

8.2.1 Encryption Overview

Encrypted track sample data in a CMAF SHALL use an encryption scheme defined in [CENC] Section 4.2. Encrypted NAL Structured Video tracks SHALL follow a Subsample encryption scheme outlined in [CENC] Section 9.5, which defines a NAL unit partial encryption scheme to allow access to NALs and unencrypted video NAL headers in an encrypted NAL Structured Video elementary stream.

The requirement to include in the Track Fragment Box ('traf') a Sample Auxiliary Information Offsets Box ('saio') and a Sample Auxiliary Information Sizes Box ('saiz'), both with an explicit or implied aux_info_type value of 'cenc' is defined in [CENC] and [ISOBMFF].

Initialization vectors, subsample byte ranges and other Sample Auxiliary Information for Common Encryption SHALL be stored in the Sample Encryption Box (`'senc'`) or equivalent UUID box in the `'moof'`.

All encrypted non-video tracks SHALL follow the schemes outlined in [CENC] Section 9.4 or 9.7, which defines full sample encryption schemes.

The following additional constraints SHALL be applied to all encrypted tracks:

— All key identifier values SHALL be 16 byte values and SHALL uniquely identify one and only one key within their scope of use. To ensure this level of uniqueness, it is strongly recommended that the key identifier values be a UUID generated according to [X667]. A UUID SHALL be stored in the KID field as 16 octets, as specified in section 6.2 of [X667].

— A `KID` value MAY be represented in text as a hyphenated hexadecimal string, as specified in section 6.4 of [X667].

8.2.2 Track Constraints

8.2.2.1 Sample Encryption Box (`'senc'`) and Sample Auxiliary Information

For encrypted track Fragments that contain Sample Auxiliary Information, the Track Fragment Box (`'traf'`) SHALL contain a Sample Auxiliary Information Offsets Box (`'saio'`) with an `aux_info_type` value of `'cenc'` as defined in [CENC] Section 7 to provide sample-specific encryption data. This Sample Auxiliary Information Offsets Box (`'saio'`) SHALL conform to the following three constraints:

1. The `offset` field SHALL point to the first byte of the first initialization vector in the Sample Encryption Box (`'senc'`) or a UUID box containing Sample Auxiliary Information;
2. The data in the Sample Encryption Box (`'senc'`) or equivalent UUID box SHALL be contiguous for all of the samples in the movie Fragment (the `CencSampleAuxiliaryDataFormat` structure has the same format as the data in the

Sample Encryption Box ('senc'), by design) and so the entry_count field of the Sample Auxiliary Information Offsets Box (‘saio’) SHALL be 1;

3. The offset field of the entry SHALL be calculated as the difference between the first byte of the containing Movie Fragment Box ('moof') and the first byte of the first InitializationVector in the Sample Auxiliary Information (using movie Fragment relative addressing where no base data offset is provided in the track Fragment header).

The size of this sample auxiliary data SHALL be specified in a Sample Auxiliary Information Sizes Box ('saiz') with an aux_info_type value of 'cenc', as defined in [CENC] Section 7, including the following exceptions:

If Subsample encryption is not used (the size of the sample auxiliary information equals default_Per_Sample_IV_Size in the 'tenc' box), and the entire sample is protected (see [CENC] 9.4 for further details). In this case, all auxiliary information will have the same size and hence the default_sample_info_size of the Sample Auxiliary Information Sizes box ('saiz') will be equal to the default_Per_Sample_IV_Size of the Initialization Vectors, and 'saiz' box MAY be omitted.

If Per_Sample_IV_Size is also zero (because constant IVs are in use) then the sample auxiliary information would then be empty and SHOULD be omitted, as well as the 'saio' box.

Even if Subsample encryption is used, the size of the sample auxiliary information may be the same for all of the samples (if all of the samples have the same number of Subsamples) and the default_sample_info_size may be used.

The Sample Auxiliary Information Sizes Box ('saiz') SHALL conform to the following two constraints:

1. the sample_count field SHALL match the sample_count in the Sample Encryption Box ('senc');

2. the `default_sample_info_size` SHALL be zero ('0') if the size of the per-sample information is not the same for all of the samples in the Sample Encryption Box ('senc').

   Note: Sample encryption information, is located by Fragment byte offsets stored in the Sample Auxiliary Information Offsets Box ('saio'), and in some cases by size information stored in the Sample Auxiliary Information Size Box ('saiz'). Sample Auxiliary Information, such as per-sample initialization vectors and subsample byte ranges is not intended to be read directly from the Sample Encryption Box ('senc') or an equivalent UUID Box. This specification recommends storage in a Sample Encryption Box ('senc') in each movie Fragment for consistency, parsing efficiency, and conformance testing, but the Sample Auxiliary Information Offsets Box ('saio') can be used to locate auxiliary information that may have been refragmented or stored in other boxes in the Track Fragment Box ('traf').

8.2.2.2 Track Encryption Box ('tenc')

As specified in [CENC], a Track Encryption Box indicates that media samples in the track might be encrypted, it identifies the encryption scheme used, and contains default encryption parameters for the ISO Media track.

8.2.2.3 Protection System Specific Header Box ('pssh')

Common Encryption specifies version zero and version one Protection System Specific Header Boxes ('pssh'). Protection System Specific Header Boxes can be used to store licenses in downloaded files, signal in CMF Headers that license downloads are required, and deliver licenses, keys, and usage information in CMF Fragments. Protection System Specific Header Boxes contain a registered SystemID that uniquely identifies the protection system intended to use the information. Information contained in the data[] array is considered opaque to Players, file parsers, and other DRM systems, and might be encrypted by the protection system.

The Common Encryption standard [CENC] also specifies XML elements to contain license acquisition information for use in Manifests. CMAF strongly recommends signaling license acquisition information only in the Manifest, not in CMAF Headers, to avoid triggering event handling each time a CMAF Header is processed when adaptive switching in some players. A Player can download all licenses that will be needed for playback as soon as it parses the Manifest and before it downloads CMAF Headers. It is particularly useful to acquire licenses in advance of a large live streaming event that would otherwise result in a large number of synchronized license requests if triggered by the media stream.

CMAF specifies the following constraints:

- `'pssh'` boxes SHOULD NOT be present in CMAF Headers, except as specified below, and MAY be ignored if present.
- A Common `'pssh'` box and additional `'pssh'` boxes in CMF Header MAY be used with Presentation-specific playback applications that can read the KID from the Common `'pssh'` and implement application-specific license management. A Common PSSH is defined by W3C Encrypted Media Extension specification as a version one `'pssh'` box with a W3C designated SystemID, and the track's default_KID listed in the version one `'pssh'` box KID array, but it contains no data in the data[] array.
- `'pssh'` version one boxes MAY be present in CMAF Fragments for the purpose of providing protection system information, such as encrypted keys in licenses, for use by the DRM system with a SystemID matching the `'pssh'` SystemID. Multiple `'pssh'` boxes with different SystemIDs may be present to enable different protection systems on different devices.

  For example, version one `'pssh'` boxes in CMAF Fragments can be used to deliver the same encrypted licenses to all Players by a channel or subscription service, but only users who have purchased and downloaded an entitlement license bound to their particular device or account can decrypt the licenses in the CMAF Fragments needed to decrypt the Fragment. Since one license decrypts the other, they are said to be “chained”, and chained licenses in combination with periodic key changes can verify that each viewer is authorized by a valid account-bound entitlement license (this process is often called “key rotation”).

8.2.3 Encryption Constraints

8.2.3.1 General

- For a given KID, Initialization Vectors and counter values SHALL be used only once and SHALL follow the guidelines outlined in [CENC] Section 9.2 and 9.3.
- Initialization Vectors for used with the 'cenc' scheme SHALL be limited to 8-bytes as defined in [CENC] section 9.2 to avoid block counter value overlap.
- Each KID and default_KID SHALL never reference more than one key value for all CMAF Tracks. Therefore each KID SHALL be generated by the UUID algorithm specified in UUID as specified in [X667]. Note that it is possible for two different KIDs to reference the same key value.
- Default_KID and key values SHALL be applied to CMAF Switching Sets such that the default_KID identifies a license or key sufficient to enable an authorized DRM system to decrypt the CMAF Switching Set. Any additional keys described by sample groups MAY be delivered in version 1 Protection System Specific Header Boxes ('pssh') in CMAF Segments, identifying the contained KID(s), and protected by DRM specific methods.
- Textual representation of KID and SystemID in Manifests SHOULD use the hexadecimal string representation specified in [X667] section 6.4, derived from the 16 octet binary representation specified in section 6.2 and equivalent byte arrays specified in [CENC] boxes.

Subtitle tracks SHALL NOT be encrypted.

The following additional constraints SHALL be applied to the encryption of NAL Structured Video tracks:

- Slice headers, NAL type headers, NAL size headers, and all Non-video NALs SHALL be unencrypted.
- VCL data SHALL be protected, either by full encryption or pattern encryption of the VCL slice data.
- `'cenc'` scheme `bytesOfProtectedData` SHALL be a multiple of 16 bytes.
- `'cbcs'` scheme `bytesOfProtectedData` SHALL start on the first complete byte of video data following the slice header, and `BytesOfProtectedData` = size of video slice data starting from the first complete byte of video data following the slice header

8.2.3.2 Clear Samples within an Encrypted Track

In an encrypted track, the isProtected flag in the Track Encryption Box (`'tenc'`) SHALL be set to 1, indicating that all samples are protected by default. Sample Groups may indicate unprotected Samples, as specified in [CENC].

Each CMAF Fragment SHALL be constrained to Samples that are all protected or all unprotected, not a mix.

9 CMAF Video Tracks

9.1 Introduction

CMAF Video Tracks conform to the general CMAF Track constraints specified in section 7 and 8, and define additional requirements that video Media Profiles SHALL specify, including video codec parameters, elementary stream format, and access units, and how those are packaged in CMAF samples, Headers, Fragments, Tracks and Switching Sets.

Media Profiles specify CMAF video Tracks, Switching Set constraints, and Adaptive Switching Processes.

Section 9 specifies

- general video Switching Set and Adaptive Switching Process requirements and constraints
- specific constraints for NAL Structured Video that are referenced by Media Profiles defined in Annex A. and Annex B.
- a specific CMAF Track format for AVC video referenced by Video Media Profiles defined in Annex A.2 and required by the Presentation Profile specified in Annex A.1.2.

CMAF video Tracks conform to ISO Media files containing a single ISO Media video track with all samples stored in movie fragments as specified in [ISOBMFF] and [ISOVIDEO], with additional constraints specified in this section.

Constraints on Switching Sets of multiple CMAF video Tracks that enable seamless adaptive switching are specified in Sections 9.2.2, 9.2.3, and CMAF Video Media Profiles specified in Annex A. to signal widely used operating points, such as SD, HD, and UHD, which further constrain codec profiles, levels, resolutions, framerates, bit depth, electro-optical transfer function, color subsampling, etc. All CMAF video Media Profiles specified in Annex A.2 and are encoded as frames i.e. “progressive scan”.

For additional information see Annex A.

9.2 Video Tracks

9.2.1 General Requirements for CMAF Video Tracks

CMAF Media Profiles for video SHALL specify or reference a track format and sample entry specification, such as MPEG-4 Part 15, “Carriage of NAL unit structured video in the ISO Base Media File Format” [ISOVIDEO]

CMAF Media Profiles for video SHALL specify additional constraints for CMAF video Tracks necessary for adaptive streaming, such as constraints on the location of decoding, decryption, and display parameters, and if they are allowed to change between CMAF Fragments.

CMAF Media Profiles for video SHALL specify the format of video samples, sample dependencies and randomly accessible sequences, and how those sequences are stored within a CMAF Fragment.

CMAF Media Profiles for video SHALL specify any additional Codec features, such as layering and scaling of video spatial, temporal, resolution, bit depth, dynamic range, and quality characteristics enabled by a specific codec and Media Profile.

CMAF Media Profiles for video SHALL specify the playback initialization process, including linkage to other CMAF Tracks containing related layers to be decoded, selection and initialization of substreams, layers, presents, etc. for initialization and decoding.

Video tracks SHALL either:

a) contain a version 1 Track Run Box (`'trun'`) in CMAF video Fragments with composition offsets (negative composition offsets where necessary) to adjust the earliest sample presentation time to equal the earliest sample decode time stored in the `baseMediaDecodeTim`e field of the Track Fragment Decode Time Box (`'tfdt'`).

b) or include an offset edit list in the associated CMAF Header to subtract the composition delay added by positive composition offsets in version 0 Track Run Box (`‘trun’`).

A video track SHALL NOT use both signed composition offsets and an edit list box.

9.2.2 General Requirements for CMAF Video Switching Sets and Adaptive Switching Process

All video CMAF Fragments within a Switching Set are intended to be displayed with the same height, width and position; and SHALL be encoded with the same aspect ratio, framing (position and cropping within coded blocks), transfer function, bit depth, color subsampling, color volume, and presentation timing so that switching between CMAF Tracks at allowed points in a Switching Set will result in continuous appearance when the CMAF Segments are scaled to the same device determined display aperture.

Media Profiles MAY specify constraints on Switching Sets between multiple CMAF Tracks, and if so SHALL specify one or more associated Adaptive Switching Processes intended as a hypothetical render model that can adaptively and seamlessly decode Switching Sets that conform to those constraints.

Media Profiles SHALL specify constraints on picture sequences and their storage in CMAF Fragments, and the resulting switching points that conform to an Adaptive Switching Process.

A Media Profile Adaptive Switching Process SHALL specify CMAF Fragment or sample splicing, initialization of a decoder or multiple decoders, scaling of video size, selection of tiles, selection of frame rates, combination of layers, rending a point of view, synchronization to presentation time, and other codec functions necessary to adaptively switch between one or more alternative CMAF Tracks in a conforming Switching Set at Media Profile defined switching points.

9.2.3 Switching Set Constraints and Adaptive Switching Processes for NAL Structured Video Tracks and Media Profiles

9.2.3.1 General Switching Set Constraints for NAL Structured Video Tracks and Media Profiles

[[Editor’s note: This recommendation is still under discussion. NBs are requested to provide comment on this section.]]

Each CMAF Track in a CMAF Switching Set containing NAL Structured video conforming to MPEG-4 Part 15 NAL Structure Video Track Format [ISOVIDEO] SHALL conform to general Switching Set constraints in 7.3.10 and general requirements in 9.2.2; and:

1. SHALL include a CMAF Header containing a Track Header Box ('tkhd') with normalized display `width` and `height` values, and one sample entry that describes

every NAL Structured Video Sample in the Track, i.e. the profile, level, etc. are greater or equal to those used to encode each CMAF Fragment. See Section 9 for more information on calculating normalized display width and height.

2. SHALL contain the same video sample entry as all other CMAF Tracks in the Switching Set, e.g. `‘avc1’`.
3. SHALL contain only complete Coded Video Sequences delimited by SAP type 1 or 2 in all CMAF video Fragments.
4. SHALL store all referenced video decoding parameter sets in the Decoder Configuration Record in the sample entry for appropriate sample descriptions, e.g. ‘avc1’.
5. SHALL store all referenced video decoding parameters in SPS and PPS NAL Units in SAP Type 1 and 2 video samples for appropriate sample descriptions, e.g. ‘avc3’. In this case, a decoding parameter set in the sample entry of the CMAF Header SHALL NOT be referenced by slice parameter indexes, and is only used for decoder and display initialization.

CMAF Switching Sets MAY be constrained for Single Initialization, in which case Players only need to process a CMAF Header once before the first CMAF Fragment is decoded, and can switch between CMAF Tracks in the same CMAF Switching Set at that start of any CMAF Fragment without processing any additional CMAF Headers.

Note: A Player’s Adaptive Switching Process is not constrained by this specification. The Adaptive Switching Processes defined are hypothetical, to define Player interoperability with the CMAF Switching Set encoding constraints. A Player that has multiple decoders, tightly integrates manifest and media processing, etc. can perform additional switching optimizations, for instance to avoid multiple initialization of a Switching Set that conforms to Multiple Initialization Switching Set constraints.

9.2.3.2 Single Initialization Switching Set Constraints for NAL Structured Video Tracks and Media Profiles

Switching Sets conforming to the CMAF video Switching Set constraints above MAY be additionally constrained to enable the Single Initialization Adaptive Switching Process on standard ISOBMFF parsers and NAL Structure Video decoders. Constraints for Single Initialization NAL Structured Video Switching Sets SHALL conform to section 9.2.4, and:

1. SHALL use v1 Track Run Boxes ('trun') and negative composition offsets as needed to adjust CMAF Fragment track presentation time to `baseMediaDecodeTime` and SHALL NOT use an edit list in the track (no offset edit lists).
2. SHALL constrain all CMAF Tracks in the Switching Set to the same CMAF Media Profile.
3. SHALL constrain all CMAF Headers in the Switching Set to an identical decoder configuration record, or use a sample description with all slice NAL referenced decoding parameters stored in a sample within the CMAF Fragment, e.g. 'avc3'.

Note: In practice, Single Initialization allows continuous live streaming of independently encoded video programs that do not rely on different CMAF Headers to adjust presentation time or provide decoding and display parameters.

9.2.3.3 Multiple Initialization Adaptive Switching Process for NAL Structured Video Tracks and Media Profiles

CMAF Switching Sets that conform to Multiple Initialization constraints require that the CMAF Header of a CMAF Track is processed before decoding each sequence of CMAF Fragments from that Track. Any offset edit list in the CMAF Header SHALL be applied to the composition times of samples to determine their track presentation times. Adaptive video scaling is required by display processors, as specified in section 7.3.10.

A CMAF Header from each Track in the Switching Set need only be downloaded once (it is assumed they do not change during live encoding), but the associated CMAF Header needs to be processed every time that Track is switched to, in order to set the correct decoder configuration record, decoding parameters, and indexes in the sample entry that match the parameter set indexes encoded in video slices in CMAF Fragments, apply the correct cropping and scaling, codec decoding and display parameters, and possible edit list offset to track presentation time. It is not assumed that decoders can determine which CMAF Track each CMAF Fragment is from, or store the CMAF Header for each CMAF Track in the decoder. Players that process CMF Headers on each Track switch can evaluate parameter changes to minimize the amount of reconfiguration required and minimize presentation disruption.

9.2.3.4 Single Initialization Adaptive Switching Process for NAL Structured Video Tracks and Media Profiles

CMAF Switching Sets that conform to Single Initialization constraints specified by a Media Profile, such as those in Annex A.2, MAY be decoded and displayed using standard ISOBMFF parsers and NAL Structured Video decoders by sequencing CMAF Fragments from the same Switching Set after once initializing a Player with a CMAF Header for that Media Profile. Adaptive video scaling is required by display processors, as specified in section 7.3.10.

9.2.3.5 Identifiers for NAL Structured Video Switching Set Constraints

The following identifiers are defined to signal CMAF Switching Set constraints associated with an Adaptive Switching Process:

- Single Initialization CMAF Switching Set: urn:mpeg:cmaf:siss
- Multiple Initialization CMAF Switching Set: urn:mpeg:cmaf:miss
- Manifests can identify Single Initialization Switching Sets explicitly, but if they do not, the above defaults based on the sample entry can be used by Players.

9.2.4 CMAF Track Format Constraints for NAL Structured Video

9.2.4.1 Track Header Box ('tkhd')

For video Tracks, the fields of the Track Header Box ('tkhd') SHALL be set to the values constrained below and specified in [ISOBMFF].

- flags = 0x000007, except for the case where the track belongs to an alternate group
- The values of width and height are the decoded and cropped image size in spatial samples measured on a uniformly sampled square grid as specified in [ISOBMFF].
- The values of width and height in a CMAF Track Header Box SHALL be normalized to width and height of the encoded video, as defined below:
  - The video elementary stream SHALL contain only spatial samples intended for presentation after SPS cropping parameters are applied in the decoder. The Clean Aperture box SHOULD NOT be present.
  - The normalized Presentation height SHALL be the number of vertical samples after SPS cropping parameters are applied to the vertical sample count in the SPS referenced by video slices.

    Note: The height field of the visual sample entry is also the number of encoded vertical samples after cropping.
  - The Normalized Presentation width SHALL be the number of horizontal samples after SPS cropping parameters are applied to the encoded horizonta sample count in the referenced SPS parameter, then multiplied by the sample aspect ratio. The sample aspect ratio is defined by the PixelAspectRatioBox if present in the sample entry, or otherwise by the aspect_ratio_idc (and if applicable the sar_width and sar_height) in SPS VUI parameters.

    Note: The width field in the visual sample entry is the number of encoded horizontal samples after cropping, not the value of the Track Header width field.

- The value of the matrix field signals the video orientation. Non-identity matrices SHALL be rotations in multiples of 90 degrees.
  - When video is not rotated, matrix SHALL be {0x00010000, 0, 0, 0, 0x00010000, 0, 0, 0, 0x40000000}.
  - When video should be rotated 90 degrees clockwise for display, matrix SHOULD be {0, 0x00010000, 0, 0xFFFF0000, 0, 0, height<<16, 0, 0x40000000}.
  - When video should be rotated 180 degrees for display, matrix SHOULD be {0xFFFF0000, 0, 0, 0, 0xFFFF0000, 0, width<<16, height<<16, 0x40000000}.
  - When video should be rotated 90 degrees counter-clockwise for display, matrix SHOULD be {0, 0xFFFF0000, 0, 0x00010000, 0, 0, 0, width<<16, 0x40000000}.

NOTE A Player is expected to select a video aperture that adapts the aspect ratio of the decoded video image (Track Header `width` /`height)` to that of the current display. The Player can frame the video aperture with letterbox bars, pillarbox bars, a window, etc. A player needs to scale all CMAF Fragments from a Switching Set to the selected video aperture to maintain seamless playback without size, shape, or location errors.

9.2.4.2 Sample Description Box ('stsd')

The Sample Description Box (`'stsd'`) in a video track SHALL contain a Sample Entry which SHALL include:

- `width` and `height` field values equal to the largest cropped horizontal and vertical sample counts in any Sequence Parameter Set referenced by a video slice in the ISOBMFF track; and
- a Decoder Configuration Record that:
  - SHOULD signal the lowest Profile, Level, height and width values in the first (or only) parameter set that are sufficient to decode all CMAF Fragments in the CMAF Track, and SHALL signal other sequence parameter set and picture parameter set fields used by the video track as specified in [ISOVIDEO];

Note: Although it is valid to signal a higher profile and level than is necessary to decode the CMAF Fragments in the CMAF Track, that may exclude decoders capable of decoding the CMAF Fragments, but not able to initialize the higher profile and level signaled in the CMAF Header.

- For a Visual Sample Entry with codingname `'avc3'` SHALL contain a single parameter set of each type, intended only for decoder and display initialization (containing one SPS and one PPS NAL for AVC Video). The parameter set SHALL have a parameter set index that will not be referenced by video slices for decoding (i.e. these parameter sets are not referenced by the stream). The SPS parameter values, profile_idc, level_idc, picture width, and picture height SHALL equal or exceed the values contained in SPS NALs in CMAF Fragments in the CMAF Track. Each video slice in the CMAF Track SHALL reference SPS and PPS NALs stored in that video CMAF Fragment, in the first sample of each Coded Video Sequence;
- For a Visual Sample Entry with codingname `'avc1',` SHALL contain one or more decoding parameter sets. (Containing SPS and PPS NALs for AVC Video). Each video sample in the CMAF Track SHALL reference this parameter set in the sample entry;
- SHOULD set `LengthSizeMinusOne` field to the value "`3`" (to indicate 4 bytes) to simplify conversion of elementary streams between MPEG-2 TS bytestreams with startcodes and [ISOVIDEO] with NAL length headers.

  Note: The size of the NAL header length field defined in video tracks conforming to [ISOVIDEO] is stored in the field `LengthSizeMinusOne` in the corresponding decoder configuration record. For AVC video `AVCDecoderConfigurationRecord` and for HEVC video `HEVCDecoderConfigurationRecord`.

9.2.4.3 Picture Access Units

Picture Access Units SHALL conform to the requirements of a sample for the indicated sample entry, as specified in [ISOVIDEO]. Picture Access Units MAY be delimited by

Access Unit Delimiter NALs. Each Access Unit is a sample stored in a Media Data Box (`'mdat'`), as specified in [ISOVIDEO].

CMAF Segments containing inline parameter sets SHALL contain all SPS and PPS NALs referenced by a Coded Video Sequence in the first Access Unit of that sequence, immediately following its first Access Unit Delimiter NAL (if any).

Access Units of type `'avc3'` MAY retain filler data (NAL units or SEI messages) and SEI messages that might change hypothetical reference decoder bitstream conformance if removed. Bitstream conformance could be necessary when bitstreams are to be repackaged and conformance tested in MPEG-2 Transport Streams.

As specified in [ISOVIDEO], timing information provided within a video elementary stream SHALL be ignored. Instead, sample timing in the Track Run Box (`'trun'`) SHALL determine picture presentation order and timing.

9.2.4.4 Additional Random Access Pictures within CMAF Video Fragments

It is recommended to encode a picture of Stream Access Point (SAP) type 1, 2, or 3 as defined by [ISOVIDEO] approximately every 2 seconds or less within a Video Track to allow periodic random access. For longer coded video sequences and resulting Fragment durations, additional type 3 SAPs ("open GOP" independently decodable pictures) allow fast forward, rewind, and continuous playback of all subsequently presented pictures, while improving video continuity and lowering bitrate relative to coded video sequences of the equivalent duration.

9.3 AVC Track Constraints

9.3.1 Storage of AVC Elementary Streams

9.3.1.1 Conformance

AVC video tracks SHALL comply with Section 5 of [ISOVIDEO].

9.3.1.2 Visual Sample Entry

The syntax and values for visual sample entry SHALL conform to `AVCSampleEntry` (`'avc1'`) or `AVCSampleEntry` (`'avc3'`) as defined in [ISOVIDEO], and the general video sample entry requirements of Section 9.2.4.2.

9.3.2 Constraints on AVC Elementary Streams

9.3.2.1 Picture type

All pictures SHALL be encoded as coded frames, and SHALL NOT be encoded as coded fields.

Media Profiles with with sample entry `‘avc3’` SHALL conform to Single Initialization Switching Set constraints, unless externally explicitly signaled using the indicators in 9.2.3.5.

Note: `'avc3'` Coded Video Sequences, contain the necessary Sequence Parameter Set and Picture Parameter Set NAL Units to signal decoding parameters changes allowed between CMAF Tracks in the same Single Initialization Switching Set.

Switching Sets containing a Media Profile, such as those listed in Annex A.2, conforming to CMAF Switching Set constraints, and AVC Media Profiles with sample entry `‘avc1’` SHALL use the Multiple Initialization Adaptive Switching Process by default, unless identified as conformant to the Single Initialization Adaptive Switching Process using the indicators in 9.2.3.5.

9.3.2.2 Sequence Parameter Sets (SPS)

9.3.2.2.1 SPS Field Constraints

Sequence Parameter Set NAL Units that occur in an AVC video CMAF Track SHALL conform to [AVC] with the following additional constraints:

– The following fields have pre-determined values as follows:

  - frame_mbs_only_flag SHALL be set to 1
  - mb_adaptive_frame_field_flag SHALL be set to 0 if present
  - vui_parameters_present_flag SHALL be set to 1
  - gaps_in_frame_num_value_allowed_flag SHOULD be set to 0
- The values of the following fields SHALL NOT change throughout a CMAF Track:
  - chroma_format_idc
  - bit_depth_luma_minus8
  - bit_depth_chroma_minus8
- The values of the following fields SHALL NOT change throughout an 'avc1' CMAF Track:
  - profile_idc
  - level_idc
  - pic_width_in_mbs_minus1
  - pic_height_in_map_units_minus1
  - frame_crop_left_offset
  - frame_crop_right_offset
  - frame_crop_top_offset
  - frame_crop_bottom_offset
  - max_num_ref_frames
- The values of the following fields SHOULD NOT change throughout a CMAF Track:
  - seq_parameter_set_id

9.3.2.2.2 Visual Usability Information (VUI) Parameters

VUI parameters that occur within a CMAF AVC video Track SHALL conform to [AVC] with the following additional constraints:

- The following fields SHALL have pre-determined values as follows:
    - If video_signal_type_present_flag is set to 0, then video_full_range_flag SHALL be inferred to be 0.

  Note: This indicates normal black "setup", i.e. 16 for 8-bit video.

  - aspect_ratio_info_present_flag SHALL be set to 1. aspect_ratio_idc SHALL NOT be set to 0. If sample aspect ratio is 1:1 (square), aspect_ratio_idc SHALL be set to 1.

    Note: This indicates Sample aspect ratio is present, not picture aspect ratio.

  - chroma_loc_info_present_flag SHALL be set to 0

    Note: Indicates standard progressive 4:2:0 color subsampling.

  - overscan_info_present_flag, if present, SHALL be set to 0
- The following fields SHOULD have pre-determined values as follows:
  - colour_description_present_flag SHOULD be set to 1. (Note that the actual values of the color description are defined by the profiles in Annex A.)
  - If colour_description_present_flag is set to 0, this SHALL indicate the following default video coding and mastering:
    - SHALL be encoded using the video parameters defined by [R709]; and
    - Video SHOULD be graded in a viewing environment that complies with [R2035] for presentation on a display which uses the electro-optic transfer function specified in [R1886] with a peak luminance of 100 cd/m2.

    NOTE Per [AVC], if the colour_description_present_flag is set to 1, the colour_primaries, transfer_characteristics and matrix_coefficients fields SHALL be present.
- The values of the following fields SHALL NOT change in a CMAF Track:
  - low_delay_hrd_flag

- colour_primaries, when present
- transfer_characteristics, when present
- matrix_coefficients, when present

9.3.2.3 Picture Parameter Sets (PPS)

Picture Parameter Set NAL Units that occur within a CMAF SHALL conform to [AVC] with the following additional constraints:

The value of the entropy_coding_mode_flag SHALL NOT change throughout a CMAF Track using an 'avc1' sample entry.

9.3.2.4 Measurement of Maximum Bitrate

The maximum bitrate of [AVC] elementary streams SHALL be calculated by implementation of the buffer and timing model defined in [AVC] Annex C.

9.4 Video codec parameters

9.4.1 AVC signaling of "codecs" parameter (Informational)

Presentation Applications SHOULD signal the video codec profile and level of each AVC Track and Switching Set using parameters conforming to [RFC6381] and [ISOVIDEO] Annex E.

9.4.2 Encoding Overview

Video Tracks SHALL only encode spatial samples intended for presentation on all displays. Padding such as letterbox and pillarbox bars SHOULD NOT be encoded.

The active image SHOULD be upper left justified, and only one row or one column of partially filled macroblocks encoded if the image height or width is not a multiple of the coding block size. Extra samples SHALL be cropped by setting SPS cropping parameters frame_crop_bottom_offset or frame_crop_right_offset for AVC.

The VUI parameter, `aspect_ratio_idc`, SHALL be present if square samples are not encoded, and SHOULD always be present to avoid incorrectly assuming the default value of 1.

Each device and display system is expected to frame the decoded and cropped video to its video display aperture using methods such as scaling, stretching, cropping, padding with letterbox or pillarbox bars, or framing in a window. Each visual Segment may be encoded with different vertical and horizontal sample counts, so devices need to scale each Segment to fit the same display aperture and maintain the correct picture aspect ratio to avoid scaling and placement errors during adaptive switching.

9.4.3 Video color and dynamic range mastering

Video streams conforming to CMAF SHALL be encoded using the transfer characteristics, color parameters, and grading specified in each video Media Profile, and that Media Profile indicated by its compatibility brand in the CMAF Header. See Video Media Profiles and Track Brands, Annex [A.2].

For all Media Profiles, unless signaled otherwise according to the mechanism defined in that video Media Profile, default transfer characteristics and grading SHALL be assumed. Default grading is defined as a viewing environment that complies with [R2035] for presentation on a display which uses the electro-optic transfer function specified in [R1886] with a peak luminance of 100 cd/m2.

Video MAY be graded for presentation on a display which uses an electro-optic transfer function not specified in [R1886] with a peak luminance greater than 100 cd/m2, in which case, the grading profile SHOULD be signaled by VUI and/or [ST2086] metadata in SEI messages contained in the sample entry `'avcC'` box, or other mechanism defined in the video Media Profile.

9.5 Video Elementary Stream Embedded Captions

[[Editor's note: The MPEG File Systems group is specifying a method of signaling the presence of caption data in video SEI NAL units in ISOBMFF, and a planned amendment to Part 30. The amendment is expected to be available for reference here when CMAF is published.]]

9.5.1 Carriage of CEA-608/708 in SEI messages

CEA 608/708 caption data [CEA608], [CEA708]MAY be stored in SEI messages described as user data registered by Rec. ITU-T T.35, with SEI `payloadType = 4` and the registered identifier in the field `user_data_registered_itu_t_t35`. See [AVC] Section D.2.5.

9.5.2 Signaling the presence of CEA-608/708 in SEI messages

The presence of CEA-608/708 data in SEI messages in a video track SHOULD be signaled as specified by MPEG-4 Part 30 Amendment 1 [ISOTXT].

9.5.3 CEA-608/708 in SEI messages

ANSI/SCTE 128 2013-a [SCTE128] defines in section 8.1 Encoding and transport of caption, active format description (AFD) and bar data. Based on this, a video track MAY carry SEI messages that carry CEA-608/708 closed caption data. The SEI message `payloadType=4` is used to indicates that Rec. ITU-T T.35 based SEI messages are contained in an SEI message.

In summary the following is included in [SCTE128] to signal CEA-608/708 in SEI messages:

- SEI `payloadType` is set to 4
- `itu_t_t35_country_code` — A fixed 8-bit field, the value of which shall be `0xB5`.
- `itu_t_t35_provider_code` — A fixed 16-bit field registered by the ATSC. The value shall be `0x0031`.

- `user_identifier` — This is a 32-bit code that indicates the contents of the `user_structure()` and is `0x47413934` (“GA94”).
- `user_structure()` — This is a variable length data structure `ATSC1_data()` defined in section 8.2 of ANSI/SCTE 128 2013-a.
- `user_data_type_code` is set to `0x03` for indicating captioning data in the `user_data_type_structure()`
- `user_data_type_structure()` is defined in section 8.2.2 of [SCTE128] for Closed Captioning and defines the details on how to encapsulate the captioning data.

It is recommended that Manifest signal the presence of SEI-stored closed captions, and the services and languages included. Players may automatically select Tracks signaled to contain captions if the user or Player indicates a preference for audio accessibility.

10 CMAF Audio Tracks

10.1 Overview

This section describes audio Tracks, their general file and Track constraints, and requirements for audio Media Profiles that specify codecs, elementary streams and sample formats.

In addition, a specific Track format and Media Profile binding is specified for a “core” set of widely interoperable AAC stereo codecs. The system layer specified in [MP4SYS] is used to embed the AAC audio elementary streams in Tracks.

10.2 General Requirements for CMAF Media Profiles (Audio)

All audio Media Profiles SHALL define a compatibility brand for the `'ftyp'` box in the CMAF Header.

All audio Media Profiles SHALL define the sample entry and decoder configuration information that is required in the CMAF Header.

All audio Media Profiles SHALL define Internet Media Type “codecs” parameters that can be used to identify the CMAF Track in a Manifest.

All audio Media Profiles SHALL define CMAF Fragment and sample encoding constraints and signaling necessary to random access Segments and samples, and initiate decoded output (e.g. “priming” predicted samples).

CMAF Media Profiles MAY define Switching Set constraints and at least on Adaptive Switching Process that allows seamless switching of CMAF audio Tracks at defined switch points in a conforming Switching Set.

All audio Tracks SHOULD contain loudness and dynamic range information in the bitstream conforming to DRC Presentation Mode in ISO/IEC 14496-3:2009 and Amendment 4 [AAC].

See [A.2] for details of audio media profiles and track brands.

10.3 Audio Track File Constraints

10.3.1 General

The common ISO Media track structure for storing audio in a CMAF Track is described below. All audio formats SHALL comply with these constraints.

10.3.2 Track Header Box (`'tkhd'`)

For audio tracks, the fields of the Track Header Box SHALL be set to the values specified below. Other fields may be set per [ISOBMFF] Section 8.3.2.

- `flags` = 0x000007, except for the case where the track belongs to an alternate group
- `layer` = 0
- `volume` = 0x0100

- `matrix` = {0x00010000, 0, 0, 0, 0x00010000, 0, 0, 0, 0x40000000} // unity matrix
- `width` = 0
- `height` = 0
- `duration` = 0 (duration of the `'trak'` box, which contains no samples)

10.3.3 Sync Sample Box (`'stss'`)

The Sync Sample Box (`'stss'`) SHALL NOT be used.

NOTE "sync sample" in movie Fragments cannot be signaled by the absence of the Sync Sample box (`'stss'`) or by the presence of the Sync Sample box (`'stss'`), since this box is not designed to list sync samples in movie Fragments.

- For audio formats in which every audio access unit is a random access point (sync sample), signaling can be achieved by other means such as setting the `'sample_is_non_sync_sample`' flag to "0" in the '`default_sample_flags`' field in the Track Extends box (`'trex'`).

10.3.4 Handler Reference Box (`'hdlr'`)

The syntax and values for the Handler Reference Box (`'hdlr'`) for audio tracks SHALL conform to [ISOBMFF] with the following additional constraints:

- The `handler_type` field SHALL be set to "`soun`"

10.3.5 Sound Media Header Box (`'smhd'`)

The syntax and values for the Sound Media Header Box SHALL conform to [ISOBMFF] with the following additional constraints:

- The following fields SHALL be set as defined:
  - `balance` = 0

10.3.6 Sample Description Box ('stsd')

As specified in [ISOBMFF], the Sample Description Box ('stsd') contains the AudioSampleEntry Box or AudioSampleEntryV1 Box.

See the guidelines in section 12.2.3.1 of the fifth edition of [ISOBMFF]. For each of the audio formats supported by the CMAF, a specific audio sample entry box is derived from one of the AudioSampleEntry Boxes defined in [ISOBMFF]. Each codec-specific SampleEntry Box is identified by a unique codingname value, and specifies the audio format used to encode the audio track, and describes the configuration of the audio elementary stream.

Table 7 — AAC Core audio formats lists audio formats used by Media Profiles that are defined in the CMAF Presentation Profile in Annex A.1.2, and the corresponding SampleEntry that is present in the Sample Description Box for each format.

Table 7 — AAC Core audio formats

codingname	Audio Format	SampleEntry Type	Section Reference
mp4a	MPEG-4 AAC LC	MP4AudioSampleEntry	Section 10.5.3
mp4a	MPEG-4 HE-AAC or v2	MP4AudioSampleEntry	Section 10.5.5

10.3.7 Shared elements of AudioSampleEntry

For all audio formats supported by CMAF, the following elements of the AudioSampleEntry box defined in [ISOBMFF] are shared:

```
class AudioSampleEntry(codingname)
   extends SampleEntry(codingname)
{
   const unsigned int(32)zreserved[2] = 0;
   template unsigned int(16)  channelcount;
   template unsigned int(16)  samplesize = 16;
   unsigned int(16)           pre_defined = 0;
   const unsigned int(16)     reserved = 0;
   template unsigned int(32)  sampleRate;
```

```
    (codingnamespecific)Box
}
```

For all audio Media Profiles in CMAF, the value of the `samplesize` parameter SHALL be set to 16.

Each of the audio formats supported by the CMAF extends the `AudioSampleEntry` box through the addition of a box (shown above as “`(codingnamespecific)Box`”) containing codec-specific information that is placed within the `AudioSampleEntry`. This information is described in the following codec-specific sections.

10.4 General Constraints on Audio Codecs

10.4.1 Introduction

Audio codecs often define features and other details that present options for implementers. In order to increase interoperability, CMAF seeks to limit these options. While every codec offers its own set of choices, this section presents general rules that should be applied to all CMAF File-compliant audio.

10.4.2 Byte Order

If a codec requires explicit signaling of byte order (i.e. big-endian or little-endian), then all CMAF Media Profiles for that codec SHALL require that a particular byte order be used.

10.4.3 Objects

If a codec defines any optional object types, then all CMAF Media Profiles for that codec SHALL indicate which object types may be present.

10.4.4 Profiles and Levels

If a codec defines any profiles or levels, then all CMAF Media Profiles for that codec SHALL indicate which profiles and/or levels are included.

10.4.5 Priming

If a decoder requires multiple input packets before it produces its first output packet, then all CMAF Media Profiles for that codec SHALL indicate how audio that is not intended for presentation is to be signaled in the Track. It is recommended that it be signaled with an edit list as described in 7.5.12.

10.4.6 CMAF Fragment Independence

With the exception of audio samples needed for priming (10.4.5) and flushing a progressive audio decoder, CMAF Media Profiles for audio codecs listed in Annex A.3 SHALL require every audio sample in a CMAF Fragment be decodable without reference to other Fragments in the Track.

10.4.7 Loudness

If a codec defines signaling for loudness (e.g. relative to dialog), then all CMAF Media Profiles for that codec SHALL strongly recommend or require that Tracks carry that signaling.

10.4.8 Sample mapping

All CMAF Media Profiles for a codec SHALL define how codec frames or access units are mapped to ISOBMFF Samples.

10.4.9 Channels

If a codec can encode multiple (dependent or independent) audio channels, then all CMAF Media Profiles for that codec SHALL specify which channel configurations are included. In addition, channel configuration SHALL NOT change within a single CMAF Track.

10.4.10 Out-of-band signaling

CMAF Media Profiles for audio codecs SHALL prohibit encoding variations that place the audio decoder module inputs other than within the CMAF Track.

Note: This does not preclude track selection metadata normally found in the manifest (e.g. language, video description) or personalization items (e.g. dialog loudness, video description enabled).

10.4.11 Framing

CMAF Media Profiles for audio codecs should discourage the use of codec-specific framing information (e.g. LATM) and instead rely on ISOBMFF sample identification.

10.4.12 Flags and Parameter Values

CMAF Media Profiles for audio codecs should seek to define fixed values for configuration flags and other parameters that reflect common encoding practices and client capabilities.

10.5 CMAF Requirements for AAC Audio

10.5.1 Signaling

The signaling of the codecs MIME parameter is according to [RFC6381] as shown in

Table 8.

Table 8 — AAC Codecs MIME parameter according to RFC 6381

Codec	MIME type	codecs parameter	ISOBMFF Encapsulation
MPEG-4 AAC-LC	audio/mp4	mp4a.40.2	ISO/IEC 14496-14
MPEG-4 HE-AAC	audio/mp4	mp4a.40.5	ISO/IEC 14496-14
MPEG-4 HE-AAC v2	audio/mp4	mp4a.40.29	ISO/IEC 14496-14

Note: HE-AAC is a superset of AAC-LC, and HE-AACv2 is a superset of HE-AAC. It is assumed that a decoder capable of fully decoding HE-AACv2 is also capable of decoding HE-AAC or AAC-LC.

10.5.2 General Considerations for AAC Audio Encoding

The AAC codec uses frames of a fixed length, and a transform which applies over two frames. To obtain the correct audio from a frame, both frames in the transform are needed, and hence both the prior encoded frame and the current encoded frame need to be decoded to output the first frame. This is sometimes called "priming".

In order to obtain correctly synchronized audio at the start of a Track, encoders typically add some silent audio before the start of the audio signal so the first audio frame will decode in sync with the first video sample. All encoders necessarily encode an additional frame to output the last frame of audio content. All frames at the end of a recording that contain content added to flush the encoder SHOULD NOT be included in a Track.

Files SHALL indicate that initial added audio be removed, by using an edit list, as specified in Section 7.5.12.

The pre-roll sample group SHOULD be used to indicate how many previous frames need to be passed to the decoder in order to get the correct audio out of any frame; the edit list (section 10.4.5) is used to discard any output from frames present purely to allow for this pre-roll.

Figure 14 — Example of AAC Access Units illustrates an AAC bit-stream, both encoded and framed in Access Units in a track.

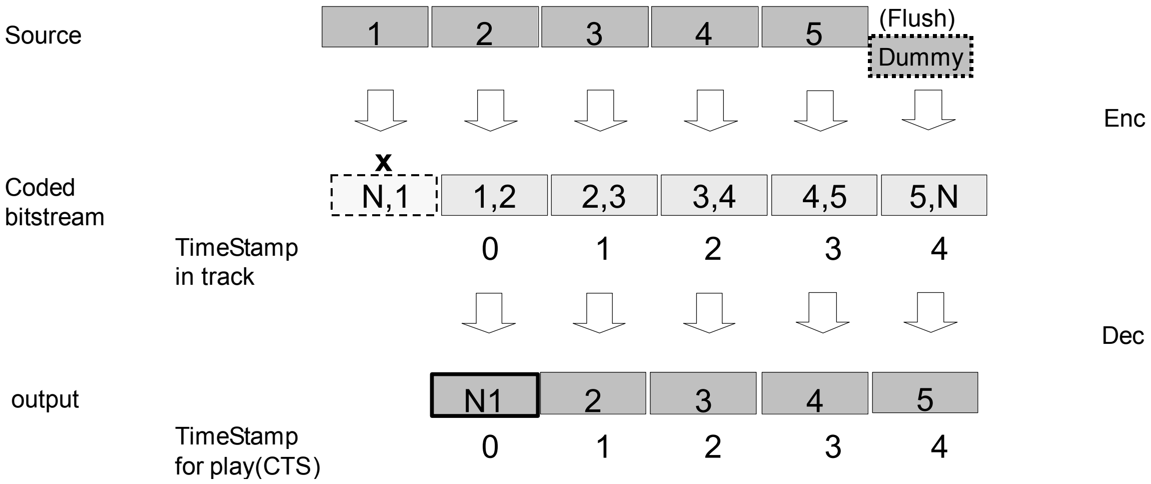


Figure 14 — Example of AAC Access Units

In Figure 5, the enclosed numbers indicate which input blocks are used for prediction and encoding in an encoded frame; for example, [1,2] indicates that block 2 is predicted from frame 1. The source block labelled “Dummy” is audio or silence encoded to output block 5, but is not included in the coded stream in the Track. The letter N indicates encoded silence in the last frame. The last audio frame MAY contain silence if the audio source ends prior to the frame boundary.

The audio stream SHOULD contain DRC and loudness metadata according to [AAC]. The audio encoder SHOULD set the Program Reference Level to the loudness level of the audio stream. The audio decoder SHALL use the Program Reference Level, if available, to achieve a desired target loudness, if applicable. The audio encoder SHOULD generate DRC metadata for light compression encoded in the `dyn_rng_ctl` and `dyn_rng_sgn` fields of `dynamic_range_info()` in the FIL element and DRC metadata for heavy compression in the `compression_value` field of `MPEG4_ancillary_data()` in the DSE (data stream element). The audio decoder SHALL apply the DRC metadata, if present, according to [AAC] Amendment 4 including the DRC Presentation Mode value of the `drc_presentation_mode` field.

10.5.3 MPEG-4 AAC LC

10.5.3.1 Storage of MPEG-4 AAC LC Media Samples

Storage of MPEG-4 AAC LC elementary stream samples within a CMAF Track SHALL be according to [MP4]. The following additional constraints also apply:

- An audio sample SHALL consist of a single AAC audio access unit.
- The parameter values of `AudioSampleEntry`, `DecoderConfigDescriptor`, and `DecoderSpecificInfo` SHALL be consistent with the configuration of the AAC audio stream.

10.5.3.2 Audio Sample Entry Box for MPEG-4 AAC LC

10.5.3.2.1 Field Values

The syntax and values of the `AudioSampleEntry` SHALL conform to `MP4AudioSampleEntry` (`'mp4a'`) as defined in [MP4], and the following fields SHALL be set as defined:

- channelcount = as appropriate for the stream

For MPEG-4 AAC, the (codingnamespecific)Box that extends the MP4AudioSampleEntry is the ESDBox defined in [MP4], which contains an ES_Descriptor.

10.5.3.2.2 ESDBox

The syntax and values for ES_Descriptor SHALL conform to [MP4SYS], and the fields of the ES_Descriptor SHALL be set to the following specified values. Descriptors other than those specified below SHALL NOT be used.

- ES_ID = 0
- streamDependenceFlag = 0
- URL_Flag = 0;
- OCRstreamFlag = 0
- streamPriority = 0
- decConfigDescr = DecoderConfigDescriptor
- slConfigDescr = SLConfigDescriptor, predefined type 2

10.5.3.2.3 DecoderConfigDescriptor

The syntax and values for DecoderConfigDescriptor SHALL conform to [MPEG4S], and the fields of this descriptor SHALL be set to the following specified values. In this descriptor, decoderSpecificInfo SHALL be used, and ProfileLevelIndicationIndexDescriptor SHALL NOT be used.

- objectTypeIndication = 0x40 (Audio)
- streamType = 0x05 (Audio Stream)
- upStream = 0
- decSpecificInfo = AudioSpecificConfig

10.5.3.2.4 AudioSpecificConfig

The syntax and values for AudioSpecificConfig SHALL conform to [AAC], and the fields of AudioSpecificConfig SHALL be set to the following specified values:

- `audioObjectType` = 2 (AAC LC)
- `GASpecificConfig`

Channel assignment SHALL NOT be changed within the audio stream within a Track.

10.5.3.2.5 GASpecificConfig

The syntax and values for `GASpecificConfig` SHALL conform to [AAC], and the fields of `GASpecificConfig` SHALL be set to the following specified values:

- `frameLengthFlag` = 0 (1024 lines IMDCT)
- `dependsOnCoreCoder` = 0
- `extensionFlag` = 0

10.5.4 MPEG-4 AAC LC Elementary Stream Constraints

10.5.4.1 General Encoding Constraints

MPEG-4 AAC elementary streams SHALL conform to the requirements of the MPEG-4 AAC profile at Level 2 as specified in [AAC] with the following restrictions:

- Only the MPEG-4 AAC LC object type SHALL be used.
- The elementary stream SHALL be a Raw Data stream. ADTS and ADIF SHALL NOT be used.
- The transform length of the IMDCT for AAC SHALL be 1024 samples for long and 128 for short blocks.
- The following parameters SHALL NOT change within the elementary stream
    - Audio Object Type
    - Sampling Frequency
    - Channel Configuration

10.5.4.2 Syntactic Elements

10.5.4.2.1 The syntax and values for syntactic elements SHALL conform to [AAC]. Arrangement of Syntactic Elements

- Syntactic elements SHALL be arranged in the following order for the channel configurations below.
  - <SCE>, <optional additional elements>, <TERM>… for mono
  - <CPE>, <optional additional elements>, <TERM>... for stereo

  Note: Angled brackets (<>) are delimiters for syntactic elements.

10.5.4.2.2 individual_channel_stream

- The syntax and values for `individual_channel_stream` SHALL conform to [AAC]. The following fields SHALL be set as defined:
  - `gain_control_data_present` = 0

10.5.4.2.3 Maximum Bitrate

The maximum bitrate of MPEG-4 AAC LC [2-Channel] elementary streams SHALL be calculated in accordance with the AAC buffer requirements as defined in ISO/IEC 14496-3:2009, section 4.5.3. Only the raw data stream SHALL be considered in determining the maximum bitrate (system-layer descriptors are excluded).

10.5.5 MPEG-4 HE-AAC and HE-AACv2

10.5.5.1 Storage of MPEG-4 HE-AAC and HE-AACv2 Media Samples

Storage of MPEG-4 HE-AAC or HE-AACv2 elementary streams within a CMAF Track SHALL be according to [MP4FILE]. The following constraints also apply.

- An audio sample SHALL consist of a single HE-AAC or HE-AACv2 audio access unit.
- The parameter values of `AudioSampleEntry`, `DecoderConfigDescriptor`, and `DecoderSpecificInfo` SHALL be consistent with the configuration of the MPEG-4 HE-AAC or HE-AACv2 audio stream.

10.5.5.2 Audio Sample Entry Box for MPEG-4 HE-AAC and HE-AACv2

10.5.5.2.1 Field Values

The syntax and values of the `AudioSampleEntry` box SHALL conform to `MP4AudioSampleEntry` (`'mp4a'`) defined in [MP4FILE], and the following fields SHALL be set as defined:

- `channelcount` = 1 as appropriate for the stream

For the core MPEG-4 AAC, the `(codingnamespecific)Box` that extends the `MP4AudioSampleEntry` is the `ESDBox` defined in ISO 14496-14 [MP4FILE], which contains an `ES_Descriptor`.

10.5.5.2.2 ESDBox

The `ESDBox` contains an `ES_Descriptor`.

- The syntax and values for `ES_Descriptor` SHALL conform to [MP4SYS], and the fields of the `ES_Descriptor` SHALL be set to the following specified values. Descriptors other than those specified below SHALL NOT be used.
    - `ES_ID` = 0
    - `streamDependenceFlag` = 0
    - `URL_Flag` = 0
    - `OCRstreamFlag` = 0 (false)
    - `streamPriority` = 0
    - `decConfigDescr` = `DecoderConfigDescriptor`
    - `slConfigDescr` = `SLConfigDescriptor`, predefined type 2

10.5.5.2.3 DecoderConfigDescriptor

- The syntax and values for `DecoderConfigDescriptor` SHALL conform to [MPEG4S], and the fields of this descriptor SHALL be set to the following specified values. In this descriptor, `DecoderSpecificInfo` SHALL be used, and `ProfileLevelIndicationIndexDescriptor` SHALL NOT be used.

- `objectTypeIndication` = 0x40 (Audio)
- `streamType` = 0x05 (Audio Stream)
- `upStream` = 0
- `decSpecificInfo` = `AudioSpecificConfig`

10.5.5.2.4 AudioSpecificConfig

– The syntax and values for `AudioSpecificConfig` SHALL conform to [AAC] and the fields of `AudioSpecificConfig` SHALL be set to the following specified values:

  - `audioObjectType` = 2 (AAC LC)
  - `channelConfiguration` = 1 (for HE-AACv2 or mono HE-AAC) or 2 (for stereo HE-AAC), or as appropriate for streams with more channels.
  - `GASpecificConfig`
  - `extensionAudioObjectType` = 5 (SBR) or 29 (PS)

This configuration uses explicit hierarchical signaling to indicate the use of the SBR coding tool, and the PS coding tool.

10.5.5.2.5 GASpecificConfig

– The syntax and values for `GASpecificConfig` SHALL conform to [AAC], and the fields of `GASpecificConfig` SHALL be set to the following specified values.

  - `frameLengthFlag` = 0 (1024 IMDCT lines)
  - `dependsOnCoreCoder` = 0
  - `extensionFlag` = 0

10.5.6 MPEG-4 HE-AAC and HE-AACv2 Elementary Stream Constraints

10.5.6.1 General Encoding Constraints

Note: MPEG-4 HE-AACv2 is a superset of MPEG-4 AAC LC and MPEG-4 HE-AAC.

The MPEG-4 HE-AAC and HE-AACv2 elementary stream as defined in [AAC] SHALL conform to the requirements of MPEG-4 HE-AAC at Level 2 and MPEG-4 HE-AACv2 at Level 2, respectively, except as follows:

- The elementary stream MAY be encoded according to the MPEG-4 AAC LC, HE-AAC or HE-AACv2. Use of the MPEG-4 HE-AACv2 is recommended for 32 kbps or lower.
- When using HE-AAC and HE-AACv2 bitstreams, explicit backwards compatible signaling SHALL be used to indicate the use of the SBR and PS coding tools. CMAF Segments containing HE-AAC SHALL start with a type 1 SAP, notably, the SBR configuration information SHALL be in the first packet.
- The audio SHALL be encoded in mono, parametric stereo or 2-channel stereo. (Mono only if the source content is recorded in mono.)
- The IMDCT size for AAC SHALL be 1024 lines for long and 128 lines for short blocks.
- The elementary stream SHALL be a Raw Data stream. ADTS and ADIF SHALL NOT be used.
- The following parameters SHALL NOT change within the elementary stream:
    - Audio Object Type
    - Sampling Frequency
    - Channel Configuration

10.5.6.2 Syntactic Elements

10.5.6.2.1 Syntax and Values of Syntactic Elements

The syntax and values for syntactic elements SHALL conform to [AAC]. The following element SHALL NOT be present in an MPEG-4 HE-AAC or HE-AACv2 elementary stream:

· coupling_channel_element (CCE)

If the program_config_element (PCE) element is present then it SHALL only list a set of channels corresponding to one of the fixed channel configurations specific in Table 1.19 of [AAC], and the element SHALL NOT change during the track.

10.5.6.2.2 Arrangement of Syntactic Elements

- Syntactic elements SHALL be arranged in the following order for the channel configurations below.
- <SCE><optional additional elements><TERM>… for HE-AACv2 and mono HE-AAC
- <CPE><optional additional elements><TERM>... for stereo HE-AAC

10.5.6.3 Maximum Bitrate

The maximum bitrate of MPEG-4 HE-AAC or HE-AACv2 elementary streams in a CMAF SHALL be calculated in accordance with the AAC buffer requirements as defined in [AAC] section 4.5.3. Only the raw data stream SHALL be considered in determining the maximum bitrate (system-layer descriptors are excluded).

11 Subtitles and Captions

11.1 Introduction

CMAF supports the following formats for carrying subtitles and captions:

- WebVTT
- IMSC1 (a profile of TTML)
- CEA-608
- CEA-708

The term “subtitles” in this document is used to mean a visual presentation of text that is synchronized with video and audio tracks, inclusive of “closed captions”. Subtitles are presented for various purposes such as dialog language translation, localized titles, content description, and descriptive captions for hearing impaired viewers or presentation situations where audio is unavailable or inappropriate.

CEA708 (608) embedded in broadcast signals, as commonly used in North America, containing subtitles intended primarily for the hearing impaired viewers, is also supported,. See Embedded Captions.

A CMAF Presentation MAY offer the same subtitle content in more than one CMAF Track and format to reach Players with different decoding capabilities. Because each CMAF Track may have different qualities, it is recommended that manifests be able to indicate the preferred CMAF Tracks in a Selection Set.

11.2 WebVTT

WebVTT is a subtitle format that is commonly used to render subtitles in web browsers [VTT]. It can also be used to render subtitles in other types of media presentation applications.

In CMAF, WebVTT documents are encapsulated in CMAF Media Segments contained in a CMAF Track. WebVTT Segments may be sequentially downloaded, synchronized, and presented the same as audio and video CMAF Segments. A WebVTT document can be contained in a single Segment that spans the entire duration of a prerecorded Presentation, or span Segments with durations similar to audio and video, which is necessary in the case of low latency live streaming. The encapsulation of WebVTT documents in ISO Base Media movie fragments and tracks is defined in MPEG-4 Part 30 [ISOTXT].

WebVTT subtitle tracks are defined using a track handler_type of `'text'` with a codingname of `'wvtt'`. WebVTT subtitle tracks store samples corresponding to zero or more simultaneously presented WebVTT cues in movie fragments and CMAF Segments that span variable durations on the Track timeline. WebVTT subtitle tracks synchronize with other tracks selected for presentation using regular CMAF track synchronization methods based on ISO Base Media File Format and a Presentation Description Manifest. No sample shall contain a WebVTT cue with a duration that crosses a CMAF Segment boundary (i.e. which starts before the Segment starts and/or ends after the Segment ends); the format includes support for splitting cues into multiple samples to avoid this case.

Note: Forced titles are supported in TTML/IMSC1, but not VTT. A workaround is to create two VTT Tracks, one with only the forced titles, and the other with both forced and regular subtitles. A player would then play the forced-only track by default, and switch to the forced + subtitles track when subtitles are selected.

11.3 IMSC Text and Image Tracks

11.3.1 General

TTML Profiles for Internet Media Subtitles and Captions 1.0 (IMSC1) [IMSC] is an application of the Timed Text Markup Language (TTML) for subtitle and caption delivery. IMSC1 defines a text-only profile and an image-only profile.

An IMSC1 Text Track is a CMAF Track that conforms to the provisions of Sections 11.3.2 and 11.3.3.

An IMSC1 Image Track is a CMAF Track that conforms to the provisions of Sections 11.3.2 and 11.3.4.

11.3.2 Common Constraints

The CMAF Track SHALL conform to ISO/IEC 14496-30 [ISOTXT].

The `namespace` field of the `XMLSubtitleSampleEntry` box SHALL contain one instance of the string "`http://www.w3.org/ns/ttml`".

11.3.3 IMSC1 Text Track Constraints

All subtitle samples of the CMAF Track SHALL conform to the Text Profile specified in [IMSC].

The `schema_location` field of the `XMLSubtitleSampleEntry` box SHALL contain one instance of the string "`http://www.w3.org/ns/ttml/profile/imsc1/text`".

The media type of the subtitle samples is "`application/ttml+xml`" and SHALL include the `codecs` parameter `'im1t'` that signals that an IMSC1 Text processor is required, as specified in [TTMLREG].

As described in Annex I.2 of [IMSC], a document that conforms to the IMSC Text Profile also generally conforms to EBU-TT-D [EBUTTD].

EXAMPLE: The Media Type "application/ttml+xml;codecs=im1t|etd1" signals an IMSC1 Track that also conforms to [EBUTTD].

11.3.4 IMSC1 Image Tracks Constraints

All subtitle samples of the CMAF Track SHALL conform to the Image Profile specified in [IMSC].

The `schema_location` field of the `XMLSubtitleSampleEntry` box SHALL contain one instance of the string "`http://www.w3.org/ns/ttml/profile/imsc1/image`".

The media type of the subtitle samples is "`application/ttml+xml`" and SHALL include the `codecs` parameter `'im1i'` that signals that an IMSC1 Image processor is required, as specified in [TTMLREG].

EXAMPLE: The Media Type "application/ttml+xml;codecs=im1i" signals an IMSC1 Image Track.

Each `smpte:backgroundImage` attribute SHALL be a URN as specified in Section 5.6 of ISO/IEC 14496-30 that MAY conform to the following `bg-image-urn` syntax expressed using [ABNF]:

```
bg-image-urn = "urn:mpeg:14496-30:subs:" 1*DIGIT
```

The `auxiliary_mime_types` field of the `XMLSubtitleSampleEntry` box includes one instance of the media type "`image/png`" if any image is used.

11.4 CEA-608 and CEA-708

CEA-608 and CEA-708 are formats developed for delivering closed captions for accessibility purposes for broadcast television in North America. These formats can also be used to deliver captions in other scenarios. Closed captions can be embedded in the video elementary stream's SEI messages. Please refer to section 9.5 for the format of CEA-608/708 in SEI Messages.

11.5 Metadata for Subtitles

Text tracks SHOULD normally be labeled with their Role. See section [7.5.3]. Text tracks SHOULD also be tagged with their language. See Section [7.5.5].

Annex A.

CMAF Presentation and Media Profiles

A.1. CMAF Presentation Profiles

A.1.1. General

CMAF Presentation Profiles SHALL define Required and conditionally Required Media Profiles of CMAF Tracks that SHALL be available in all conforming CMAF Presentations. Required Media Profiles are a minimum constraint. Presentations containing the Required CMAF Media Profiles MAY also include additional CMAF Media Profiles and remain conformant.

The CMAF specification defines one CMAF Presentation Profile in Annex A.1.2, and an associated MPEG registered CMAF Presentation Profile Identifier.

CMAF Presentation Profiles MAY be specified by other specifications than CMAF, and other entities than MPEG. Externally specified CMAF Presentation Profiles SHALL be identified by a URL in the namespace of the specifying entity. The URL SHOULD resolve to a specification document that defines the Required Media Profiles of the Presentation Profile.

CMAF Media Profiles are considered optional by default in CMAF Presentation Profiles where they are not Required or conditionally Required.

CMAF Media Profiles SHALL conform to sections 7, 8, 9.2.1, 9.2.2, 10.2, 10.3, and 10.4.

CMAF Media Profiles MAY specify Switching Set constraints and one or more associated Adaptive Switching Process intended to result in seamless adaptive switching between multiple CMAF Tracks in a Switching Set conforming to those constraints. If a Media

Profile does not define adaptive switching, then only a single CMAF Track of that Media Profile SHALL be contained in each Switching Set.

If a CMAF Media Profile specifies Switching Set constraints for an Adaptive Switching Process, the Media Profile SHALL specify an identifier, similar to those in section 9.2.3.5 to identify those constraints in manifests.

All CMAF Media Profiles SHALL specify a compatibility brand and register that brand at www.mp4ra.org and SHALL reference to the Media Profile specification defining codec constraints, CMAF Track format, and optional Switching Set constraints and associated Adaptive Switching Process, as specified above.

CMAF Media Profiles SHALL be indicated by the presence of the Media Profile brand in the `compatible_brands` of the CMAF Header's File Type box.

Media Profiles MAY define Switching Set Profile encoding and seamless switching constraints, and an associated Switching Set Profile ID. A Switching Set Profile SHOULD also specify the associated processing model for seamless switching, for instance if reinitialization with the CMAF Header is required on each track switch or if a single initialization of the Switching Set is sufficient.

It is recommended that each Switching Set include Tracks conforming to one CMAF Media Profile so that a Player compatible with that CMAF Media Profile can automatically switch between all the contained CMAF Tracks once a Switching Set is selected by a Player. A CMAF Switching Set that combines multiple CMAF Media Profiles requires additional logic to determine if a Player can switch up to a higher CMAF Media Profile during adaptive streaming. Switching Sets of different CMAF Media Profiles often also differ regarding the need for different encryption keys, DRM licenses, output protection, and authorization, e.g. a different price and license for a HD and UHD.

Switching Set Media Profiles, encoding parameters, encryption, and licensing SHALL be constrained so that a single license acquisition is sufficient to present all the Tracks in the Switching Set. For instance, SD content may be available over analog display connections, HD may require HDMI, and UHD may require HDCP 2.2 over HDMI to reach an external display. These output controls are expressed in DRM licenses and enforced by DRM systems probably not exposed to the playback application, so they need to be constrained by content authoring to result in automatic seamless switching behavior.

A.1.2. The CMFHD Presentation Profile

The CMFHD Presentation Profile is intended to provide basic interoperability on the widest range of Internet video devices in use today. Other CMAF Media Profiles that are not Required in CMFHD, are considered optional.

Requirements of CMAF Presentation Profile CMFHD.

- Presentation Profile ID= “**urn:mpeg:cmaf:presentation_profile:cmfhd:2016**”
- If containing video, SHALL include at least one Switching Set constrained to the `'cfhd'` Media Profile in A.2
- If containing audio, SHALL include at least one audio Switching Set constrained to the `'caac'` Media Profile in A.3.
- If containing subtitles, SHALL include two Switching Sets for each language and role provided in subtitles; one constrained to the `'cwvt'` Media Profile in A.4, and a second constrained to the '`cttt’` Media Profile in A.4.
- If a CMAF Switching Set is encrypted, it SHALL use either ‘cenc’ or ‘cbcs’ Common Encryption scheme specified in Section 8.

Note: MPEG is considering requiring one encryption mode if that would meet the CMAF Requirements. National Bodies are requested to comment. National Bodies may also comment if they would prefer selecting an encryption mode per codec and Media Profile, or per Presentation Profile.

— CMAF Tracks containing optional CMAF Media Profiles in Switching Sets SHOULD be included in Selection Sets with Switching Sets containing Required Media Profiles.

It is strongly recommended, that encryption keys not be shared between audio and video Switching Sets. Audio decoding and decryption systems may have lower key security than video decoding systems so should not expose a key also used for video. Premium UHD/HDR content may require hardware security, watermarking, etc. only available on some devices and DRMs.

A.2. Video Media Profiles and Track Brands

The following Video Media Profiles indicate the track and media encoding constraints of a CMAF Track, and provide interoperability points between encoders and decoders. Video Media Formats SHALL conform to CMAF File and CMAF Track general video track format constraints specified in Sections 7, 8, and 9.

A CMAF Track SHALL include the Media Profile brand in the `compatible_ brands` table of the File Type box.

CMAF Tracks containing a Media Profile brand SHALL not exceed the limits specified in Table 9 — Video Media Profiles. Players SHOULD equal or exceed the limits of a Profile in order to reliably decode it.

CMAF Video Media Profiles listed in Table 9 SHALL conform to the CMAF Track Format Constraints for NAL Structured Video in section 9.2.4, and Switching Set Constraints and Adaptive Switching Processes for NAL Structured Video in section 9.2.4.

CMAF Media Profiles SHOULD be indicated by Manifests so that Players can identify the maximum decoding requirements of each CMAF Track and select a compatible CMAF Track with appropriate encoding quality for the Player’s available bandwidth, display capabilities, etc.

Switching Sets described by Manifests SHOULD indicate the highest Media Profile of the included CMAF Tracks, and MAY also indicate the codec parameters for each CMAF Track. A Player can initialize and play a subset of CMAF Tracks in a Switching Set, by selecting CMAF Tracks using codec parameters in the manifest or CMAF Header, and ignore other CMAF Tracks in the Switching Set that exceed its decoding capabilities.

The following table lists the maximum encoding parameters and minimum decoding parameters necessary for interoperability. Profiles include lower codec profiles and levels, and can encode video less than the maximum height or width, but SHALL NOT exceed the indicated Max Frame Height or Width, or Max Frame Rate. Picture aspect ratios SHALL be constrained to the limiting dimension. For instance, movies are typically produced in aspect ratios ranging from 1.85 to 2.4 so will be width limited, and will therefore have to be encoded with fewer lines when using a square sample aspect ratio, e.g. an aspect ratio of 2.4 results in 1920x800 limited to an HD frame.

A smaller frame size allows a higher frame rate since it is limited by the fill rate of the specified codec level, but it SHALL also limited by the Max Frame Rate, whichever is smaller.

For example, the HD video Media Profile allows 60Hz, which is possible at 1280x720 size, even though it is limited to 30Hz by fill rate at 1920x1080 size. SD Profile allows 854x480 at 60Hz. See Annex D. for detailed examples.

Table 9 — Video Media Profiles

Media Profile	Codec	Profile	Level	Color Coding	Transfer Characteristics	Max Frame Height	Max Frame Width	Max Frame Rate	CMAF File Brand
SD	AVC	High	3.1	BT.709 or BT.601	BT.709 or BT.601 OETF	576	854	60	`'cfsd'`
HD	AVC	High	4.0	BT.709	BT.709 OETF	1080	1920	60	`'cfhd'`
HDHF	AVC	High	4.2	BT.709	BT.709 OETF	1080	1920	60	`'chdf'`

[[Editor's note: MPEG requests comment on the parameters of the included Media Profiles, and recommendations for missing profiles that are application critical. HLG and SHVC Media Profiles have been proposed. The intent is to specify the minimum number of Media Profiles in the CMAF standard to encourage interoperability. Media Profiles and Presentation Profiles need not be specified in this document.]]

A.3. Audio Media Profiles and Track Brands

The following Audio Media Profiles are interoperability points between encoders and decoders that conform to the CMAF File and CMAF Track format and constraints specified in Sections 7, 8, and 10.

CMAF Tracks SHALL include the Media Profile brand in the `compatible_brands` of the File Type box.

For the AAC Core `‘caac’` Media Profile in Table 10, the audio track SHALL include stereo or mono audio as specified in section 10.5. Note that Switching Set constraints and an Adaptive Switching Process are not defined for this Media Profile. Only single CMAF Track Switching Sets are defined.

Table 10 — Audio Track Media Profiles

Media Profile	Codec and Profile	Number of channels	Max Sampling Rate	File Brand
AAC Core	AAC-LC, HE-AAC or HE-AAC v2	Mono or Stereo	48 kHz	`'caac'`

See Section 10 for general requirements and constraints on CMAF audio tracks.

A.4. Subtitle Media Profiles and Track Brands

CMAF specifies two subtitle profiles, WebVTT and TTML IMSC1, to provide interoperability between CMAF Presentations and Players.

Table 11 — Subtitle Track Profiles

Media Profile	Format	Notes	File Brand
WebVTT	WebVTT, Version 1.0	—pending normative reference https://w3c.github.io/webvtt/	`'cwvt'`
TTML IMSC1 Text	TTML W3C IMSC Version 1	Text Profile http://www.w3.org/TR/ttml-imsc1/#text-profile	`'cttt'`
TTML IMSC1 Image	TTML W3C IMSC Version 1	Image Profile http://www.w3.org/TR/ttml-imsc1/#image-profile	`‘ctti’`
CEA	CEA-608 and CEA-708	Caption data is embedded in SEI messages in video track; multiple closed caption streams may be present	`'ccea'`

See Section 11 for more details of these CMAF Track formats and subtitle formats.

Annex B.

HEVC Media Profile and Track Format

B.1. CMAF File

HEVC Tracks SHALL conform to sections 7, 8, and 9, except as provided in this Annex.

B.2. Video Tracks

B.2.1. Switching Set Constraints and Adaptive Switching Processes for CMAF Track and Media Profiles

B.2.1.1. General

Section 9.2.4, “CMAF Track Format Constraints for NAL Structured Video” applies except as follows.

Switching Sets containing a Media Profile listed in Annex B.4, conforming to CMAF Switching Set constraints, and HEVC Media Profiles with sample entry ‘hev1’ SHALL use the Single Initialization Adaptive Switching Process unless explicitly signaled using an indicator defined in 9.2.3.5.

'hev1' Coded Video Sequences, as specified in CMAF, contain the necessary Sequence Parameter Set and Picture Parameter Set NAL Units to signal decoding parameters changes allowed between CMF Tracks in the same Switching Set.

Switching Sets containing a Media Profile listed in Annex B.4, conforming to CMAF Switching Set constraints, and HEVC Media Profiles with sample entry ‘hvc1’ SHALL use the Multiple Initialization Adaptive Switching Process by default, unless identified as conformant to the Single Initialization Adaptive Switching Process by the identifier defined in 9.2.3.5.

B.2.1.2. Sample Description Box ('stsd')

The rest of section 9.2.4.2 shall apply except as follows:

A Decoder Configuration Record:

- SHALL signal other sequence parameter set and picture parameter set fields used by the video track as specified in [ISOVIDEO] Section 8.3.3.1.
- For a Visual Sample Entry with codingname `'hev1'`, SHALL contain a single parameter set intended only for decoder and display initialization. (Containing SPS and PPS NALs for AVC Video, or VPS, SPS, and PPS NALs for HEVC Video). The parameter set SHOULD have a parameter set index that will not be referenced by video slices for decoding. The SPS parameter values, profile_idc, level_idc, picture width, and picture height SHALL equal or exceed the values contained in SPS NALs in CMAF Fragments in the CMAF Track. Each video slice in the CMAF Track SHALL reference SPS and PPS NALs stored in that video CMAF Fragment, in the first sample of each Coded Video Sequence;
- For a Visual Sample Entry with codingname `'hvc1'`, SHALL contain one or more decoding parameter sets. (Containing VPS, SPS, and PPS NALs for HEVC Video). Each video sample in the CMAF Track SHALL reference this parameter set in the sample entry.
- MAY contain additional SEI NAL units to signal color encoding and rendering, such as `mastering_display_colour_volume`, SEI `payloadType` =137 [HEVC] Section D.2.27;

B.2.1.3. Picture Access Units

The rest of section 9.2.4.3 SHALL apply except as follows.

Picture Access Units SHALL conform to the requirements of a sample for the indicated format (`'hvc1'` or `'hev1'`) as specified in [ISOVIDEO]. Picture Access Units MAY be delimited by Access Unit Delimiter NALs. Each Access Unit is a sample stored in a Media Data Box ('mdat'), as specified in [ISOVIDEO].

CMAF Segments containing 'hev1' Picture Access Units SHALL contain all SPS and PPS NALs referenced by a Coded Video Sequence in the first Access Unit of that sequence, immediately following its first Access Unit Delimiter NAL (if any).

Access Units of type 'hev1'MAY retain filler data (NAL units or SEI messages) and SEI messages that would change hypothetical reference decoder bitstream conformance if such conformance is necessary, such as the case where bitstreams are to be repackaged and conformance tested in MPEG-2 Transport Streams.

B.3. Sample and CMAF Fragment Constraints

B.3.1. Storage of HEVC Elementary Streams

B.3.1.1. Conformance

HEVC video tracks SHALL comply with Section 8 of [ISOVIDEO].

B.3.1.2. Visual Sample Entry

The syntax and values for a visual sample entry SHALL conform to HEVCSampleEntry ('hvc1') or HEVCSampleEntry ('hev1') sample entries as defined in [ISOVIDEO].

B.3.1.3. HEVCDecoderConfigurationRecord

- For video that is captured or color graded with characteristics other than BT.709 defaults, one or more SEI NALs with additional transfer characteristics or color volume information SHOULD be stored in the HEVCDecoderConfigurationRecord to enable color and dynamic range calibration during decoder and display initialization. This includes the following SEI messages specified in Annex D of [HEVC]:

  - SEI payloadType 137, mastering_display_colour_volume
  - SEI payloadType 144, content_light_level_info

- CMAF Player display processors can use SEI color grading, color volume, and dynamic range information in the HEVCDecoderConfigurationRecord to calibrate conversion of decoded HEVC numeric values to color values that are appropriate for their display characteristics and viewing conditions. Track Switching Set encoding SHALL be constrained to allow a single initialization, i.e. color lookup table selection, for the entire Track Switching Set.
- A CMAF player can pass ST-2084 [ST2084] and other SEI message data to a display over HDMI, as specified in CEA 861.3 to enable color and transfer function calibration within a capable UHD/HDR display.

B.3.2. Constraints on HEVC Elementary Streams

B.3.2.1. Introduction

The following general constraints apply to all CMAF HEVC elementary streams. See Annex B.6 for Video Profile constraints on Tier, Profile, Level, and frame rates.

B.3.2.2. Picture type

All pictures SHALL be encoded as coded frames, and SHALL NOT be encoded as coded fields.

B.3.2.3. Video Parameter Sets (VPS)

Each HEVC video sample in the CMAF Track SHALL reference the VPS in the CMAF Header sample entry. VPS does not change within CMAF Tracks or between CMAF Tracks in a Switching Set. A CMAF HEVC track SHALL conform to [HEVC] with the following additional constraints:

- The following fields SHALL have pre-determined values as follows:
  - ▪ `general_progressive_source_flag` SHALL be set to 1
  - ▪ `general_frame_only_constraint_flag` SHALL be set to 1
  - ▪ `general_interlaced_source_flag` SHALL be set to 0
- The condition of the following fields SHALL NOT change throughout an HEVC elementary stream:
  - ▪ `general_profile_space`
  - ▪ `general_profile_idc`
  - ▪ `general_tier_flag`
  - ▪ `general_level_idc`

B.3.2.4. Sequence Parameter Sets (SPS)

B.3.2.4.1. SPS Fields

Sequence Parameter Set NAL Units that occur within a CMAF HEVC track SHALL conform to [HEVC] with the following additional constraints:

- The following fields SHALL have pre-determined values as follows:
  - ▪ `vui_parameters_present_flag` SHALL be set to 1

B.3.2.4.2. Visual Usability Information (VUI) Parameters

VUI parameters that occur within a CMAF HEVC track SHALL conform to [HEVC] with the following additional constraints:

- The following fields SHALL have pre-determined values as defined:
  - ▪ `aspect_ratio_info_present_flag` SHALL be set to 1
  - ▪ `chroma_loc_info_present_flag` SHALL be set to 0
  - ▪ `video_full_range_flag SHALL be set to` 0

– The following fields have the following values:

- ▪ `colour_description_present_flag` SHOULD be set to 1.

  - If colour_description_present_flag is set to 0, the following default video coding and mastering is assumed:
    - Video is encoded using the video parameters defined by [R709]; and
    - Video is appropriate for presentation on a display which uses the electro–optic transfer function specified in [R1886] with a peak luminance of 100 cd/m2 in a reference viewing environment that complies with [R2035].

  NOTE Per [HEVC], if the colour_description_present_flag is set to 1, the colour_primaries, transfer_characteristics and matrix_coefficients fields are present in the VUI.

  - overscan_info_present_flag SHALL be set to 0
  - overscan_appropriate SHALL NOT be present

- The values of the following fields SHALL NOT change throughout a CMAF Track and Switching Set:
  - low_delay_hrd_flag
  - colour_description_present_flag
  - colour_primaries, when present
  - transfer_characteristics, when present
  - matrix_coeffs, when present
- The values of the following fields SHOULD NOT change throughout a CMAF Track:
  - vui_time_scale
  - vui_num_units_in_tick

B.3.2.5. Maximum Bitrate

The maximum bitrate of HEVC elementary streams SHALL be calculated by implementation of the buffer and timing model defined in [HEVC] Annex C.

B.3.2.6. Frame rate in the Elementary Stream

The frame timing, including frame rate, is determined by the timing values provided in the track.

B.4. Video Codec Parameters

B.4.1. HEVC signaling of "`codecs`" parameters (Informational)

Presentation Applications SHOULD signal video codec profile and levels of each HEVC Track and Switching Set using parameters conforming to [RFC6381] and [ISOVIDEO] Annex E.

B.4.2. Encoding Overview

Video Tracks SHALL only encode spatial samples intended for presentation on all displays after the application of SPS cropping (if any). Padding such as letterbox and pillarbox bars SHOULD NOT be encoded.

The active image SHOULD be upper left justified, and only one row or one column of partially filled macroblocks encoded if the image height or width is not a multiple of the coding block size. Extra samples SHALL be cropped by setting SPS cropping parameters `conf_win_bottom_offset` or `conf_win_right_offset`.

The VUI parameter, `aspect_ratio_idc`, SHALL be present if square samples are not encoded, and SHOULD always be present to avoid incorrectly assuming the default value of 1.

Each device and display system is expected to frame the decoded and cropped video to its video display aperture using methods such as scaling, stretching, cropping, padding with letterbox or pillarbox bars, or framing in a window. Each visual Segment may be encoded with different vertical and horizontal sample counts, so devices need to scale each Segment to fit the same display aperture and maintain the correct picture aspect ratio to avoid scaling and placement errors during adaptive switching.

B.4.3. Video color and dynamic range mastering

The rest of section 9.4.3 SHALL apply except as follows.

Video MAY be graded for presentation on a display which uses an electro-optic transfer function not specified in [R1886] with a peak luminance greater than 100 cd/m2, in which case, the grading profile SHOULD be signaled by VUI and/or [ST2086] metadata in SEI messages contained in the sample entry `'hvcC'` box, or other mechanism defined in the video Media Profile.

B.5. Video Elementary Stream Embedded Captions

B.5.1. Carriage of CEA-608/708 in SEI messages

CEA 608/708 caption data [[CEA608], [CEA708] MAY be stored in SEI messages described as user data registered by Rec. ITU-T T.35, with SEI `payloadType = 4` and the registered identifier in the field `user_data_registered_itu_t_t35`. See [HEVC] Section D.2.6.

B.6. HEVC Media Profile and Track Brands

HEVC Media Profiles and Track Brands shall conform to Section A.2 except for Table 9

Table 12 — HEVC Video Media Profiles

Media Profile	Codec	Profile	Level	Color Coding	Transfer Characteristics	Max Frame Height	Max Frame Width	Max Frame Rate	CMAF File Brand
HHD8	HEVC	Main MainTier	4.1	BT.709	BT.709 OETF	1080	1920	60	`'chhd'`
HHD10	HEVC	Main10 MainTier	4.1	BT.709	BT.709 OETF	1080	1920	60	`'chh1'`
UHD8	HEVC	Main8	5.0	BT.709	BT.709 OETF	2160	3840	60	`'cud8'`

		MainTier							
UHD10	HEVC	Main10 MainTier 10-bit	5.1	BT.709, BT.2020	BT.709 OETF, BT.2020 OETF	2160	3840	60	'cud1'
HDR10	HEVC	Main10 MainTier 10-bit	5.1	BT-2020	ST-2084 EOTF	2160	3840	60	'chr1'

[[Editor's note: MPEG requests comment on the parameters of the included Media Profiles, and recommendations for missing profiles that are application critical. HLG and SHVC Media Profiles have been proposed. The intent is to specify the minimum number of Media Profiles in the CMAF standard to encourage interoperability. Media Profiles and Presentation Profiles need not be specified in this document.]]

B.7. HEVC Subsampling

For spatial subsampling, the following applies.

The width and height in active image spatial samples in a Coded Video Sequence is specified by the combination of the following sequence parameter set fields in the video elementary stream or sample entry. For HEVC:

- `pic_width_in_luma_samples` which defines the number of horizontal samples
- `pic_height_in_luma_samples` which defines the number of vertical samples
- `aspect_ratio_idc`, which defines the aspect ratio of each sample
- `conf_win_left_offset`, cropping parameter in SPS
- `conf_win_right_offset`, cropping parameter in SPS
- `conf_win_top_offset`, cropping parameter in SPS
- `conf_win_bottom_offset` cropping parameter in SPS

Also, see section Annex C.

For cropping to the active picture area, the following applies.

It is a normative requirement of HEVC that decoders perform cropping as signaled in Sequence Parameter Sets (SPS NALs):

- Coding Tree Units define the sample coding unit boundary (and are 64×64, 32×32, or 16×16 blocks).

Also, see section C.2.

Annex C.

(Informative) Subsampling of Tracks in Track Switching Sets

C.1. Spatial and Temporal Subsampling and Scaling

Spatial and temporal subsampling are encoding methods commonly used to adaptively reduce bitrate during adaptive streaming while optimizing video quality.

- Spatial subsampling encodes a fraction of the samples in the source video, e.g. 50% of the source's cropped vertical and horizontal sample count. See Section C.2.
- Temporal subsampling encodes a fraction of the frames in the source video, e.g. 50% of 24Hz to result in a 12Hz Track with half the samples, each of twice the duration, but the same CMAF Segment durations and start times. A Switching Set can include Tracks at different framerates, e.g. 15Hz and 30Hz (and perhaps 60Hz and 120Hz for UHD), but only if they are all exact submultiples of the lowest framerate.

In order for Track Switching Sets to be presented to viewers with minimal visible disturbance during adaptive bitrate switching, encoding and playback of all the alternative Tracks should adhere to the following best practices:

- Tracks in a Switching Set SHALL be alternative encodings of the same source content (same active video area, color encoding, source spatial sampling, etc.).
- All subsamples are exact sub-multiples of the number of spatial samples in the active area of the source video, i.e. no fractional or rounded subsamples after subsampling and possibly decoder cropping of partially empty macroblocks.
- The Player SHALL determine the display aperture i.e. the number of vertical and horizontal pixels rendered. A player may select a display aperture based on the picture aspect ratio of the Tracks in a Switching Set (`'tkhd' width/height`),

available display shapes and sizes, or output signal formats (such as HDMI EDIDs), as well as application and user preferences in framing the source aspect ratio within the application determined display area (full screen, windowed, letterboxed, portrait or landscape device orientation, etc.). Precise subsampling and rescaling is specified in this Annex to avoid visible changes in image size, position, or shape between alternative Tracks and Segments in a Switching Set. Also, different Switching Sets within a Selection Set SHOULD maintain exactly the same aspect ratio to prevent visible distortion when selecting e.g. different camera angles.

- Players SHALL scale the decoded and cropped samples in all Tracks in a Switching Set to the same display aperture.
- Players SHALL display all Tracks in a Switching Set at the same refresh rate. The Player SHALL determine a display refresh rate for the Switching Set and maintain that by refreshing each decoded image multiple times if necessary. Note: Framerate changes are impractical for many video interfaces to smoothly handle, and they may lack operating points for low framerates.

C.2. Spatial Sub-sampling

Spatial sub-sampling can be a helpful tool for improving coding efficiency of a video elementary stream. It is achieved by reducing the resolution of the coded picture relative to the source picture, while adjusting the sample aspect ratio if necessary to compensate for any change in the width to height sample ratio. For example, by reducing the horizontal resolution of the coded picture by 50% while increasing the sample aspect ratio from 1:1 to 2:1, the coded picture size is reduced by half. While this does not necessarily correspond to a 50% decrease in the amount of coded picture data, the decrease can nonetheless be significant.

The width and height in active image spatial samples in a Coded Video Sequence is specified by the combination of the following sequence parameter set fields in the video elementary stream or sample entry:

- [AVC]:
  - `pic_width_in_mbs_minus1` which defines the number of horizontal samples
  - `pic_height_in_map_units_minus1`, which defines the number of vertical samples
  - `aspect_ratio_idc`, which defines the aspect ratio of each sample
  - `frame_crop_left_offset`, cropping parameter in SPS
  - `frame_crop_right_offset`, cropping parameter in SPS
  - `frame_crop_top_offset`, cropping parameter in SPS
  - `frame_crop_bottom_offset`, cropping parameter in SPS
- [HEVC]:
  - `pic_width_in_luma_samples` which defines the number of horizontal samples
  - `pic_height_in_luma_samples` which defines the number of vertical samples
  - `aspect_ratio_idc`, which defines the aspect ratio of each sample
  - `conf_win_left_offset`, cropping parameter in SPS
  - `conf_win_right_offset`, cropping parameter in SPS
  - `conf_win_top_offset`, cropping parameter in SPS
  - `conf_win_bottom_offset` cropping parameter in SPS

The presentation size in the video track header box (`'tkhd'`) is defined in terms of square pixels (i.e. 1:1 sample aspect ratio) in the `width` and `height` fields of the Track Header Box (`'tkhd'`) of the video track. These values are used to determine the appropriate aspect ratio when displaying a Track. The `width` and `height` in the sample entry is the cropped sample count, which can be converted to square pixels by multiplying by the ratio indexed by `aspect_ratio_idc`.

All Tracks in a Track Switching Set SHALL have the same picture aspect ratio, equal to that of the active image area of the source. A Player may ignore the presentation size indicated in each Track, and scale all Segments to the Player-determined presentation aperture for that Switching Set.

C.3. Sub-sample Factor and Sample Aspect Ratio (SAR)

The original picture aspect ratio is conveyed in NAL-structured video by the sample aspect ratio of the source video as well as the cropped horizontal and vertical sample counts (SAR is the enumerated sample width to height ratio indexed by the SPS VUI parameter, `aspect_ratio_idc`). The picture aspect ratio is the cropped width divided by the cropped height times the SAR ratio.

Subsampling may change the encoded video sample aspect ratio when it changes the encoded sample counts, and has to be accurately encoded in the VUI information in Sequence Parameter Sets in each Fragment of every Track. MPEG decoder models include an undefined display processor that uses SPS NAL and VUI information to convert decoded 4:2:0 numerical values to a scaled image with the SAR, transfer function and color space indicated in VUI. This Annex further defines how multiple Tracks in a Track Set are scaled to a common display aperture with a fixed picture aspect ratio. The SAR and cropped sample counts SHALL produce the same picture aspect ratio as the source video.

The extent of sub-sampling applied to a Track can be characterized by a *sub-sample factor* in each of the horizontal and vertical dimensions, defined as follows:

- The *horizontal sub-sample factor* is defined as the ratio of the number of columns of the *luma* sample array in the source frame after sample cropping has been applied, divided by the number of columns of the *luma* sample array after sample cropping has been applied in subsampled Track. For example, a 1920 wide source image subsampled to 960 horizontal samples in a subsampled Track would have a horizontal sub-sample factor of 0.5

- The *vertical sub-sample factor* is defined as the ratio of the number of rows of the *luma* sample array in the source frame after sample cropping has been applied, divided by the number of rows of the *luma* sample array after sample cropping has been applied in a subsampled Track. For example, 1088 vertical samples cropped to 1080 in the source, subsampled to 544 samples cropped to 540 in a subsampled Track would have a vertical sub-sample factor of 0.5.

C.4. Examples of Single Dimension Sub-sampling

If a 1920 x 1080 square pixel (SAR 1:1) source picture is horizontally sub-sampled and encoded at a resolution of 1440 x 1080 (SAR 4:3), which corresponds to a 1920 x 1080 square pixel (SAR 1:1) picture, then the horizontal sub-sample factor is 1440 ÷ 1920 = 0.75, while the vertical sub-sample factor is 1.0 since there is no change in the vertical dimension.

Similarly, if a 1280 x 720 (SAR 1:1) source picture is vertically sub-sampled and encoded at a resolution of 1280 x 540 (SAR 3:4), which corresponds to a 1280 x 720 (SAR 1:1) picture size, then the horizontal sub-sample factor is 1.0 since the is no change in the horizontal dimension, and the vertical sub-sample factor is 540 ÷ 720 = 0.75.

C.5. Example of Mixed Sub-sampling

If a 1280 x 1080 (SAR 3:2) source picture is vertically sub-sampled and encoded at a resolution of 1280 x 540 (SAR 3:4), corresponding to a 1920 x 1080 square pixel (SAR 1:1) picture size, then the horizontal sub-sample factor is 1280 ÷ 1920 = $^{2}/_{3}$, and the vertical sub-sample factor is 540 ÷ 1080 = 0.5. To understand how this is an example of mixed sub-sampling, it is helpful to remember that the initial source picture resolution of

1280 x 1080 (SAR 3:2) can itself be thought of as having been horizontally sub-sampled from a higher resolution picture.

C.6. Cropping to Active Picture Area

CMAF players typically control adaptation of the source image to whatever display environment is currently in use. Source content like movies, old TV, and videos from cellphones, etc. have a variety of picture aspect ratios, as do video devices that include phones, tablets, computers, TVs, projectors, and wall displays. In some cases, display aspect ratios will change dynamically when a device like a tablet is rotated from vertical to horizontal orientation, or video is directed to a different display, and the video aperture may be switched from full screen to a window at any time. A Player conforming to the CMAF Application model SHALL adapt the source Active Image size and shape to its display environment by methods such as scaling, padding, cropping and stretching, and apply the same adaptation to every CMAF Segment in a Switching Set, adjusted for subsampling. Image padding added during production to adapt images to a particular TV aspect ratio like 4:3 or 16:9 (i.e. letter box bars or pillar box bars) SHOULD not be encoded.

Since the sub-sampled picture area might not always fall exactly on the sample coding unit boundary employed by the video elementary stream, additional cropping parameters are used to further define the dimensions of the coded picture. It is a normative requirement of AVC and HEVC that decoders perform cropping as signaled in Sequence Parameter Sets (SPS NALs).

- [AVC]:
  - “Macroblocks” define the sample coding unit boundary (and are 16x16 blocks)
- [HEVC]:
  - “Coding Tree Units” define the sample coding unit boundary (and are 64×64, 32×32, or 16×16 blocks). See section C.2.

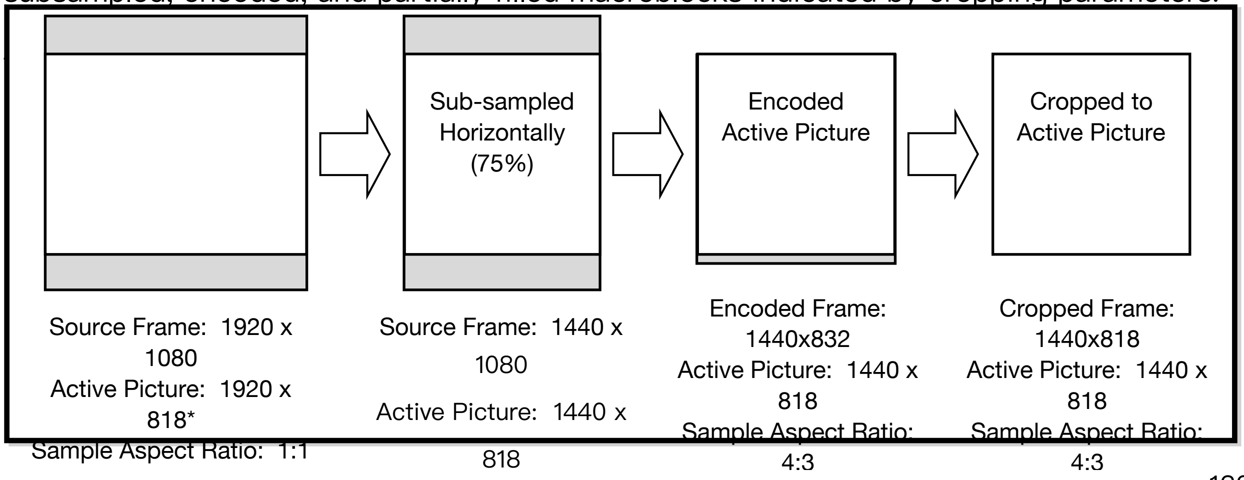
C.7. Relationship of Cropping and Sub-sampling

When spatial sub-sampling is applied, additional cropping parameters are often needed to compensate for the mismatch between the coded picture size and the macroblock ([AVC]) / coding tree unit ([HEVC]) boundaries. The specific relationship between theses mechanisms is defined as follows:

- Each picture is decoded using the coding parameters, including horizontal and vertical sample counts and cropping fields, defined in the sequence parameter set corresponding to that picture's Coded Video Sequence.
- The display aperture is determined by the CMAF Player, and each Segment scaled to fill that aperture using the same method to maintain registration, e.g. common sides, common top bottom, exact match, etc. For example, to output the video to an HDTV, the decoded image might need to be scaled to the display aperture width, then additional letterbox matting applied in order to match a valid HDMI television input format. A newer TV or projector might accept this picture aspect ratio directly without padding.

C.8. Example Encoding and Decoding Process

The following example shows a typical movie picture aspect ratio that was padded and encoded with letterbox bars to fit a 16:9 TV display. The active image is extracted, subsampled, encoded, and partially filled macroblocks indicated by cropping parameters.


Sub-sampled Horizontally (75%)
Encoded Active Picture
Cropped to Active Picture
Source Frame: 1920 x 1080
Active Picture: 1920 x 818*
Sample Aspect Ratio: 1:1
Source Frame: 1440 x 1080
Active Picture: 1440 x 818
Encoded Frame: 1440x832
Active Picture: 1440 x 818
Sample Aspect Ratio: 4:3
Cropped Frame: 1440x818
Active Picture: 1440 x 818
Sample Aspect Ratio: 4:3

* AVC cropping can only operate on even numbers of lines, requiring that the selected height be rounded up to 818 rather than 817.

Figure 15 — Example of encoding process of letterboxed source content

Figure 15 shows an example of the encoding process that can be applied. Table 13 shows the parameter values that could be used.

Table 13 — Example sub-sample and cropping values for Figure 15

Object	Field	Value
Picture Format	width	1920
Frame Size	height	1080
Sub-sample Factor	horizontal	0.75
	vertical	1.0
Track Header Box	width	1920
	height	818
[AVC] Parameter Values	chroma_format_idc	1 (4:2:0)
	aspect_ratio_idc	14 (4:3)
	pic_width_in_mbs_minus1	89

Object	Field	Value
	pic_height_in_map_units_minus1	51
	frame_cropping_flag	1
	frame_crop_left_offset	0
	frame_crop_right_offset	0
	frame_crop_top_offset	0
	frame_crop_bottom_offset	7
[HEVC] Parameter Values	chroma_format_idc	1 (4:2:0)
	MinCbSizeY	16
	log2_min_luma_coding_block_size_minus3	1
	aspect_ratio_idc	14 (4:3)
	pic_width_in_luma_samples	1440
	pic_height_in_luma_samples	832
	conformance_window_flag	1
	conf_win_left_offset	0
	conf_win_right_offset	0
	conf_win_top_offset	0
	conf_win_bottom_offset	7

Note 1: as `chroma_format_idc` is 1, `SubWidthC` and `SubWidthC` are set to 2 per [AVC] and [HEVC]. This results in a doubling of frame crop parameters (so `frame_crop_bottom_offset` and `conf_win_bottom_offset` both equate to 14 pixels in the above example).

Note 2: As [HEVC] `MinCbSizeY` is 16 and `log2_min_luma_coding_block_size_minus3` is 1, the Coding Tree Unit size is 16x16 (matching the [AVC] macroblock size of 16x16).

The decoding and display process for this content is illustrated in Figure 16, below. In this example, the decoded picture dimensions are 1440 x 818, one line larger than the original active picture area. This is due to a limitation in the cropping parameters to crop only even pairs of lines. Note: 816 lines or 51 macroblocks might be more practical, but makes a less informative example.

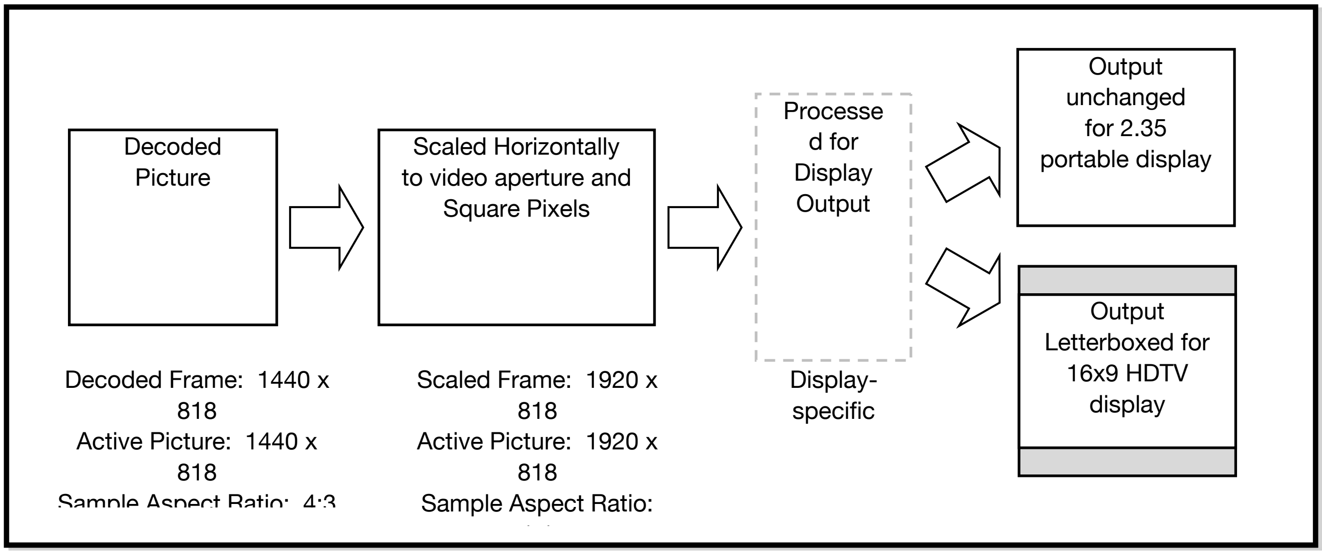

Decoded Picture
Scaled Horizontally to video aperture and Square Pixels
Processe d for Display Output
Output unchanged for 2.35 portable display
Output Letterboxed for 16x9 HDTV display
Decoded Frame: 1440 x 818
Active Picture: 1440 x 818
Sample Aspect Ratio: 4:3
Scaled Frame: 1920 x 818
Active Picture: 1920 x 818
Sample Aspect Ratio:
Display-specific

Figure 16 — Example of display process for letterboxed source content

Figure 17, below, illustrates what might happen when both sub-sampling and cropping are working in the same horizontal dimension. The original source picture content is first sub-sampled horizontally from a 1:1 sample aspect ratio at 1920 x 1080 to a sample aspect ratio of 4:3 at 1440 x 1080, then the 1080 x 1080-pixel active picture area of the sub-sampled image is encoded. However, the actual coded samples have a size of 1088 x 1088 samples due to the coding unit boundaries falling on even multiples of 16 pixels in this example — therefore, additional cropping parameters are provided in both horizontal and vertical dimensions.

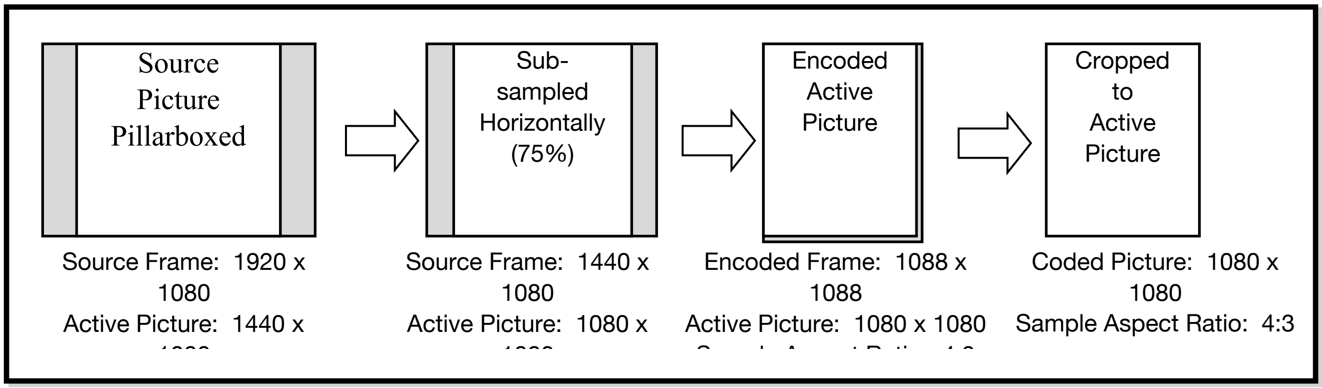

Source Picture Pillarboxed
Sub-sampled Horizontally (75%)
Encoded Active Picture
Cropped to Active Picture
Source Frame: 1920 x 1080
Active Picture: 1440 x
Source Frame: 1440 x 1080
Active Picture: 1080 x
Encoded Frame: 1088 x 1088
Active Picture: 1080 x 1080
Coded Picture: 1080 x 1080
Sample Aspect Ratio: 4:3

Figure 17 — Example of encoding process for pillarboxed source content

Table 14 lists the various parameters that might appear in the resulting file for this sample content.

Table 14 — Example sub-sample and cropping values for Figure 17

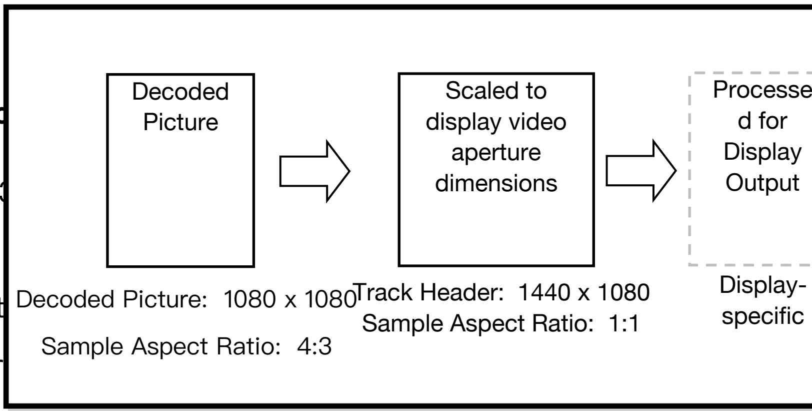
Object	Field	Value
Picture Format	width	1920
Frame Size	height	1080
Sub-sample Factor	horizontal	0.75
	vertical	1.0
Track Header Box	width	1440
	height	1080
[AVC] Parameter Values	chroma_format_idc	1 (4:2:0)
	aspect_ratio_idc	14 (4:3)
	pic_width_in_mbs_minus1	67
	pic_height_in_map_units_minus1	67
	frame_cropping_flag	1
	frame_crop_left_offset	0

Object	Field	Value
	frame_crop_right_offset	4
	frame_crop_top_offset	0
	frame_crop_bottom_offset	4
[HEVC] Parameter Values	chroma_format_idc	1 (4:2:0)
	MinCbSizeY	16
	log2_min_luma_coding_block_size_minus3	1
	aspect_ratio_idc	14 (4:3)
	pic_width_in_luma_samples	1088
	pic_height_in_luma_samples	1088
	conformance_window_flag	1
	conf_win_left_offset	0
	conf_win_right_offset	4
	conf_win_top_offset	0
	conf_win_bottom_offset	4

Note 1: as `chroma_format_idc` is 1, `SubWidthC` and `SubWidthC` are set to 2 per [AVC] and [HEVC]. This results in a doubling of frame crop parameters (so `frame_crop_bottom_offset` and `conf_win_bottom_offset` both equate to 14 pixels in the above example).

Note 2: As [HEVC] `MinCbSizeY` is 16 and `log2_min_luma_coding_block_size_minus3` is 1, the Coding Tree Unit size is 16x16 (matching the [AVC] macroblock size of 16x16).

The process for reconstructing the video for display is shown in Figure 18. As in the previous example, the decoded picture hasto be scaled back up to the original 1:1 sample aspect ratio.



Decoded Picture
Scaled to display video aperture dimensions
Processe d for Display Output
Decoded Picture: 1080 x 1080
Sample Aspect Ratio: 4:3
Track Header: 1440 x 1080
Sample Aspect Ratio: 1:1
Display-specific

Figure 18 — example of display proc

If this content was to be displayed on a 4: processing of the image would be necessary. on a 16:9 HDTV, it might be necessary for it right sides to reconstruct the original pillar displays properly. Or, a user might elect to zoom the image to full width and crop the top and bottom. Standard procedure for most TV programs shot on 1.85 aspect ratio film is to “protect” the 1.78 aspect ratio area (16:9) so that the top and bottom can be cropped when displayed full width on HDTVs without loss of significant content.

C.9. Example Track Switching Set

Example encoding “ladders” for adaptive bitrate and resolution Track Switching Sets are included in Annex D. .

The examples show worksheets that calculate a series of bitrates and sizes given a maximum desired bitrate and the percentage bitrate change between Tracks, referred to as “gradient”. The number of Tracks calculated is fixed at a large number, but only the necessary number of tracks need be encoded (starting from the top). Additional rows can be ignored.

Scaling and macroblock are calculated and compared to a codec level and profile to calculate frame rate, etc. Horizontal and vertical subsample ratios are checked for integer accuracy after cropping parameters are applied. A figure of merit (bits per picture element) is calculated to help manually adjust subsample ratios against the bitrate to maintain a consistent level of video quality at each bitrate (aside from resolution).

Note that max bitrate, gradient, the number of Tracks, and bit per PEL need to be adjusted to actual source material, deliver strategy, target networks, target devices, encoder efficiency, etc.

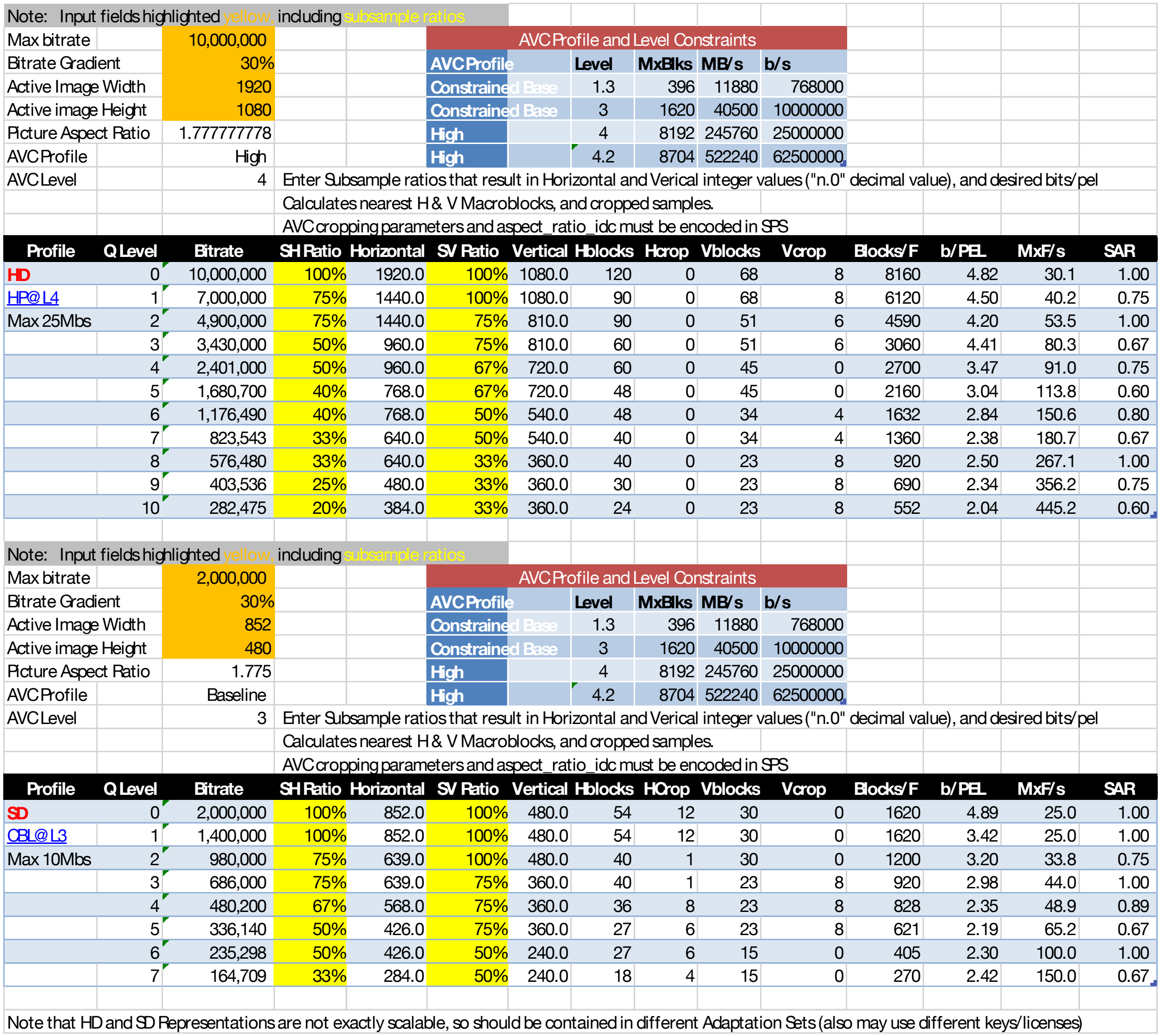
Some common source content aspect ratios and frame rates are selected for examples; including 16x9 HD, nearly 16x9 SD, a wide screen movie aspect ratio (2.4), 4x3, and frame sizes for 60Hz.

Annex D.

(Informative) Example Encoding Parameters for CMAF Switching Sets

D.1. 16x9 Aspect Ratio HD and SD

Table 15 — Example 16x9 Aspect Ratio HD and SD Switching Set Encoding Parameters



Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	10,000,000
Bitrate Gradient	30%
Active Image Width	1920
Active image Height	1080
Picture Aspect Ratio	1.777777778
AVC Profile	High
AVC Level	4

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	Hcrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
HD	0	10,000,000	100%	1920.0	100%	1080.0	120	0	68	8	8160	4.82	30.1	1.00
HP@L4	1	7,000,000	75%	1440.0	100%	1080.0	90	0	68	8	6120	4.50	40.2	0.75
Max 25Mbs	2	4,900,000	75%	1440.0	75%	810.0	90	0	51	6	4590	4.20	53.5	1.00
	3	3,430,000	50%	960.0	75%	810.0	60	0	51	6	3060	4.41	80.3	0.67
	4	2,401,000	50%	960.0	67%	720.0	60	0	45	0	2700	3.47	91.0	0.75
	5	1,680,700	40%	768.0	67%	720.0	48	0	45	0	2160	3.04	113.8	0.60
	6	1,176,490	40%	768.0	50%	540.0	48	0	34	4	1632	2.84	150.6	0.80
	7	823,543	33%	640.0	50%	540.0	40	0	34	4	1360	2.38	180.7	0.67
	8	576,480	33%	640.0	33%	360.0	40	0	23	8	920	2.50	267.1	1.00
	9	403,536	25%	480.0	33%	360.0	30	0	23	8	690	2.34	356.2	0.75
	10	282,475	20%	384.0	33%	360.0	24	0	23	8	552	2.04	445.2	0.60

Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	2,000,000
Bitrate Gradient	30%
Active Image Width	852
Active image Height	480
Picture Aspect Ratio	1.775
AVC Profile	Baseline
AVC Level	3

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

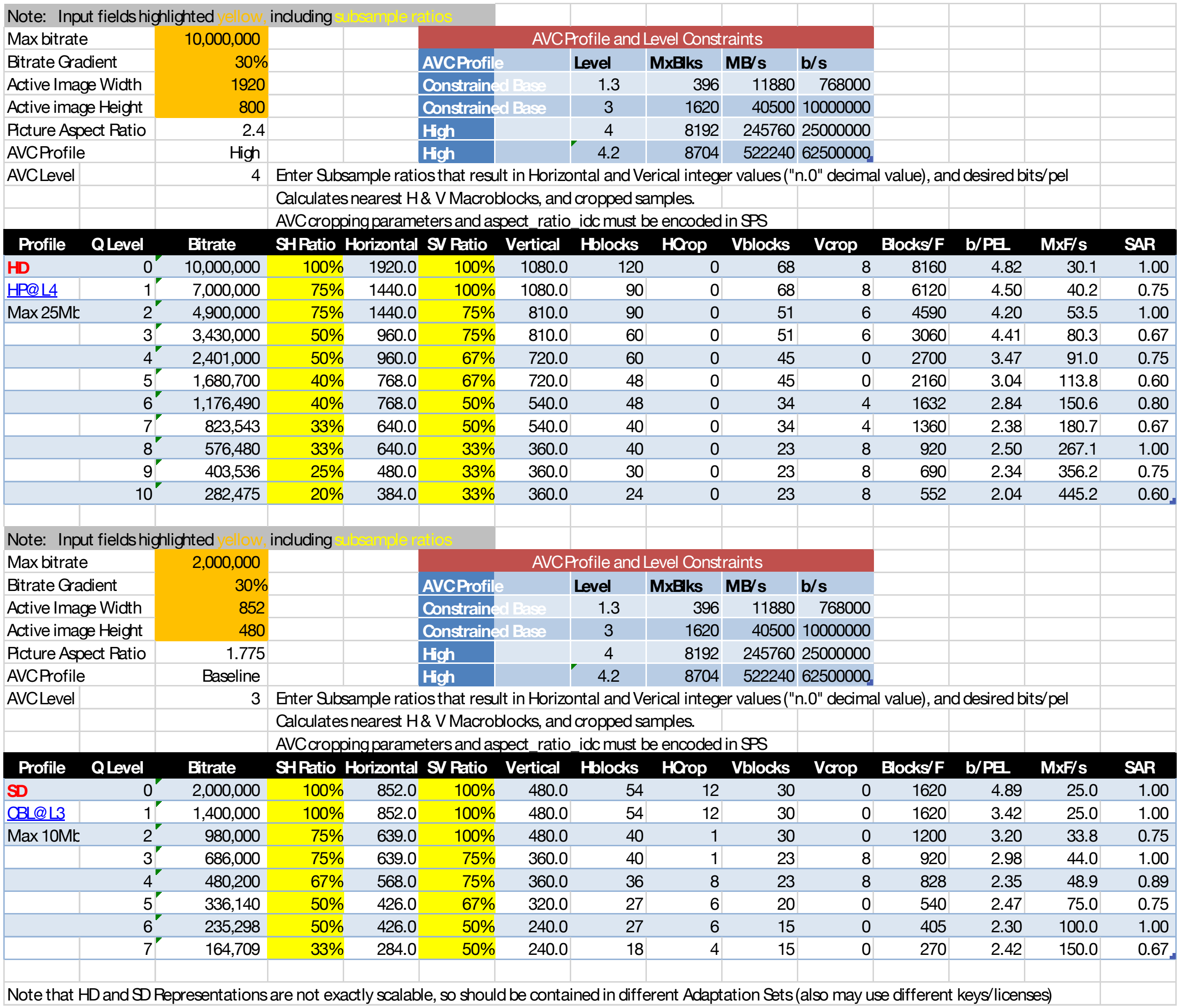
AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
SD	0	2,000,000	100%	852.0	100%	480.0	54	12	30	0	1620	4.89	25.0	1.00
CBL@L3	1	1,400,000	100%	852.0	100%	480.0	54	12	30	0	1620	3.42	25.0	1.00
Max 10Mbs	2	980,000	75%	639.0	100%	480.0	40	1	30	0	1200	3.20	33.8	0.75
	3	686,000	75%	639.0	75%	360.0	40	1	23	8	920	2.98	44.0	1.00
	4	480,200	67%	568.0	75%	360.0	36	8	23	8	828	2.35	48.9	0.89
	5	336,140	50%	426.0	75%	360.0	27	6	23	8	621	2.19	65.2	0.67
	6	235,298	50%	426.0	50%	240.0	27	6	15	0	405	2.30	100.0	1.00
	7	164,709	33%	284.0	50%	240.0	18	4	15	0	270	2.42	150.0	0.67

Note that HD and SD Representations are not exactly scalable, so should be contained in different Adaptation Sets (also may use different keys/licenses)

D.2. 2.4 Aspect Ratio HD and SD

Table 16 — 2.4 Aspect Ratio HD and SD Switching Set Encoding Parameters



Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	10,000,000
Bitrate Gradient	30%
Active Image Width	1920
Active image Height	800
Picture Aspect Ratio	2.4
AVC Profile	High
AVC Level	4

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
HD	0	10,000,000	100%	1920.0	100%	1080.0	120	0	68	8	8160	4.82	30.1	1.00
HP@L4	1	7,000,000	75%	1440.0	100%	1080.0	90	0	68	8	6120	4.50	40.2	0.75
Max 25Mb	2	4,900,000	75%	1440.0	75%	810.0	90	0	51	6	4590	4.20	53.5	1.00
	3	3,430,000	50%	960.0	75%	810.0	60	0	51	6	3060	4.41	80.3	0.67
	4	2,401,000	50%	960.0	67%	720.0	60	0	45	0	2700	3.47	91.0	0.75
	5	1,680,700	40%	768.0	67%	720.0	48	0	45	0	2160	3.04	113.8	0.60
	6	1,176,490	40%	768.0	50%	540.0	48	0	34	4	1632	2.84	150.6	0.80
	7	823,543	33%	640.0	50%	540.0	40	0	34	4	1360	2.38	180.7	0.67
	8	576,480	33%	640.0	33%	360.0	40	0	23	8	920	2.50	267.1	1.00
	9	403,536	25%	480.0	33%	360.0	30	0	23	8	690	2.34	356.2	0.75
	10	282,475	20%	384.0	33%	360.0	24	0	23	8	552	2.04	445.2	0.60

Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	2,000,000
Bitrate Gradient	30%
Active Image Width	852
Active image Height	480
Picture Aspect Ratio	1.775
AVC Profile	Baseline
AVC Level	3

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

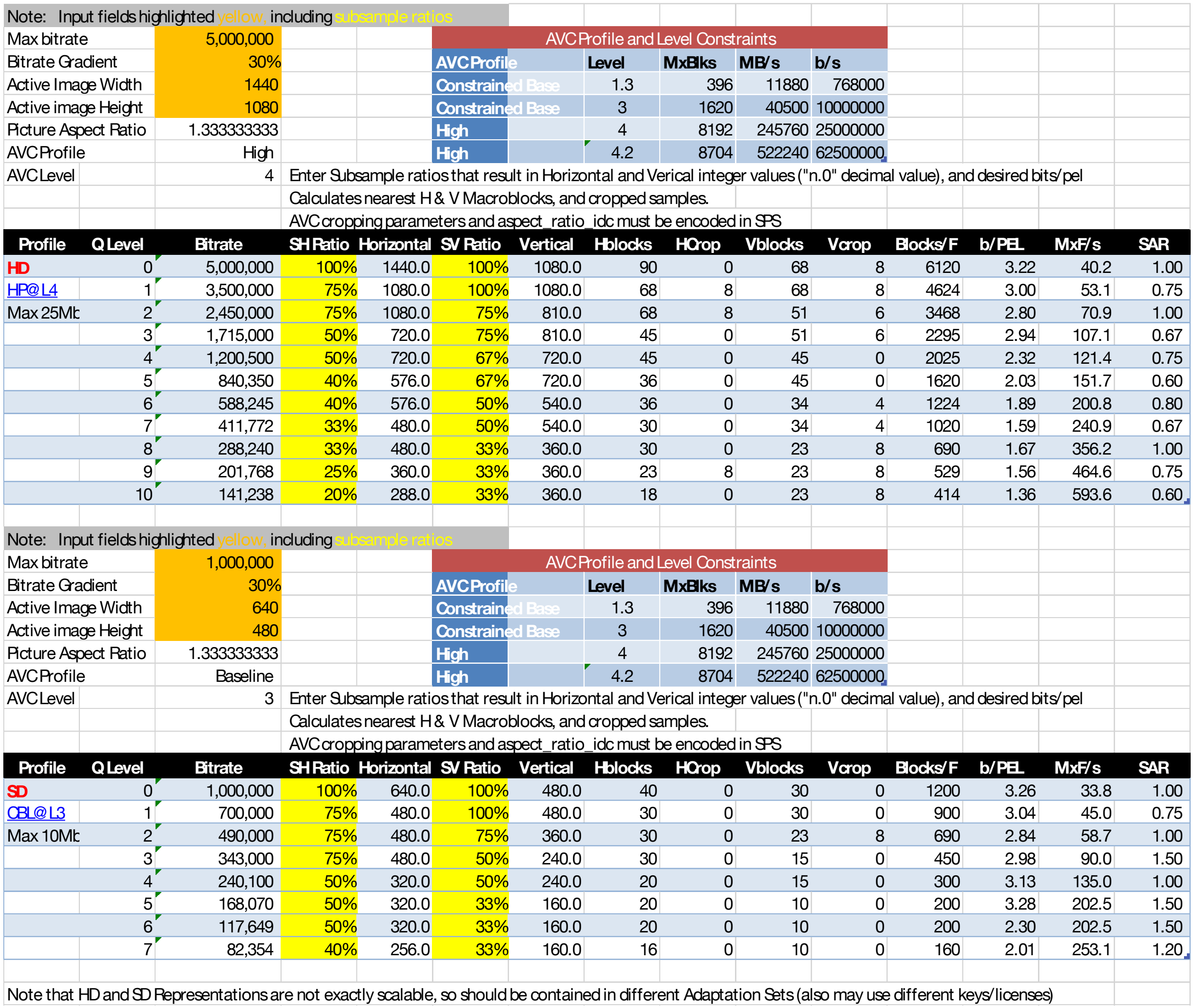
AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
SD	0	2,000,000	100%	852.0	100%	480.0	54	12	30	0	1620	4.89	25.0	1.00
CBL@L3	1	1,400,000	100%	852.0	100%	480.0	54	12	30	0	1620	3.42	25.0	1.00
Max 10Mb	2	980,000	75%	639.0	100%	480.0	40	1	30	0	1200	3.20	33.8	0.75
	3	686,000	75%	639.0	75%	360.0	40	1	23	8	920	2.98	44.0	1.00
	4	480,200	67%	568.0	75%	360.0	36	8	23	8	828	2.35	48.9	0.89
	5	336,140	50%	426.0	67%	320.0	27	6	20	0	540	2.47	75.0	0.75
	6	235,298	50%	426.0	50%	240.0	27	6	15	0	405	2.30	100.0	1.00
	7	164,709	33%	284.0	50%	240.0	18	4	15	0	270	2.42	150.0	0.67

Note that HD and SD Representations are not exactly scalable, so should be contained in different Adaptation Sets (also may use different keys/licenses)

D.3. 4x3 Aspect Ratio 1080 Line Source

Table 17 — Example 4x3 Aspect Ratio 1080 line Switching Set Encoding Parameters



Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	5,000,000
Bitrate Gradient	30%
Active Image Width	1440
Active image Height	1080
Picture Aspect Ratio	1.333333333
AVC Profile	High
AVC Level	4

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
HD	0	5,000,000	100%	1440.0	100%	1080.0	90	0	68	8	6120	3.22	40.2	1.00
HP@L4	1	3,500,000	75%	1080.0	100%	1080.0	68	8	68	8	4624	3.00	53.1	0.75
Max 25Mb	2	2,450,000	75%	1080.0	75%	810.0	68	8	51	6	3468	2.80	70.9	1.00
	3	1,715,000	50%	720.0	75%	810.0	45	0	51	6	2295	2.94	107.1	0.67
	4	1,200,500	50%	720.0	67%	720.0	45	0	45	0	2025	2.32	121.4	0.75
	5	840,350	40%	576.0	67%	720.0	36	0	45	0	1620	2.03	151.7	0.60
	6	588,245	40%	576.0	50%	540.0	36	0	34	4	1224	1.89	200.8	0.80
	7	411,772	33%	480.0	50%	540.0	30	0	34	4	1020	1.59	240.9	0.67
	8	288,240	33%	480.0	33%	360.0	30	0	23	8	690	1.67	356.2	1.00
	9	201,768	25%	360.0	33%	360.0	23	8	23	8	529	1.56	464.6	0.75
	10	141,238	20%	288.0	33%	360.0	18	0	23	8	414	1.36	593.6	0.60

Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	1,000,000
Bitrate Gradient	30%
Active Image Width	640
Active image Height	480
Picture Aspect Ratio	1.333333333
AVC Profile	Baseline
AVC Level	3

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

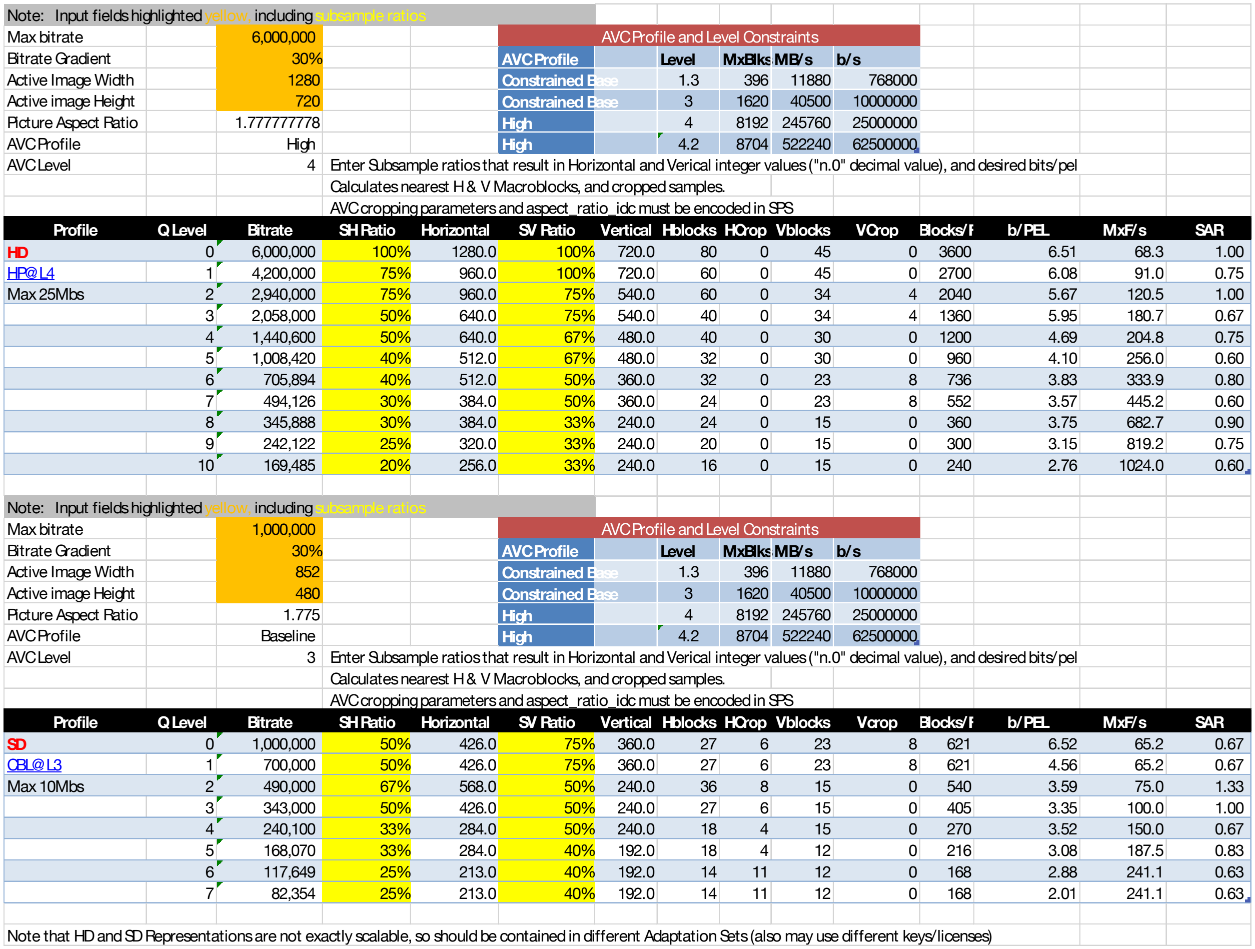
AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
SD	0	1,000,000	100%	640.0	100%	480.0	40	0	30	0	1200	3.26	33.8	1.00
CBL@L3	1	700,000	75%	480.0	100%	480.0	30	0	30	0	900	3.04	45.0	0.75
Max 10Mb	2	490,000	75%	480.0	75%	360.0	30	0	23	8	690	2.84	58.7	1.00
	3	343,000	75%	480.0	50%	240.0	30	0	15	0	450	2.98	90.0	1.50
	4	240,100	50%	320.0	50%	240.0	20	0	15	0	300	3.13	135.0	1.00
	5	168,070	50%	320.0	33%	160.0	20	0	10	0	200	3.28	202.5	1.50
	6	117,649	50%	320.0	33%	160.0	20	0	10	0	200	2.30	202.5	1.50
	7	82,354	40%	256.0	33%	160.0	16	0	10	0	160	2.01	253.1	1.20

Note that HD and SD Representations are not exactly scalable, so should be contained in different Adaptation Sets (also may use different keys/licenses)

D.4. 60Hz from 1280x720p60 Source

Table 18 — Example 60Hz 720 line Switching Set Encoding Parameters



Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	6,000,000
Bitrate Gradient	30%
Active Image Width	1280
Active image Height	720
Picture Aspect Ratio	1.777777778
AVC Profile	High
AVC Level	4

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	VCrop	Blocks/F	b/PEL	MxF/s	SAR
HD	0	6,000,000	100%	1280.0	100%	720.0	80	0	45	0	3600	6.51	68.3	1.00
HP@L4	1	4,200,000	75%	960.0	100%	720.0	60	0	45	0	2700	6.08	91.0	0.75
Max 25Mbs	2	2,940,000	75%	960.0	75%	540.0	60	0	34	4	2040	5.67	120.5	1.00
	3	2,058,000	50%	640.0	75%	540.0	40	0	34	4	1360	5.95	180.7	0.67
	4	1,440,600	50%	640.0	67%	480.0	40	0	30	0	1200	4.69	204.8	0.75
	5	1,008,420	40%	512.0	67%	480.0	32	0	30	0	960	4.10	256.0	0.60
	6	705,894	40%	512.0	50%	360.0	32	0	23	8	736	3.83	333.9	0.80
	7	494,126	30%	384.0	50%	360.0	24	0	23	8	552	3.57	445.2	0.60
	8	345,888	30%	384.0	33%	240.0	24	0	15	0	360	3.75	682.7	0.90
	9	242,122	25%	320.0	33%	240.0	20	0	15	0	300	3.15	819.2	0.75
	10	169,485	20%	256.0	33%	240.0	16	0	15	0	240	2.76	1024.0	0.60

Note: Input fields highlighted yellow, including subsample ratios

Max bitrate	1,000,000
Bitrate Gradient	30%
Active Image Width	852
Active image Height	480
Picture Aspect Ratio	1.775
AVC Profile	Baseline
AVC Level	3

AVC Profile and Level Constraints

AVC Profile	Level	MxBlks	MB/s	b/s
Constrained Base	1.3	396	11880	768000
Constrained Base	3	1620	40500	10000000
High	4	8192	245760	25000000
High	4.2	8704	522240	62500000

Enter Subsample ratios that result in Horizontal and Verical integer values ("n.0" decimal value), and desired bits/pel

Calculates nearest H & V Macroblocks, and cropped samples.

AVC cropping parameters and aspect_ratio_idc must be encoded in SPS

Profile	Q Level	Bitrate	SH Ratio	Horizontal	SV Ratio	Vertical	Hblocks	HCrop	Vblocks	Vcrop	Blocks/F	b/PEL	MxF/s	SAR
SD	0	1,000,000	50%	426.0	75%	360.0	27	6	23	8	621	6.52	65.2	0.67
CBL@L3	1	700,000	50%	426.0	75%	360.0	27	6	23	8	621	4.56	65.2	0.67
Max 10Mbs	2	490,000	67%	568.0	50%	240.0	36	8	15	0	540	3.59	75.0	1.33
	3	343,000	50%	426.0	50%	240.0	27	6	15	0	405	3.35	100.0	1.00
	4	240,100	33%	284.0	50%	240.0	18	4	15	0	270	3.52	150.0	0.67
	5	168,070	33%	284.0	40%	192.0	18	4	12	0	216	3.08	187.5	0.83
	6	117,649	25%	213.0	40%	192.0	14	11	12	0	168	2.88	241.1	0.63
	7	82,354	25%	213.0	40%	192.0	14	11	12	0	168	2.01	241.1	0.63

Note that HD and SD Representations are not exactly scalable, so should be contained in different Adaptation Sets (also may use different keys/licenses)

Annex E.

(Informative) Description and Delivery of CMAF content with MPEG-DASH

E.1. Introduction

The annex provides some general guidelines for description and delivery of CMAF content with MPEG DASH [DASH]. This annex only provides the general concepts and does not cover all possible mapping cases.

E.2. DASH description of CMAF content

CMAF is a content encoding format and it does not specify a mechanism for delivery. MPEG-DASH is a content description and delivery format that can describe any encoding format.

DASH defines a manifest format using an XML schema and semantics, as well as constraints on DASH segments that can be addressed for delivery, and synchronized for presentation. The DASH manifest is called a Media Presentation Description (MPD). An MPD XML document can be used by a Player in the CMAF application model to describe CMAF media objects and address CMAF Resources.

A DASH MPD provides a timing model for just in time delivery and presentation of CMAF Resources, defines various addressing schemes to address CMAF Resources, describes content with many attributes and descriptors, schedules presentation related events, and specifies manifest update methods which are needed for changeable presentations and realtime content encoding and delivery. CMAF Presentations can be described and delivered using MPEG-DASH standard.

Table 19 outlines the description of CMAF Presentations and Resources using a DASH MPD. Note that the table indicates how CMAF media objects can be described by DASH entities and doesn’t imply equivalency between two standards. For instance, DASH segment and Adaptation Set constraints are more general than CMAF Segments and Switching Sets, and every DASH segment is not necessarily compliant to CMAF file, Track, Resource, or Media Profile constraints.

Table 19 – Description of CMAF Media Objects by a DASH Media Presentation Description

CMAF Entity	**DASH Entity**
CMAF Presentation	A Period in a Media Presentation Description
CMAF Selection Set	A group of Adaptation Sets with the same value of @group
CMAF Switching Set	Adaptation Set
Single initialization Switching Set	Adaptation Set with `@bitstreamSwitching =True`*
CMAF Resource	Segment
CMAF Track File	On Demand profile segment*
CMAF Segment	Live profile segment*
CMAF Chunk	Broadcast TV profile segment*
CMAF Header	Initialization Segment*

CMAF Fragment	none

* With additional constraints

As is shown in Table 19:

1. CMAF Presentation may be described by a DASH Media Presentation Description (MPD). CMAF only defines one set of synchronized CMAF Tracks as a Presentation. DASH can describe a CMAF Presentation as one DASH period or multiple DASH periods. A DASH MPD can describe multiple CMAF Presentations as a sequence of DASH periods.
2. A CMAF Selection Set contains one or more synchronized Switching Sets of the same media type (audio, video, or subtitles), containing alternative content or encoding for the same Presentation over the same presentation time. Each Selection Set can be represented as a DASH Group. The `@group` attribute on an AdaptationSet element identifies adaptation sets that describe CMAF Switching Sets in a CMAF Selection Set identified by that `@group` value.
3. A CMAF Switching Set can be represented by a DASH Adaptation Set in the MPD. CMAF Switching sets are more constrained than the DASH Adaptation Sets. Note that a Switching Set may be divided into multiple Adaptation Sets, e.g. HD and SD Adaptation Sets.
4. A CMAF Single Initialized Switching Set can be represented by a DASH Adaptation Set with `@bitstreamSwitching=True.` The DASH player may switch between representations without downloading and using the initialization segment at the switching point for seamless switching. In addition, an AdaptationSet element can include a Descriptor whose schemeIdUri identifies the Media Profile specific encoding constraints and processing model for the CMAF Switching Set referenced. For instance, it can tell a player if it needs to process the CMAF Header only once, or on each Representation switch. More complicated Media Profiles may have more

complicated processing rules that allow switching on predicted frames, using additional layers, etc.

5. A CMAF Track can be represented as a DASH Representation with single segment. The DASH Extended On Demand profile uses such representation. In this case, the entire CMAF Track is addressed with single URL. A separate segment index file may also be provided for providing indexing information of the segment.
6. A CMAF Segment can be represented as a DASH Live profile's segment. Note that CMAF segment has additional constraints. Any of DASH addressing schemes can be used for these segments.
7. A CMAF Low Latency Chunk can be represented as a DASH's Broadcast TV profile segment format. The location of switching points can be signaled using DASH's **`Switching`** element. Additional random access point can be inserted during encoding of some of CMAF Fragments and be signaled using **`RandomAccess`** element in MPD.
8. A CMAF Header can be represented as a DASH Initialization Segment.

E.3. Static MPD for on-demand CMAF content

For on-demand content, all CMAF segments are available at same time and therefore static type MPD can be used. The `@availabilityStartTime` and `@availabilityEndTime` attributes may be used if the content is available and/or only offered during a wall clock time interval. All DASH's segment addressing schemes can be used for CMAF Resource addressing. Other DASH features, such as multiple periods, multiple **`BaseURLs`**, various descriptors can be used with CMAF content.

E.4. Dynamic delivery of CMAF Content

For live content and presentations that change during delivery, the CMAF Resources usually become available in real time. In this case, a dynamic type MPD is used for DASH delivery. The attributes such as `@availabilityStartTime`, `@mediaPresentationDuration`, `@timeShiftBufferDepth` and `@minBufferTime` are set according to availability of each segment at the media server. Other dynamic type MPD's attributes, such as MPD minimum update period or MPD validity expiration inband events may be used for MPD updates. Note that none of above information are present in CMAF segments or tracks and the author of the content need to present them along with CMAF content for MPD authoring.

Similar to on-demand content, all DASH's addressing schemes can be used for segment addressing. Other DASH features, such as multiple periods, multiple **`BaseURLs`**, various descriptors, and remote elements can be used with CMAF content.

Annex F.

(Informative) Event Messages

Event Messages are used to associate sparse metadata to presentation times in the CMAF Track, such as program segments, ratings, user interface data, advertisement identification, or availability splice points where video advertisements can be inserted. Event Message Boxes do not need to be repeated, but can be repeated, e.g. at ten seconds and five seconds before the event, in order to allow players tuning into a live stream to receive at least one copy of the Event Message.

Each Event Message Box (`'emsg'`) contains a `schemeIdUri` that functions as a URN message scheme identifier, and defines the payload of the message. Some schemes such as `urn:scte:scte35` are standardized by SDOs and consortia for interoperability, in this case, ad and program segmentation splicing and signaling. But, any application provider can define its own scheme using a URL they control, and may locate a specification at that URL if they choose. If a player has a handler for the scheme, that is the only component that needs to understand the payload and protocol. Message parsing and routing is generic.

Example uses include delivering sports scores, interactive components, presentation chaining, server redirection, sparse content description metadata, etc. An Event Message scheme can store any information, including a URL, in the message stored in the `message_data[]`section of the box. The message can contain a URL intended to trigger a Player request. That allows a server to determine a response that can be different based on clock time, the Player, the device, its location, etc. To reduce CMAF Track bitrate, it is advisable to deliver a URL when the response is large, or not relevant to a significant fraction of Players. Delivering data in the Event Message Box has the advantage of low latency and avoiding additional download requests, especially for live streaming.

Because Event Messages are contained in or attached to CMAF Segments that are already being requested, e.g. for audio playback, the Player's normal adaptive Segment downloading behavior will automatically transfer them. Event Messages can also be duplicated in a Manifest to provide random access and Event Message history for VOD playback, or live random access within a time shift buffer. Manifest signaling also allows Players to detect Event Messages by frequently requesting updated Manifests.

Event Message Boxes are added prior to a 'moof' box in a Fragment. They can be appended during encoding or delivery without needing to reformat the CMAF Fragment Movie Fragment Box or Media Data Box. Each event message box includes a time offset relative to the 'tfdt' in the following 'moof' box, and an optional duration. They are primarily intended to provide data and trigger action that is synchronized to a time or time range in the media. Event Messages can provide notification any time in advance of the event's presentation time so a player can complete actions, such as downloading signaled information or media, e.g. a video ad. It is advisable to insert Event Message boxes on all Tracks within a Selection Set (e.g. audio) so the message will be read regardless of the Track selected.

A Manifest can notify a Player of the Event Message schemes that will be sent in a Presentation so that Player can register a handler for each scheme_id_uri that can parse and process that Event Message scheme. Schemes can be specified for private or public use, as long as the scheme_id_uri is unique for that scope, and specified and managed accordingly.

Note: Additional text is pending to clarify the calculation of event time relative to BaseMediaDecodeTime in a CMAF Track, and how composition offsets and edit lists affect event presentation time. Additionally, the handling of 'emsg' boxes during defragmentation and refragmentation of a CMAF Track will be clarified.

Annex G.

(Informative) Error Handling for Missing Media

Valid CMAF Tracks do not allow media time discontinuities resulting from missing samples or Fragments. Gaps in decode time would result in audio video synchronization errors because the ISO Base Media File Format calculates decode and presentation times as the sum of prior sample durations in a track.

If audio or video frames are unavailable during recording, recorded samples may be extended in duration or filled with media data such as silent audio or repeated pictures.

Long gaps and gaps synchronized across all media streams can be recorded as two sequential, but non-continuous CMAF Tracks and Presentations that can be sequenced by a Manifest. A Manifest can sequence Tracks on a presentation timeline to remove a time gap, or maintain a presentation gap between two Tracks so they remain in sync with other Tracks.

Player handling of delivery errors that result in invalid CMAF Fragment sequences is out of scope of CMAF, but the Track Fragment Decode Time Box (`'tfdt'`) in each Fragment enables synchronization to Manifest presentation time following a gap in media delivery, trick play, etc.

A DASH manifest can represent missing Segments in a Switching Set as duration gaps in a SegmentTimeline element that contains a list of Segments and their durations, but can indicate ‘tfdt’ time of a Segment that follows a gap in the accumulated Segment durations.

It is recommended to provide either a manifest indication of a missing Segment, or a “dummy” replacement to avoid a 404 Not Found error when a missing Segment is requested. The ISOBMFF systems group is considering how best to construct a CMAF Fragment and Segment that contains the expected duration, but no media. Players would then be able to download a Segment, but would still need special error handling to continue presentation without media to decode.

The most preferred method to handle missing media at ingest, encode, or packaging is to provide substitute media such as a repeated video frame or slate, and silent audio so downloading and decoding can continue in players.