CTA WAVE Technical Working Group Note

Independent Document

Status:		Internal Draft		TWG Review	X	Public Review		Final

Title:	Byte Stream Format for CMAF
Source:	Adhoc Group for CMAF Byte Stream Format
Supporting Companies:	tbd
Background	Browser-based playback of media is one of the prime media consumption models. In many cases, the HTML-5 Video Element and the Media Source Extension are used to playback streaming content. For consistent implementations, in user agents, the MSE specification permits the definition of byte stream formats. CMAF has been developed in order to provide a consistent playback across in web environments. CMAF is close to the requirements of the ISO BMFF byte stream format already defined in W3C, but this document provides a direct CMAF byte stream format for improved interoperability.
Summary	This document defines segment formats for implementations that choose to support the CMAF ISO/IEC 23000-19 and CTA WAVE Content Specification for the W3C Media Source Extensions. The CMAF Byte Stream Format is a restriction of the ISO BMFF Byte Stream format. It defines the MIME-type parameters used to signal codecs, and provides the necessary format specific definitions for initialization segments, media segments, and random access points required by the byte stream formats section of the Media Source Extensions spec. This specification addresses the requirements here: https://www.w3.org/TR/media-source/#byte-stream-formats
Disclaimer:	This document is not yet final. It is provided for public review until the deadline mentioned below. If you have comments on the document, please submit comments at the github repository https://github.com/cta-wave/CMAF-Byte-Stream/issues by clicking here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/new?labels=Public-Review This document already contains some open questions and issues. Editor's Notes are provided with links in the document pointing to github issues. Please join the discussion there. Based on the received comments a final document will be created latest by the expected date below. This document will be provided to W3C for a joint publication. This document is copyrighted by the Consumer Technology Association (CTA)™ and may not be reproduced, in whole or part, without written permission. Federal copyright law prohibits unauthorized reproduction of this document by any means. Requests to reproduce text, data, charts, figures or other material should be made to the Consumer Technology Association (CTA)™.
Commenting Deadline:	August 31st, 2020
Expected Final Doc:	September 30th, 2020

Byte Stream Format for CMAF

1 INTRODUCTION

This specification defines segment formats for implementations that choose to support the CMAF ISO/IEC 23000-19 [CMAF] and CTA WAVE Content Specification [CTA WAVE CONTENT].

The CMAF Byte Stream Format is a restriction of the ISO BMFF Byte Stream format https://www.w3.org/TR/mse-byte-stream-format-isobmff/ [ISO-BMFF-BYTE-STREAM].

It defines the MIME-type parameters used to signal codecs, and provides the necessary format specific definitions for initialization segments, media segments, and random access points required by the byte stream formats section of the Media Source Extensions spec. It also provides references for sourcing AudioTrack, VideoTrack, and TextTrack attribute values from data in initialization segments.

This specification addresses the requirements here: https://www.w3.org/TR/media-source/#byte-stream-formats. The coverage of the requirements is documented in Annex C of this document.

2 MEDIA CAPABILITY DISCOVERY FOR CMAF CONTENT

2.1 MIME type parameters

MIME-types for this specification must conform to the rules outlined for "audio/mp4" and "video/mp4" in RFC 6381.

Examples for appropriate MIME-type parameters for CMAF Media Profiles are for example provided in the CTA WAVE content specification [CTA WAVE CONTENT], clause 4.

> NOTE: Implementations may only implement a subset of the codecs and profiles mentioned in CMAF or the CTA WAVE content specification.

2.2 Media Capabilities (Informative)

The Media Capabilities API [MEDIACAPABILITIES] provides an improved alternative to the `isTypeSupported()` API together with the MIME Type parameters introduced clause 2.1 for determining whether a given user agent is capable of encoding, decoding and rendering a piece of content.

To determine if a user-agent can decode a particular piece of content, the `mediaCapabilities.decodingInfo()` method is called. The method takes an instance of a `MediaDecodingConfiguration` object as an argument and returns as a Promise a `MediaCapabilitiesInfo` object.

The mapping of CTA WAVE defined Media Profiles to media capabilities is documented in CTA WAVE Device Playback Specification [CTA WAVE DEVICE], clause X.

Editor's Note: the above specification is not yet available, but planning has happened.

Note: Mapping of other CMAF media profiles to media capability APIs is for ffs.

3 INITIALIZATION SEGMENTS

A CMAF Byte Stream initialization segment is defined in this specification as a CMAF Header as defined in ISO/IEC 23000-19 [CMAF], clause 7.3.2.2.

1. The user agent must run the append error algorithm if any of the following conditions are met:
    a. An Initialization Segment does not conform to a CMAF header as defined in [CMAF].
    b. An Initialization Segment does not conform to the constraints in https://www.w3.org/TR/mse-byte-stream-format-isobmff/, clause 3.
    c. The user agent does not support the codec configurations stored in the sample entry.
2. The user agent must support setting the offset from media composition time to movie presentation time by handling an Edit Box ('edts') containing a single Edit List Box ('elst') that contains a single edit with media rate one. This edit may have a duration of 0 (indicating that it spans all subsequent media) or may have a non-zero duration (indicating the total duration of the movie including fragments).
3. Valid top-level boxes such as 'pdin', 'free', and 'sidx' are allowed to appear before the 'moov' box. These boxes must be accepted and ignored by the user agent and are not considered part of the initialization segment in this specification.
4. The user agent must source (i.e., extract) attribute values for id, kind, label and language for AudioTrack, VideoTrack and TextTrack objects as described in Annex A of this document.
5. The user agent must support codec configurations stored out-of-band in the sample entry, and for those CMAF profiles that allow codec configurations stored inband in the samples themselves, the user agent must support codec configurations stored inband.

    NOTE 1: The exact requirements for what needs to be supported, is defined individually for each sample entry. As an example, the CMAF specification in ISO/IEC 23000-19, clause 9.3.2.2 provides detailed requirements on permitted configurations for inband and out-of-band storage of sample entries for NALU-based video codecs.

    NOTE 2: In the ISO BMFF byte stream format , whereas codec configurations stored out-of-band are mandatorily to be supported, for codecs which allow codec configurations stored inband in the samples themselves, the user agent is only recommended to

support this. For CMAF, this permissible extension is not included, but any sample entry that is permitted in a CMAF profile, is required to be supported.

4 MEDIA SEGMENTS

A CMAF Byte Stream Media Segment is defined as a CMAF Chunk as defined in ISO/IEC 23000-19 [CMAF], clause 7.3.2.3.

NOTE 1: For the avoidance of confusion, the definition of a Media Segment in the ISO BMFF byte stream format is basically equivalent to what is used as a CMAF Chunk in this specification. Note that CMAF fragments and CMAF Segments are potential aggregation of CMAF Chunks.

NOTE 2: A CMAF Chunk conforms to a CMAF Fragment.

A CMAF Chunk is defined as an ISO BMFF Segment as defined in ISO/IEC 14496-12 with the restrictions in terms of both cardinality and ordinality as defined in Table 1.

Table 1: CMAF Chunk - Cardinality and Ordinality

NL 0	Cardinality	Specification	Constraints	Description
styp	0/1	ISO/IEC 14496-12	CMAF constraints	Segment Type Signalling compatibility to CMAF Chunk
prft	0/1	ISO/IEC 14496-12	CMAF constraints	Producer Reference Time
emsg	*	ISO/IEC 23009-1	CMAF constraints	Event Message
free	*	ISO/IEC 14496-12	none	free box
skip	*	ISO/IEC 14496-12	none	skip box
moof	1	ISO/IEC 14496-12	CMAF constraints	Movie Fragment box and the boxes it contains.
mdat	1	ISO/IEC 14496-12	CMAF constraints	Media Data container for media samples

In addition to an optional Segment Type Box (`styp`), a single Movie Fragment Box (moof), followed by a Media Data Boxes (`mdat`), CMAF chunks can contain an optional Producer Reference Time box (`prft`) and optional Event Message boxes (`emsg`) as well as other ISO BMFF boxes.

> Note 1: For `emsg` and `prft` boxes, the content of each of these boxes if placed at the correct positions according to Table 1, are expected to be surfaced to an application. Details on exact formats and APIs are for further study.

Valid top-level boxes defined in ISO/IEC 14496-12 other than `ftyp`, `moov`, as well as those listed in Table 1 are allowed to appear between the end of an initialization segment or media segment and before the beginning of a new media segment, but are expected to be ignored.

If the Segment Type Box (`styp`) is present and contains:

- `'cmff'`: an CMAF media segment as defined in this specification must conform to a CMAF Fragment (as defined in CMAF Spec)
- `'cmfs'`: an CMAF media segment as defined in this specification must conform to a CMAF Segment (as defined in CMAF Spec)

If the Segment Type Box (`styp`) is not present, the segment must conform to the brands listed in the File Type Box (**ftyp**) in the initialization segment.

The user agent must run the append error algorithm if:

a. A Media Segment does not conform to a CMAF Chunk.

b. A CMAF Chunk that is not compatible with latest appended Initialization Segment.

> Note 1: In the ISO BMFF byte stream format, 7 additional requirements are documented. These requirements are directly addressed by mandating the CMAF chunk structure as the media segment. For some details, please find comments in Annex B of this specification.

5 RANDOM ACCESS

A random access point as defined in this specification corresponds to a Stream Access Point of type 1 or 2 as defined in Annex I of ISO/IEC 14496-12.

6 COMPATIBILITY

According to the restrictions in ISO/IEC 23000-19 [CMAF] and this document, a user agent implementing the ISO BMFF Byte Stream format is capable to also process CMAF content consistently with the following exceptions:

- ftyp requirements from ISO BMFF are not inherited as they were considered unclear.
- The clarifications provided in Annex B of this document.

Editor's Note: other aspects need to be checked: https://github.com/cta-wave/CMAF-Byte-Stream/issues/14.

7 CONFORMANCE

As well as sections marked as non-normative, all authoring guidelines, diagrams, examples, and notes in this specification are non-normative. Everything else in this specification is normative.

The key words may, must, and should are to be interpreted as described in [RFC2119].

8 REFERENCES

8.1 Normative references

[CMAF]

ISO/IEC 23000-19, Information technology — Coding of audio-visual objects — Part 19: Common media application format (CMAF) for segmented media. https://www.iso.org/standard/71975.html

[CTA WAVE CONTENT]

Web Application Video Ecosystem (WAVE) Content Specification, Consumer Technology Association (CTA), 2019 Edition, December 2019.

[MEDIA-SOURCE]

Matthew Wolenetz; Jerry Smith; Mark Watson; Aaron Colwell; Adrian Bateman. W3C. Media Source Extensions. 5 July 2016. W3C Candidate Recommendation. URL: https://www.w3.org/TR/media-source/

[ISO-BMFF-BYTE-STREAM]

Matthew Wolenetz; Jerry Smith; Mark Watson; Aaron Colwell; Adrian Bateman. W3C. ISO BMFF Byte Stream Format. 6 October 2016. W3C Candidate Recommendation. URL: https://www.w3.org/TR/mse-byte-stream-format-isobmff/

[RFC2119]

S. Bradner. IETF. Key words for use in RFCs to Indicate Requirement Levels. March 1997. Best Current Practice. URL: https://tools.ietf.org/html/rfc2119

[CTA WAVE DEVCE]

CTA-5003, Web Application Video Ecosystem (WAVE) *Device Playback Specification*, Consumer Technology Association (CTA), December 2018. URL: https://shop.cta.tech/collections/standards/products/web-application-video-ecosystem-device-playback-capabilities.

8.2 Informative references

[INBANDTRACKS]

Bob Lund; Silvia Pfeiffer. W3C. Sourcing In-band Media Resource Tracks from Media Containers into HTML. URL: https://dev.w3.org/html5/html-sourcing-inband-tracks/

[MEDIACAPABILITIES]

Mounir Lamouri, Chris Cunningham, Vi Nguyen. W3C. Media Capabilities. URL: https://w3c.github.io/media-capabilities/

A SOURCING IN-BAND TRACKS FROM CMAF

A.1 Introduction

This Annex proposes text to be associated to the Sourcing In-Band Media Resource Tracks from Media Containers into HTML unofficial draft document as a new section that focuses on sourcing metadata from a CMAF conformant container. As CMAF is a restriction on the ISOBMFF Format, the existing section on MPEG-4 ISOBMFF is used as a document base.

A.2 MPEG-CMAF Inband Sources

MIME type/subtype: audio/mp4, video/mp4, application/mp4

[MPEGCMAF] defines a multimedia format which contains segmented media objects optimized for streaming delivery and decoding on end user devices in adaptive media presentations. The presentation is assumed to be contained within a manifest that is delivered to the end client player, though the exact format of the manifest is not specified.

The CMAF track format is derived from [ISOBMFF] and therefore shares MIME types with the former format. A user agent shall distinguish a CMAF conforming track from a general ISOBMFF track by the presence of one of the CMAF brands ("`cmfc`", "`cmf2`") in the FileTypeBox (`ftyp`) as either the major brand or a compatible brand.

A.3 Track Order

A track may be created for each CMAF switching set signaled by the manifest. The order of track depends on the application.

A.4 Determining the type of track

A user agent shall create an associated HTML track from a CMAF track based on the value of the `handler_type` field in the HandlerBox (`hdlr`) of the MediaBox (`mdia`) of the TrackBox of the CMAF Header associated to the switcing set:

- text track:
  - the `handler_type` value is "`subt`" or "`text`".
  - the `handler_type` value is "`vide`" and a CTA-608/708 caption service is encapsulated in the video track signalled by a "`ccea`" compatibility brand included in the FileTypeBox.
- video track: the `handler_type` value is "`vide`"
- audio track: the `handler_type` value is "`soun`"

A.5 Track Attributes for Sourced Text Tracks

Attribute	How to source its value
id	The track is: • A CTA-608 closed caption service: the string “cc” concatenated with the decimal representation of the channel_number • A CTA-708 closed caption service: the string “sn” concatenated with the decimal representation of the service_number field in the ‘Caption Channel Service Block’ • Otherwise, the decimal representation of the `track_ID` of a TrackHeaderBox (`tkhd`) in a TrackBox (`trak`). Editor's Note: Assuming true, but does track_ID uniqueness hold for CMAF ISOBMFF tracks? Please add your comments here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/15

Attribute	How to source its value

kind	The track:
	• Contains one or more KindBox (kind) in the UserDataBox (udta) of the TrackBox:
	  ○ With an instantiation where the `schemeURI` field is "`about:html-kind`": the `value` field should be directly used
	  ○ With an instantiation where the `schemeURI` field is "`urn:mpeg:dash:role:2011`":
	  ▪ "`captions`": if the `value` field is "`caption`"
	  ▪ "`subtitles`": if the `value` field is "`subtitle`"
	  ▪ "`metadata`": otherwise
	Editor's Note: Should "metadata" kind remain possible in a CMAF conforming presentation? Answer here will inform lower usage of metadata fallback as well. Please add your comments here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/16
	• Otherwise
	  ○ "`subtitles`": if the FileTypeBox contains a compatibility brand of "`cwvt`", "`im1t`", "`im1i`", "`im2t`", or "`im2i`"
	  ○ "`captions`": if the FileTypeBox contains a compatibility brand of "`ccea`"
	  ○ "`metadata`": otherwise
	Note: It is semantically correct for multiple KindBoxes to exist in the UserDataBox. Additional box instances may be used to add specificity to a role description or present a semantically equivalent role description for a different scheme. If multiple schemes are present, the schemes should be preferred in the order described above.

label	Content of the `name` field in the HandlerBox.
language	The track: • Represents a CTA-608/708 caption service: the empty string ("") Editor's Note: Carry over from ISOBMFF definition, service language may not match encapsulating track, do we have a better

Attribute	How to source its value
	identifier? Please add your comments here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/17 • Contains an ExtendedLanguageBox (`elng`) in the MediaHeaderBox (`mdia`) of the TrackBox: the content of the `extended_language` field. • Otherwise, the content of the `language` field in the MediaHeaderBox.
inBandMetadataTrackDispatchType	• kind is “metadata”[4] Editor's Note: If "metadata" is not a valid track type for CMAF this clause goes away and this field is always the empty string. Please add your comments here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/16 o If a XMLMetaDataSampleEntry box is present the concatenation of the string “metx”, a U+0020 SPACE character, and the value of the namespace field o If a TextMetaDataSampleEntry box is present the concatenation of the string “mett”, a U+0020 SPACE character, and the value of the mime_format field o Otherwise the empty string. • Otherwise the empty string.
mode	“disabled”

A.6 Track Attributes for sourced Audio and Video Tracks

Attribute	How to source its value
id	Decimal representation of the `track_ID` of a TrackHeaderBox (`tkhd`) in a TrackBox (`trak`). Editor's Note: Similar to text tracks, can track_ID be consistently used?. Please add your comments here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/15
kind	The track: • Contains one or more KindBox (`kind`) in the UserDataBox (`udta`) of the TrackBox:

	○ With one instantiation where the `schemeURI` field is "`about:html-kind`": the `value` field should be directly used ○ With one or more instantiations where the `schemeURI` field is "`urn:mpeg:dash:role:2011`": ▪ "`alternative`": if the `value` field(s) contain "`alternate`" but not also "`main`", "`commentary`", or "`dub`" ▪ "`descriptions`": if the `value` field(s) contain "`description`" and "`supplementary`" ▪ "`main`": if the `value` field(s) contain "`main`" and not also "`dub`" ▪ "`main-desc`": if the `value` field(s) contain "`main`" and "`description`" ▪ "`translation`": if the `value` field(s) contain "`dub`" and "`main`" ▪ "`commentary`": if the `value` field(s) contain "`commentary`" and not also "`main`" • Otherwise: the empty string ("") Note: It is semantically correct for multiple KindBoxes to exist in the UserDataBox. Additional box instances may be used to add specificity to a role description or present a semantically equivalent role description for a different scheme. If multiple schemes are present, the schemes should be preferred in the order described above. Editor's Note: This is not explicitly called out as a case in the CMAF specification, but this is a correct usage of the DASH role scheme in DASH manifests and would appear to be appropriate in CMAF tracks as well. Please add your comments here: https://github.com/cta-wave/CMAF-Byte-Stream/issues/18
label	Content of the `name` field in the HandlerBox.

language	The track: • Contains an ExtendedLanguageBox (`elng`) in the MediaHeaderBox (`mdia`) of the TrackBox: the content of the `extended_language` field. • Otherwise, the content of the `language` field in the MediaHeaderBox.

A.7 Mapping Text Track content into text track cues for MPEG-CMAF

[MPEGCMAF] text tracks may be in the WebVTT or TTML format [ISO14496-30].

[MPEGCMAF] text tracks carry WebVTT data if the FileTypeBox contains a compatibility brand of "`cwvt`".

[MPEGCMAF] text tracks carry TTML data if the FileTypeBox contains a compatibility brand of "`im1t`", "`im1i`", "`im2t`", or "`im2i`".

Exposure of both CMAF WebVTT and TTML formatted tracks should follow the same exposure formats as described for general [[#ISOBMFF-TT|ISOBMFF WebVTT and TTML formatted tracks].

B COMPARISON OF ISO BMFF AND CMAF BYTE STREAM FORMAT (INFORMATIVE)

For Media Segments of the ISO BMFF Format, the following 8 requirements are provided. An analysis on why this is not necessary for CMAF byte stream format is added in *italics* below:

1. A box or field in the Movie Fragment Box is encountered that violates the requirements mandated by the *major_brand* or one of the *compatible_brands* in the Segment Type Box in this media segment or the File Type Box in the initialization segment if a Segment Type Box is not present.

The requirements are expressed by the restrictions of cmfc, as well as the chunk formats in CMAF.

2. This media segment contains a Segment Type Box that is not compatible with the File Type Box in the initialization segment.

The requirements are expressed by the restrictions of cmfc as well as the chunk formats in CMAF.

3. The Movie Fragment Box does not contain at least one Track Fragment Box (**traf**).

This is a requirement of a CMAF chunk.

4. The Movie Fragment Box does not use movie-fragment relative addressing.

This is a requirement for a CMAF Chunk.

5. External data references are being used.

This is a requirement for a CMAF Chunk.

6. At least one Track Fragment Box does not contain a Track Fragment Decode Time Box (**tfdt**).

This is a requirement for a CMAF Chunk.

7. The Media Data Boxes do not contain all the samples referenced by the Track Fragment Run Boxes (**trun**) of the Movie Fragment Box.

This is a requirement for a CMAF Chunk.

8. Inband parameter sets are not present in the appropriate samples and parameter sets are not present in the last initialization segment appended.

This is a weak statement in the ISO BMFF byte stream format as it is not defined what Inband parameters are. In this document, any sample entry needs to be mandatorily supported.

C FULFILLMENT OF GENERAL REQUIREMENTS FOR BYTE STREAM FORMAT (INFORMATIVE)

This Annex provides an overview of the requirements for all byte stream format specifications as defined in https://www.w3.org/TR/media-source/#byte-stream-formats and a pointer an reference how this is fulfilled.

Requirement	Coverage in this document
A byte stream format specification MUST define initialization segments and media segments	Covered in clause 3 and clause 4, respectively.
A byte stream format SHOULD provide references for sourcing `AudioTrack`, `VideoTrack`, and `TextTrack` attribute values from data in initialization segments.	Covered in clause 3 and Annex A.
It MUST be possible to identify segment boundaries and segment type (initialization or media) by examining the byte stream alone.	Covered by ftyp and styp. For details see clause 3 and 4.

A user agent MUST run the append error algorithm when any of the following conditions are met. It is documented why this is not the case for CMAF content.

Requirement	CMAF Content
The number and type of tracks are not consistent.	If only content within on Switching Set is appended, this is consistent.
Codecs changes across initialization segments.	If only content within on Switching Set is appended, this is consistent.
For when any combination of an initialization segment and any contiguous sequence of media segments satisfies the following conditions:	
The number and type (audio, video, text, etc.) of all tracks in the media segments are not identified.	If only content within on Switching Set is appended, this is consistent.
The decoding capabilities needed to decode each track (i.e., codec and codec parameters) are not provided.	This is provided in the CMAF Header.

Requirement	CMAF Content
Encryption parameters necessary to decrypt the content (except the encryption key itself) are not provided for all encrypted tracks.	This is provided in the CMAF Header.
All information necessary to decode and render the earliest random access point in the sequence of media segments and all subsequence samples in the sequence (in presentation time) are not provided. This includes in particular: • Information that determines the intrinsic width and height of the video (specifically, this requires either the picture or pixel aspect ratio, together with the encoded resolution). • Information necessary to convert the video decoder output to a format suitable for display.	This is provided in the CMAF Header.
Information necessary to compute the global presentation timestamp of every sample in the sequence of media segments is not provided.	This is based on the late binding model of CMAF.

A user agent MUST support the following conditions. It is documented why this is the case for CMAF content.

Requirement	CMAF Content
Track IDs changing across initialization segments if the segments describes only one track of each type.	CMAF permits this.
Video frame size changes. The user agent MUST support seamless playback.	CMAF permits this
Audio channel count changes. The user agent MAY support this seamlessly and could trigger downmixing.	CMAF permits this
The following rules apply to all media segments within a byte stream. A user agent MUST:	
Map all timestamps to the same media timeline	This is provided by the combination of tfdt, composition offsets and edit lists.

Requirement	CMAF Content
Support seamless playback of media segments having a timestamp gap smaller than the audio frame size. User agents MUST NOT reflect these gaps in the `buffered` attribute.	CMAF content does not have this "problem".