|  |  |
| --- | --- |
|  |  |

Text

Description automatically generated ISO/IEC JTC 1/SC 29/WG 03 N1086

**ISO/IEC JTC 1/SC 29/WG 03**

**MPEG Systems   
Convenorship: KATS (Korea, Republic of)**

**Document type:** Output Document

**Title: Exploration on the alignment of ISOBMFF/DASH/CMAF terminology, concepts, and solutions**

**Status:** Approved

**Date of document:** 2024-10-27

**Source:** ISO/IEC JTC 1/SC 29/WG 03

**No. of pages:** 15 (with cover page)

**Email of Convenor:** young.L@samsung.com

**Committee URL:** <https://isotc.iso.org/livelink/livelink/open/jtc1sc29wg3>

**INTERNATIONAL ORGANISATION FOR STANDARDISATION**

**ORGANISATION INTERNATIONALE DE NORMALISATION**

**ISO/IEC JTC 1/SC 29/WG 03 MPEG SYSTEMS**

**ISO/IEC JTC 1/SC 29/WG 03 N** **1086**

**October 2023, Hannover, GE**

|  |  |
| --- | --- |
| **Source** | **MPEG Systems working group** |
| **Status** | **Output** |
| **Title** | **Exploration on the alignment of ISOBMFF/DASH/CMAF terminology, concepts, and solutions** |
| **Serial Number** | **23208** |

INTERNATIONAL STANDARD© ISO/IEC 2015 – All rights reservedISO/IEC 15444-12:2015(E) 63Part 12: ISO base media file formatInformation technology — JPEG 2000 image coding systemTechnologies de l'information — Codage des objets audiovisuels — Partie 12: Format ISO de base pour les fichiers médiasInformation technology — JPEG 2000 image coding system — Part 12: ISO base media file formatE2015-02-20(60) PublicationISO/IECISO/IEC J   International Standard 2015ISO/IEC 15444‑ISO/IEC 15444‑12ISO/IEC 15444-12  Coding of audio, picture, multimedia and hypermedia informationInformation technology 291 2見出し 2見出し 1    02 STD Version 2.1c260   4C:\Users\shinji\_w\AppData\Roaming\Microsoft\Templates\STD\w15177\_14496\_5th.-restyle-R1.doc

Contents Page

1 General considerations 2

1.1 Purpose of this document 2

1.2 Background 2

1.3 Processing Diagram 4

1.4 Requirements and scenarios 4

1.5 Previous work summary 5

2 Proposals 5

2.1 Signaling the CMAF switching sets and DASH adaptation sets in ISOBMFF 5

2.1.1 Use case 5

2.1.2 Signaling the CMAF switching sets and DASH adaptation sets in ISOBMFF (based on m65284 and m65071) 5

2.1.3 Comments during MPEG#144 6

2.2 Signaling DASH descriptors in ISOBMFF (based on m65349) 7

2.2.1 Use case 7

2.2.2 Identified Problems with TrackKindBox 7

2.2.3 Proposal 7

2.2.4 Comments during MPEG#144 8

# General considerations

## Purpose of this document

This exploration document and/or its technologies may eventually evolve into one or more of the following outcomes:

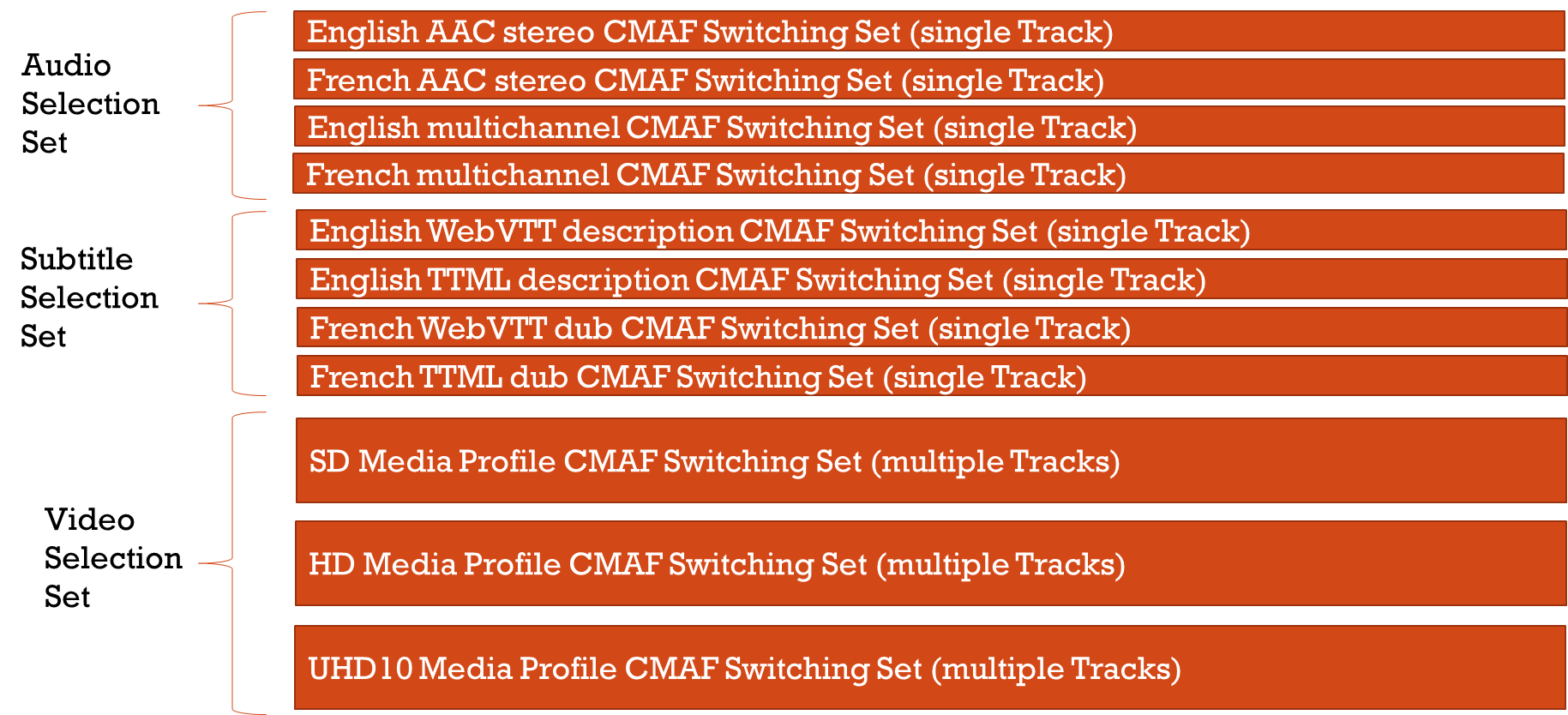
* Technical Report that describes the mapping between ISOBMFF, CMAF, and DASH terms and concepts.
* ISOBMFF extensions to indicate a track grouping that is equivalent to a CMAF Switching Set and a DASH Adaptation Set
* ISOBMFF extensions equivalent to indicating switching between DASH Adaptation Sets
* ISOBMFF extensions allowing hierarchical track grouping, for applications such as signaling preselections of DASH Adaptations Sets.

## Background

Some structural functionalities in DASH and ISO BMFF have not been fully harmonized. The file format typically assumes that all tracks are contained in a single file and a single movie header documents all tracks included in the file, but also provides the relationship of these tracks for presentation. A few file format functionality that are relevant include

* Each media stream is contained in a track specialized for that media type (audio, video, etc.), and is further parameterized by a sample entry.
* The sample entry
  + contains the ‘name’ of the exact media type (i.e., the type of the decoder needed to decode the stream) and any parameterization of that decoder needed.
  + The name also takes the form of a four-character code.
  + There are defined sample entry formats not only for MPEG-4 media but also for the media types used by other organizations using this file format family.
  + They are registered at the MP4 registration authority.
* Tracks (or sub-tracks) may be identified as alternatives to each other, and there is support for declarations to identify what aspect of the track can be used to determine which alternative to present, in the form of track selection data.
* Tracks may also be linked
* Tracks may be grouped
* Tracks may be encrypted
* Tracks have random access samples
* Tracks have assigned media-specific properties/annotation (codec, width, height, etc.)
* Relationships in tracks are expressed as
  + Track references: track N uses or refers to track(s) K (hint, chap, scale)
  + Track groups: Tracks in the same group share a common feature
  + Track Selection: Provides selection information for alternate tracks
    - New features have been added or about to be added, such as dependencies, Pre-selection, etc.

Some of the file format principles do not carry forward to CMAF or DASH, because CMAF and DASH heavily rely on the concept of late binding, i.e. each track is stored in a separate file. This is shown in the figure below.



In a streaming environment, this avoids combinatorial complexity or useless downloads, as clients only select the relevant tracks for the current situation, and only download these tracks and do synchronized playout. The HTML-5 MSE-based playback exactly permits this.

However, by doing so, each track gets its own ISO BMFF/MP4 file and all of a sudden all concepts of the file format are no longer applicable because

1. The requirements for the file to have unique track identifiers are not carried forward
2. Each file includes its movie header and the relationship between the tracks cannot be expressed

In a streaming environment, the information that is contained in a movie header for many tracks needs to be populated to the manifest such that the client can select the tracks based on this information.

In CMAF, some conceptual grouping was done, not relying on file format information, but defining its concepts

* CMAF Tracks
* CMAF Switching Sets
* CMAF Groups
* Depending on switching sets
* Aligned switching sets
* Etc.

For example, in CMAF it is not specified that all tracks in a CMAF switching track shall conform to a track group according to the ISO BMFF.

## Processing Diagram

Packaging for Streaming   
(parameters: segment duration, low latency, …)

Packaging for local playback   
(losing delivery aspects: segmentation, availability, …)

Different potential processes exist for converting ISO BMFF content into CMAF/DASH delivery and vice versa.

Processing use cases may for example be:

1. A content provider has generated CMAF content for a movie or a CMAF presentation. For storage efficiency reasons, the content provider wants to store the entire CMAF presentation in an ISO BMFF file format that is compatible with the existing file format. At the same time, the content provider wants to re-generate the CMAF Presentation when for example it is converted into a DASH/HLS streaming session.
2. A receiver wants to record a streamed DASH Presentation into an ISO BMFF file format compatible file with minimum changes.

Other cases may be defined.

## Requirements and scenarios

* It should be possible to store a CMAF presentation in a multi-track MP4 file and recover a CMAF presentation from this multi-track MP4 file.
* Using the DASH Profile for CMAF content it should be possible to generate a DASH Media Presentation (MPD and Segments) using such a recovered CMAF Presentation. Based on this, it should be possible to generate a DASH Media Presentation from a multi-track MP4 file.
* It should be possible to offer the same presentation options than the MPD offers in a local MP4, including bitrate variations (for debugging purposes but also battery efficiency use cases).

*We need to consider live cases, where you would only get the MP4 init segment (with multiple tracks in it), can you produce an MPD from that only? Not having the media segments you cannot know in advance if they are aligned and thus cannot produce the proper MPD grouping, without additional signaling. Hence, there needs to be a property assigned to multiple tracks that provides in the movie header the information that the tracks will be produced in an aligned fashion.*

## Previous work summary

Documents WG03N867 ([MDS22609](https://dms.mpeg.expert/doc_end_user/current_document.php?id=87755&id_meeting=194)) summarize the exploration before the MPEG#144 meeting.

This document is the result of the contributions to and discussions at the MPEG#144 meeting and only focuses on the results of the contributions and discussions during this meeting.

# Proposals

## Signaling the CMAF switching sets and DASH adaptation sets in ISOBMFF

### Use case

The use case allows signaling the properties of a group of the ISOBMFF tracks in an ISOBMFF file. The main common use cases are:

1. A CMAF switching set/DASH adaptation set consists of two or more tracks
2. A switchable group of CMAF switching sets/two or more DASH adaptation sets that are seamlessly switchable.
3. An alternate group of CMAF switching sets/ DASH adaptation sets in which the player selects one CMAF switching set/DASH adaptation set and plays the content. If the player decides to switch to a new switching set, the switching is not necessarily seamless.
4. Dependable CMAF switching sets/DASH adaptation sets where one CMAF switching set is not decodable without another CMAF switching set.

### Signaling the CMAF switching sets and DASH adaptation sets in ISOBMFF (based on m65284 and m65071)

# Syntax

aligned(8) class SwitchingGroupBox

extends EntityToGroupBox('swit', version=0, flags){

unsigned int(1) switchable\_flag;

unsigned int(1) timed\_aligned\_flag;

unsigned int(2) init\_type;

unsigned int(4) reserved;

if (init\_type == 0) {

unsigned int(32) principal;

}

if (flags & 0x001000) utf8string tag;

if (flags & 0x002000) utf8string structural\_brand;

if (flags & 0x004000) utf8string mediaprofile\_brand;

extended\_language\_box; //optional

Box boxes[];

}

# Semantics

switchable\_flag equal 1 indicates that the entities of this group are switchable with each other, i.e. they provide the perceptual same content. Otherwise, the entities are not perceptually the same content.

time\_aligned\_flag equal 1 indicates the tracks which directly or indirectly belong to this group are timed aligned at the substructural level (e.g. fragment/segment/subsegment). The nature of the substructure and the definition of time alignment are defined by the track types.

init\_type with the following values:

* 0: The initialization segment from the principal entity group or track shall be used to initialize a decoder for decoding any track that directly or indirectly belongs to this group. The principal entity\_id is signaled via the principal field described below.
* 1: Every track directly or indirectly belonging to this group can be used to initialize a decoder for decoding any track that directly or indirectly belongs to this group.
* 2: Every track which directly or indirectly belongs to this group contains its own initialization segment.
* 3: reserved

principal specifies the entity\_id of the entity that the initialization segment of that entity or the initialization of a direct or indirect member of the entity can be used for the initialization of any track in this group.

tag specifies the tag of the entity group which may be used for selection purposes for the decoder. For MPEG-H Audio the value of this field shall contain the whitespace-separated list of mae\_GroupIDs that are contained in the described switching group.

structural\_brand specifies the structural brand of all direct and indirect entities of this group.

mediaprofile\_brand specifies the media profile brand that all direct and indirect entities of this group conform to.

### Comments during MPEG#144

1. We welcome clarifications on the following:
   1. Coverage of grouping from DASH
   2. Coverage of grouping from CMAF
   3. Can semantics be written in a ‘pure’ ISOBMFF terminology?
   4. Where should this be standardized? ISOBMFF/CMAF/DASH
   5. Missing parameters
2. What does switchable\_flag equal to 0 mean? Why are tracks in the same switching group if they are not switchable?
3. "timed aligned at the substructural level" is vague.
4. The principal field takes 32 bits and is not friendly for editing (because file editors need to understand the 4CC-specific part of the entity group). I would rather suggest that if init\_type is equal to 0, the principal track is the first track listed in the entity group.
5. Brands in ISOBMFF are four-character codes. Why is utf8string used rather than unsigned int(32)?

## Signaling DASH descriptors in ISOBMFF (based on m65349)

### Usecase

The DASH specification uses the DASH descriptors to annotate media streams with various properties.

The upcoming DASH REaP specification proposes “Storage Track Identifiers” for its “Track format for storage of live archives”. These track identifiers are defined as counterparts for DASH and CMAF constructs. Within the file format, it is proposed to utilize the ‘kind’-Box by setting the schemeIdUri string to the respective identifier.

The File Format specification with its TrackKindBox offers an obvious solution to carry such data, but as defined today it is restricted in its feature coverage.

### Identified Problems with TrackKindBox

Considering the possibilities offered by DASH’s DescriptorType, the ISOBMFF TrackKindBox has several drawbacks:

* **Location**: While the TrackKindBox (‘kind’) is restricted to be used on tracks only, some of the DASH DescriptorType-derived elements may be present on different levels as well (e.g. on Period-level: AssetIdentifier, SupplementalProperty). Therefore, this limitation of the TrackKindBox prohibits its usage as a coequal signaling method.  
  If the TrackKindBox is envisioned to be used for the beforementioned generalized purposes, its location needs to be allowed at various other locations as well.
* **Naming**: A DASH packager and manifest generator operated on a “master” ISOBMFF file would require detailed knowledge about each possible scheme definition to write the appropriate DASH element.  
  Sometimes such generation based on implicit knowledge is difficult (e.g. both the DASH Role and Accessibility elements utilize the same schema definition), sometimes this might not be possible at all (comparing e.g. SupplementalProperty or EssentialProperty descriptors).
* **Grouping**: The possibilities offered by the @id attribute optionally available with DASH DescriptorType-derived elements cannot be implemented with the TrackKindBox.
* **Scope:** Furthermore, considering a DASH packager workflow, if the TrackKindBox would be used for the carriage of identifiers as proposed by the DASH REaP draft, such identifiers rather require getting deleted by DASH segmenters/packagers (or they are to be ignored by a player/client). To accomplish this, a packager requires knowledge of each possible schema it might ever see. Such full coverage of definitions is rather error-prone, considering that region-specific or private definitions might occur.

### Proposal

It is recommended that the file format group decides on how the TrackKindBox is to be used in the future and whether other derived boxes would be desirable. For instance, a very generic and maybe too broad solution was already provided as m64293.

As far as controlling packager operation is concerned, a dedicated box targeting packagers only might elegantly avoid the problems of the scope of an identifier, such as:

**Packager Identifier**

**Definition**

Box Type: '????'  
Container: UserDataBox, AudioElementBox or PreselectionGroupBox   
   
Mandatory: No  
Quantity: Zero or more

The PackagerIdentifierBox labels a track with its role or kind.

It is derived from and has the same syntax as the TrackKindBox: It contains a URI, possibly followed by a value. If only a URI occurs, then the property is defined by that URI; if a value follows, then the naming scheme for the value is identified by the URI.

The PackagerIdentiferBox is intended to be used to guide streaming segmenters/packagers. While segmenters are expected to remove these Boxes when generating their output files or segments, players are expected to ignore them.

**Syntax**

aligned(8) class PackagerIdentifierBox extends KindBox('????', version = 0, 0)   
{  
}

### Comments during MPEG#144

1. How do we want to map DASH Adaptationset-level descriptors into the new EntityGrouping approach (see clause 2.1), given the mismatch between descriptorType and the kind box?