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

Text

Description automatically generated ISO/IEC JTC 1/SC 29/WG 03 N0867

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

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

**Document type:** Output Document

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

**Status:** Approved

**Date of document:** 2023-05-12

**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** **0867**

**January 2023, Online**

|  |  |
| --- | --- |
| **Source** | **MPEG Systems working group** |
| **Status** | **Output** |
| **Title** | **Exploration on alignment of ISOBMFF/DASH/CMAF terminology, concepts and solutions** |
| **Serial Number** | **22609** |

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 document 2

1.2 Background 2

1.3 Processing Diagram 4

1.4 Requirements and Scenarios 4

1.5 Previous work summary 5

2 Identfied ISOBMFF text issues 5

2.1 Clarification on Alternate Groups 5

3 Proposals 5

3.1 Hierarchical track groups (m62998, m63118, m63258, m63260) 5

3.1.1 Use case 5

3.1.2 Existing solution in the ISOBMFF 8th edition DIS 6

3.1.3 Proposed solution 6

3.1.4 Proposed specification revision 7

3.2 Use of identifiers (m63400) 9

3.2.1 Overview 9

3.2.2 Examples: Use of the identifiers in 23009-9 CD 9

3.3 Simplified Preselection (m63254) 11

3.3.1 Overview 11

3.3.2 Proposal update to 14496-12 text 11

3.4 Cross-ISOBMFF files track identifiers (m63099) 13

3.4.1 Overview 13

3.4.2 Proposed Changes 13

# General considerations

## Purpose of this document

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

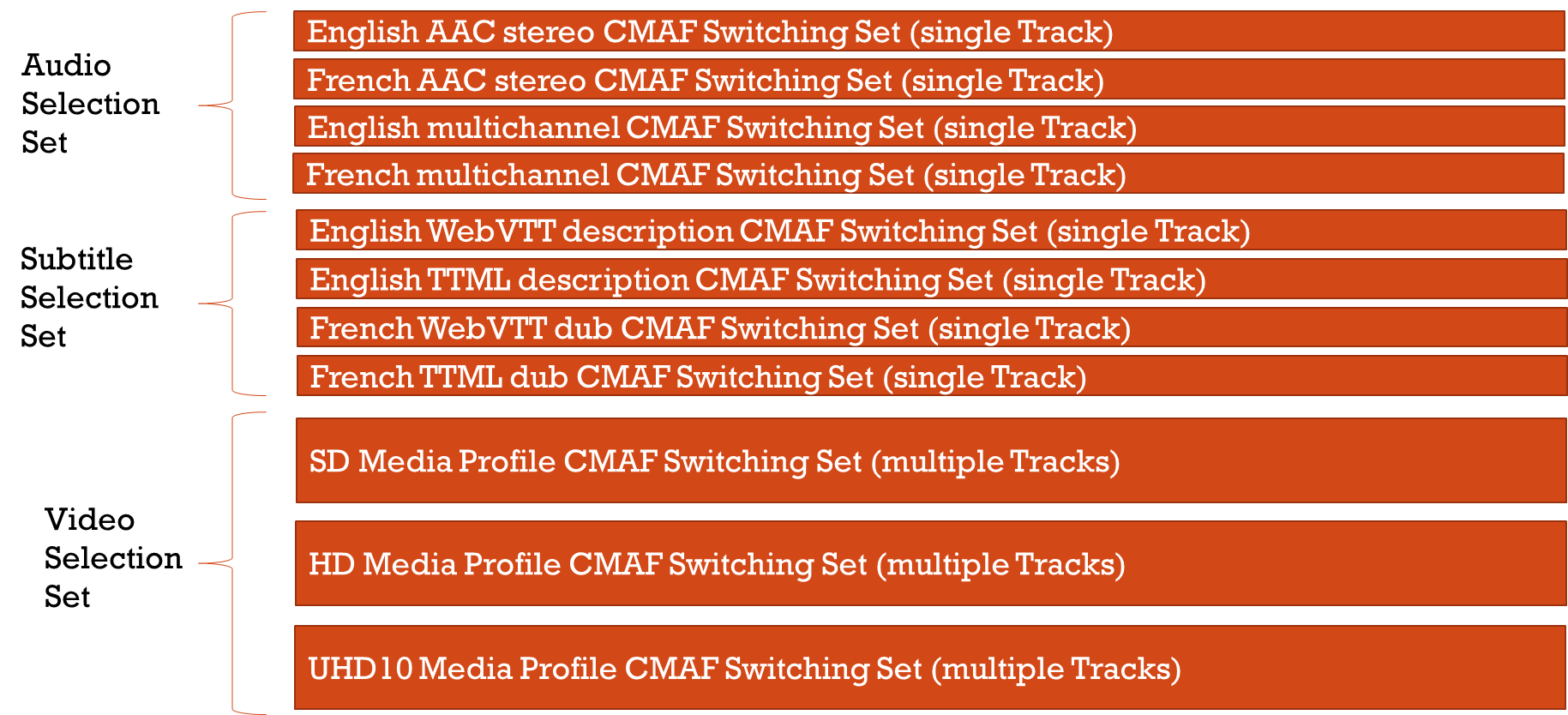
* Technical Report that describes the mapping between ISOBMFF, CMAF, and DASH terms and concepts.
* ISOBMFF extensions to indicate a track grouping that is equivalent to a CMAF Switching Set and a DASH Adaptation Set
* ISOBMFF extensions equivalent to indicating switching between DASH Adaptation Sets
* 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, scal)
  + 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 downloads these tracks and does 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 file to have unique track identifiers is not carried forward
2. Each file includes its own movie header and relationship about 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 early enough select the tracks based on this information.

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

* CMAF Tracks
* CMAF Switching Sets
* CMAF Groups
* Depending 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 providers has generated CMAF content for a movie or a CMAF presentation. For storage efficiency reason, the content provider wants to store the entire CMAF presentation in an ISO BMFF file format that is compatible to 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 debug purposes but also for 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

Document WG03N805 ([MDS22329](https://dms.mpeg.expert/doc_end_user/current_document.php?id=86329&id_meeting=193)) summarizes the exploration prior to MPEG#142 meeting. It includes:

1. A tentative mapping between the ISOBMFF, CMAF and DASH entities, and identifies the existing gaps in the mapping.
2. Possible clarifications needed on the ISOBMFF alternate groups and track groups in ISO/IEC 14496-12.
3. A conceptual entry sample processing for ISOBMFF, CMAF and DASH in a single framework
4. Various use-cases, and proposals for signaling bitrate switching, CMAF switching sets, preselection of CMAF switching sets, switching cross DASH adaptation Sets, dependent tracks in ISOBMFF.

This document is the result of the contributions to and discussions at the MPEG#142 meeting. It was agreed that the document only focuses on the results of the discussion and only includes the latest candidate proposals. For the history and previously proposed solutions, please refer to WG03N805.

# Identified ISOBMFF text issues

## Clarification on Alternate Groups

Options are:

a - alternate groups are intended for initial track selection only and should not be used for dynamic switching

b - update the spec to clarify how alternate groups can be used in dynamic switching (drawback: no parameters like time-aligned or not, same codec or not, same encryption)

We welcome contributions on clarifying the “dynamic switchability of track groups”.

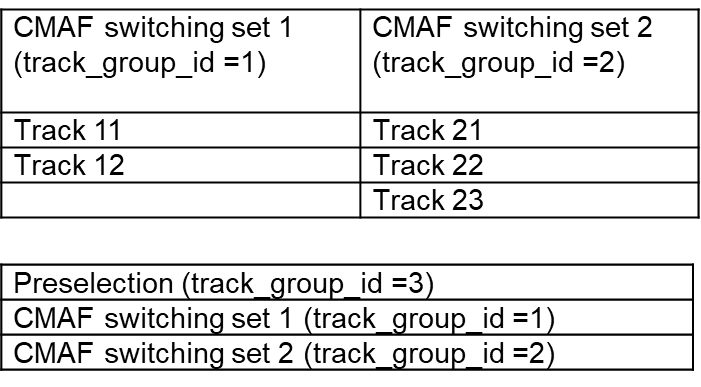
# Proposals

## 3.1 Hierarchical track groups (m62998, m63118, m63258, m63260)

### Use case

The use case allows grouping of the ISOBMFF tracks in a hierarchical manner, i.e. defining groups of groups of tracks at two or more levels. One main application is for defining a preselection of CMAF Switching Sets. In this application, one or more CMAF Switching Sets of tracks may exist in an ISOBMFF file, along with zero or more single tracks. A preselection track group is defined from zero or more CMAF Switching Sets, possibly along with one or more single tracks. In this preselection, the tracks of a CMAF Switching Set are interchangeable, i.e. any track of the Switching Set has the same property concerning its role in the preselection grouping.

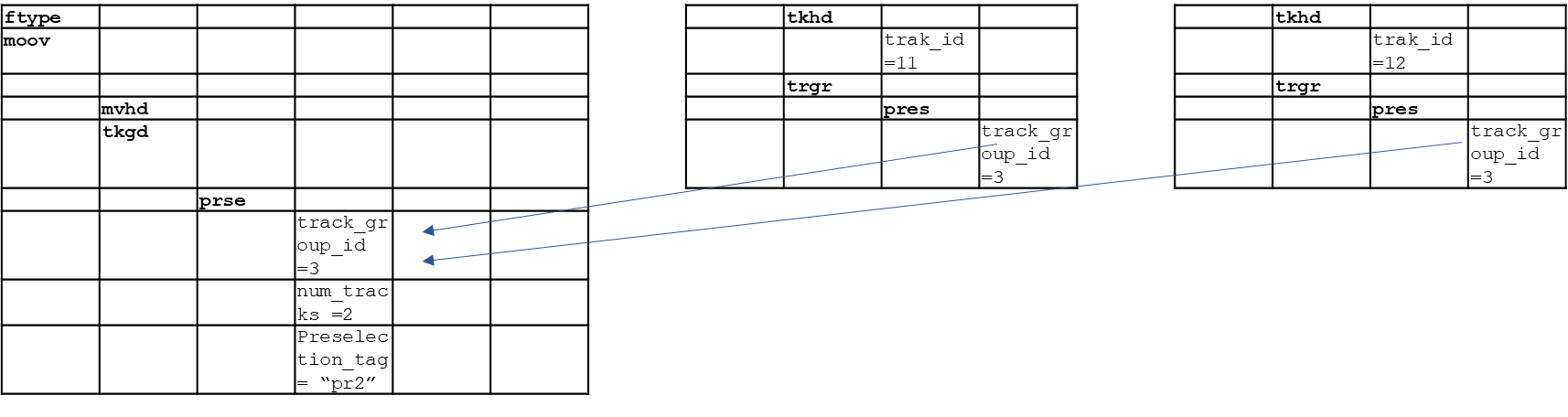
Figure 1 shows an example of the above use case.



**Figure 1 – Example: preselection of two CMAF Switching Sets.**

### Existing solution in the ISOBMFF 8th edition DIS

As shown in the example of Figure 2, the current ISOBMFF 8th edition DIS only allows single tracks to be a member of a preselection group.



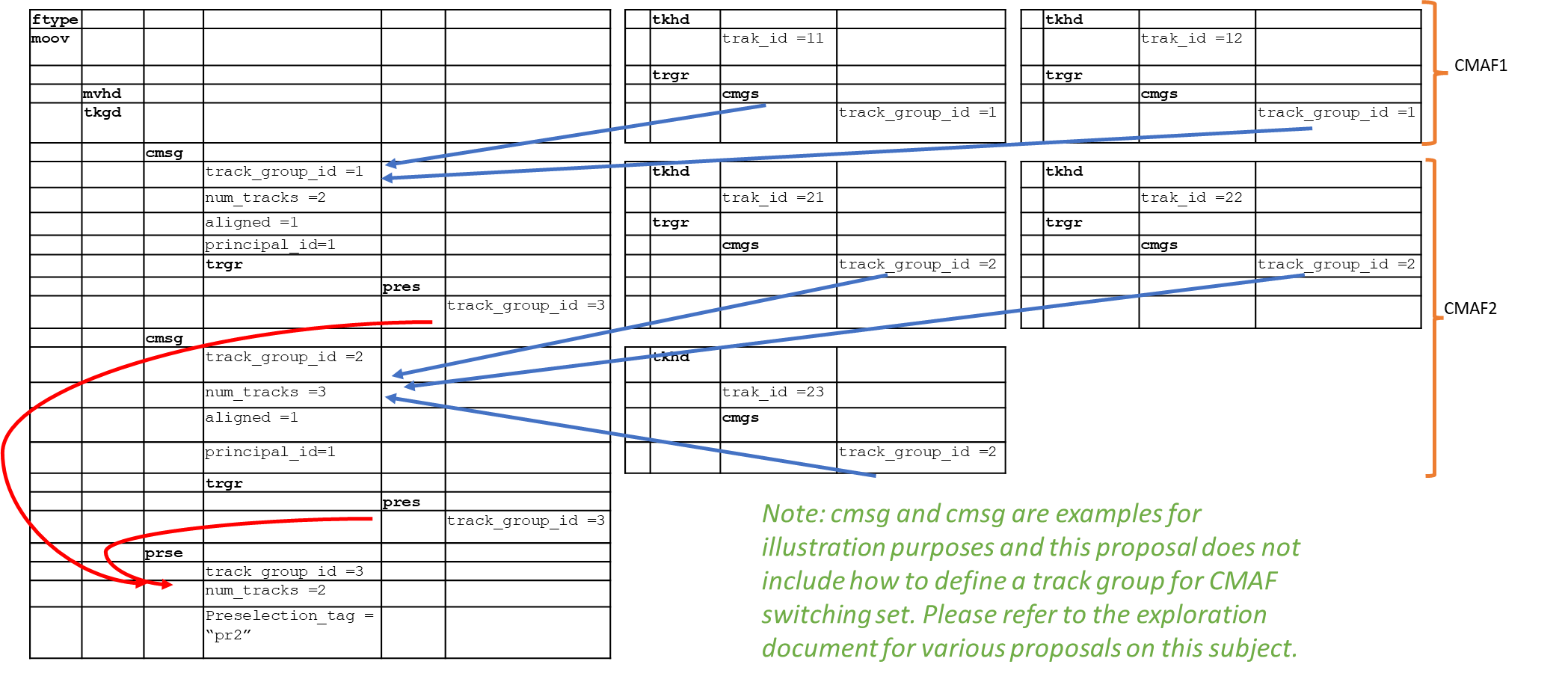
**Figure 2 – The existing preselection track groups in ISOBMFF 8th edition DIS**

In the above solution:

* The grouping by including track\_group\_id in each track
* 4cc of track group box and track group entry box show the property of grouping
  + Example: preselection “pres” and “prse”
* The track group entry box has common properties such as the number of tracks
* Each track also may have its parameters
* No solution of a group of track groups

### Proposed solution

Figure 3 demonstrates the hierarchical track grouping for the Figure 1 example.



**Figure 3 – hierarchical track grouping for Figure 1 example**

In this approach, a (child) track group may be used as a member of another (parent) track group. To show such associated, the child track group Track Entry Box may include a ‘trgr’ box that includes the track\_group\_id of the parent track group. In this approach:

* Track group is a group of single tracks and other track groups.
* The grouping is indicated by including track\_group\_id in each track or each track entry box.
* The track group entry box of the parent track group has common properties of tracks and/or children track groups.
* Each track or child track group also may have its own parameters.
* 4cc of track group box and track group entry box show the property of grouping.
  + Example: preselection “pres” and “prse”
* This approach accommodates multi-level grouping.
* The number of members in the parent track group is the number of its immediate children.
  + e.g. preselection example has two members, CMAF1 and CMAF2, i.e number\_tracks =2

NOTES:

1. We recommend not to use alternate\_group in combination with track groups, and use the hierarchical track groups instead.

2. As shown in the example, implementation of the use case is only possible if a track group type is defined for the switching sets/adaptation sets.

### Proposed specification revision

The revisions are shown in red.

**8.3.5 Track group description box**

The TrackGroupDescriptionBox provides an array of TrackGroupEntryBoxes, where each TrackGroupEntryBox provides detailed characteristics of a particular track group.

The syntax of the TrackGroupEntryBox is determined by track\_group\_entry\_type. TrackGroupEntryBox is mapped to the track group by a unique track\_group\_entry\_type that is associated with a track\_group\_type defined in subclause 8.3.4.3. More than one TrackGroupEntryBox with the same track\_group\_entry\_type may be present in TrackGroupDescriptionBox, in that case TrackGroupEntryBoxes shall have different track\_group\_id.

**8.3.5.1 Syntax**

aligned(8) class TrackGroupDescriptionBox extends Box('tkgd')  
{  
 Box boxes[];  
}

aligned(8) abstract class TrackGroupEntryBox (unsigned int(32) track\_group\_entry\_type, unsigned int(8) version, unsigned int(24) flags) extends FullBox(track\_group\_entry\_type, version, flags)   
{   
 unsigned int(32) track\_group\_id;  
 // the remaining data may be specified   
 // for a particular track\_group\_entry\_type   
}

**8.3.5.2 Semantics**

The TrackGroupDescriptionBox contains TrackGroupEntryBoxes. track\_group\_entry\_type indicates a 4CC that is associated with a track\_group\_type.

The pair of track\_group\_id and track\_group\_entry\_type identifies the track group that the TrackGroupEntryBox describes.

A TrackGroupEntryBox may contain a TrackGroupBox (‘trgr‘). If this is the case, each TrackGroupTypeBox within the TrackGroupBox indicates that the track group under description is a member of another track group identified by the track\_group\_id in the TrackGroupTypeBox.

All attributes uniquely characterizing a preselection should be present in the PreselectionTrackGroupEntryBox for the preselection.

**8.3.5.4.1.2 Syntax**

aligned(8) class PreselectionTrackGroupEntryBox  
 extends TrackGroupEntryBox('prse', version=0, flags)  
{  
 unsigned int(8) num\_~~tracks~~members;  
 utf8string preselection\_tag;  
 if (flags & 1) {  
 unsigned int(8) selection\_priority;  
 }  
 // Boxes describing the preselection  
}

**8.3.5.4.1.3 Semantics**

num\_~~tracks~~members specifies the number of ~~non-alternative tracks grouped by~~ members of this preselection track group.

A ~~track grouped by~~ member of this preselection track group is a track or track group that has the 'pres' track group with track\_group\_id equal to the ID of this preselection.  
The number of ~~non-alternative tracks grouped by~~ members of this preselection track group is the sum of the following:

* the number of tracks that ~~have alternate\_group equal to 0 and~~ are grouped by this preselection track group,
* the number of ~~unique non-zero alternate\_group values in~~ all track groups that are grouped by this preselection track group.

The value of num\_~~tracks~~members shall be greater than or equal to the number of ~~non-alternative tracks grouped by~~ members of this preselection track group in this file.  
A value equal to 0 indicates that the number of ~~tracks grouped by~~ members of this track group is unknown or not essential for processing the track group.  
NOTE 1 The value of num\_~~tracks~~members can be greater than the total number of ~~non-alternative~~ tracks and track groups containing a PreselectionGroupBox with the same track\_group\_id in this file when the preselection is split into multiple files.  
NOTE 2 When a player has access to fewer ~~non-alternative~~ tracks grouped by this preselection track group than indicated by num\_~~tracks~~members, the player might need to omit the tracks grouped by this preselection track group.

preselection\_tag is a codec specific value that a playback system can provide to a decoder to uniquely identify one out of several preselections in the media.

## Use of identifiers (m63400)

### Overview

In practice, an easy solution for supporting CMAF or DASH constructs in ISOBMFF files is to use identifiers in the tracks that allow a converter to then easily create a CMAF and/or DASH presentation from an ISOBMFF formatted file.

In addition, CMAF lacks explicit signaling, thus these identifiers may also be added to CMAF tracks. The idea is to allow a reader to do a grouping based on the values of the identifiers, i.e. tracks with the same switching set identifier are grouped together.

MPEG-DASH part 9 REaP (23009-9) defines a storage format, mainly targeting CMAF track files, that uses identifiers for constructs for DASH and/or CMAF in files. We think that as this mapping is already in Committee Draft Ballot, it may be a good starting point for adding these mappings using identifiers in *kind* and *udta* boxes. The REAP specification defines the schemeURI and values to add different identifiers for different DASH and CMAF constructs. It is possible to add additional

### Examples: Use of the identfiers in 23009-9 CD

This clause defines urn’s and identifiers to code properties in each track to allow recreating a media presentation or playlist if needed. Both identifiers corresponding to DASH and CMAF constructs are introduced and linked.

Figure 4 links the DASH identifiers and the CMAF identifier constructs. Table 2 defines the schemeURI for each identifier when used according to this proposal when signalling in a `kind` box. The schemeURI and value attribute in kind shall be set to the identifier scheme and UUID/ID respectively.

Track identifiers may be present in tracks. If present, they shall be signalled using a schemeURI in the kind box in UserDataBox (udta) as and the value as a UTF-8 character encoded and null terminated UUID or id value as string. The defined schemeURI from Table 2 may be included in a storage track file.



Figure 4 - Relationship between the manifest and its segments.



Figure 5 - Orphaned content stream or asset with information to do reverse lookup.

1. Example CMAF identifiers and mapping to DASH identifiers (informative?).

|  |  |
| --- | --- |
| **DASH construct** | **CMAF construct** |
| MPD ID | CMAF Presentation ID |
| Period ID | CMAF Presentation ID |
| AdaptationSet  Group ID | Aligned Switching Set ID |
| AdaptationSet ID | SwitchingSet ID |
| Representation ID | CMAF track id (not to confuse with track\_id) |

1. Link between constructs in DASH and HLS and defined urn’s (informative)

|  |  |
| --- | --- |
| Content ID (optional) | urn:mpeg:asset-storage-format:cmaf-content-id |
| CMAF Presentation ID (MPD) | urn:mpeg:asset-storage-format:cmaf-mpd-id |
| CMAF PresentationID (Period) | urn:mpeg:asset-storage-format:cmaf-period-id |
| Aligned Switching Set ID | urn:mpeg:asset-storage-format:cmaf-aligned-switching-id |
| SwitchingSet ID | urn:mpeg:asset-storage-format:cmaf-switching-id |
| CMAF track id (not to confuse with track\_id) | urn:mpeg:asset-storage-format:cmaf-track-id |

## Simplified Preselection (m63254)

### Overview

The contribution m62053 argued that the stream interleaving is not the task of the ISO file format parser/CMAF parser/DASH access client. This task belongs to the Sample entry processor. The result was included in the exploration document N805.

This proposal simplifies the preselection in ISOBMFF accordingly. The simplified solution provides inherent extensibility for signaling bitstream manipulation commands for various use cases.

### Proposal update to 14496-12 text

Preselection processing box

Definition

Box Type: 'prsp'  
Container: PreselectionGroupBox  
Mandatory: No  
Quantity: Zero or one

This box contains information about how the tracks contributing to the preselection can be processed. Media type specific boxes may be used to describe further processing.

Syntax

aligned(8) class PreselectionProcessingBox   
 extends FullBox('prsp', version = 0, flags )  
{  
 unsigned int(8) track\_order;  
 unsigned int(1) sample\_merge\_flag;  
 unsigned int(7) reserved;  
 utf8string interleaving\_tag  
 // further attributes and Boxes defining additional processing of  
 // the track contributing to the preselection  
}

Semantics

track\_order defines the order of this track relative to other tracks in the preselection as described below.

sample\_merge\_flag equal to 1 indicates that this track is enabled to be merged with another track as described below.

interleaving\_tag provides the interleaving instructions to be used for interleaving samples or groups of samples of this track with other tracks in this preselection. If not defined in this document, the syntax and semantics should be defined by a decoder specification or related documents.

Sample entry specific specifications might require the tracks for a preselection to be provided to the respective decoder instances in a specific order. Since other means, such as the track\_id, are not reliable for this purpose, the track\_order is used to order tracks in a preselection relative to each other. A lower value of track\_order indicates that at a given decoding time the sample of the containing track is provided to the decoder before the sample with the same given decoding time of other tracks with higher number. If two tracks in a preselection have their track\_order set to the same value or if the preselection processing box is absent for at least one of the tracks, the order of these tracks is not relevant for the preselection, and the samples with a particular decoding time for these two tracks can be provided to the decoder in any order.

A merge group is defined as a group of tracks, sorted according to track\_order, where one track with the sample\_merge\_flag set to 0 is followed by a group of consecutive tracks with the sample\_merge\_flag set to 1. All tracks of a merge group shall be of the same media type and shall have all samples decoding-time-aligned.

The process of merging samples in a merge group is out-of-scope of this specification and shall be defined by the codec specification for the sample entry type, optionally guided by the information from the interleaving\_tag.NOTE  Tracks in a merge group may have different sample entry types.

If the tracks in the merge group are all of sample entry type of “mhm2” (MPEG-H 3D Audio), the interleaving\_tag shall be set to “mhm2” (null terminated) and the specific merging process for this sample entry is defined in ISO/IEC 23008-3:2019, subclause 14.

EDITOR’S NOTE:  The above paragraph may be moved to an MPEG-H amendment when a new amendment work item starts. Then, this paragraph will be replaced by a reference to that amendment.

For tracks not part of a merge group, a merging process is not specified by this document.

## Cross-ISOBMFF files track identifers (m63099)

### Overview

ISOBMFF specifies encapsulation of media streams into tracks of a file. It typically assumes that all tracks of a media presentation are contained in a file and a single movie box documents all tracks in the file. However, some of the ISOBMFF principles are not carried forward to CMAF or DASH, because they rely on the concept of late binding, i.e., each track stored in a separate file.

In case of late binding

1. The requirement to have unique track identifiers is not carried forward
2. Each file includes its own movie header and relationship about the tracks in different files cannot be expressed

Without the uniqueness established between tracks (media streams), a file reader will have to deal with ambiguity in selecting the tracks for playback, and this may lead to unintended behaviour at the playback end.

In this proposal, we propose to address the above concerns by introducing new MovieHeaderExtensionBox which is a companion box to the MovieHeaderBox, and carries one or more unique identifiers. When two or more ISOBMFF files carry the same unique identifiers, then the files represent the same presentation and can be combined to get a conformant ISOBMFF file. Moreover, CMAF file/segment writers and MPD generators can verify based on the unique identifiers which files represent the same presentation to prepare them for the same CMAF switching set and DASH adaptation set.

### Proposed Changes

**XX Movie header extension box**

**XX.1 Definition**

Box Type: 'mvhe'  
Container: MovieBox  
Mandatory: No  
Quantity: Exactly one

The MovieHeaderExtensionBox is a companion box and immediately follows the MovieHeaderBox within the MovieBox. This box carries unique identifiers for the movie in the current ISOBMFF file.

The unique\_ID present in the MovieHeaderExtensionBox can be used for determining that different files are part of the same presentation and different files can be combined to a single conformant ISO base media file. If a track belongs to two or more different presentations multiple unique\_IDs are present in the MovieHeaderExtensionBox which can be then mapped to different presentations.

When different files carry the same unique\_ID then the files can be combined together such that the track IDs can be modified to be unique in the resulting conformant ISO base media file.

NOTE When tracks in different files do not carry a unique\_ID, there is no guarantee that combining those files lead to a conforming ISOBMFF file.

**XX.2 Syntax**

aligned(8) class MovieHeaderExtensionBox extends FullBox('mvhe', version, 0) {  
 unsigned int(8) unique\_ids\_count\_minus1;  
 unsigned int(128) unique\_ID[unique\_ids\_count\_minus1 + 1];  
}

**XX.3 Semantics**

unique\_ids\_count\_minus1 plus 1 indicates the number of unique IDs present in the movie.

unique\_ID is an array of unique values for the movie in the current file. Array elements of unique\_ID can take integer values (UUID) as defined in RFC 4122.