Content Metadata Specification

Version 2.2 31 July 2015

Content Metadata Specification Version 2.2

<u>Notice</u>:

As of the date of publication, this document is a release candidate specification subject to DECE Member review and final adoption by vote of the Management Committee of DECE in accordance with the DECE LLC Operating Agreement. Unless there is notice to the contrary, this specification will become an adopted "Ecosystem Specification" on 15 September 2015.

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Digital Entertainment Content Ecosystem (DECE) LLC ("DECE") and its members disclaim all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted herein.

This document is subject to change under applicable license provisions, if any.

Copyright © 2009-2015 by DECE. Third-party brands and names are the property of their respective owners.

<u>Optional Implementation Agreement</u>:

In addition to the UltraViolet License Agreements which cover implementation of the DECE Ecosystem Specifications within the UltraViolet Ecosystem, DECE offers an optional license agreement relating to the implementation of this document outside the Ecosystem ("RAND Agreement"). Entities executing the optional RAND Agreement receive the benefit of the commitments made by DECE's members to license on reasonable and nondiscriminatory terms their patent claims necessary to the implementation of this document in exchange for a comparable patent licensing commitment. Copies of the license agreements are available at the DECE web site referenced below.

<u>Contact Information</u>:

Licensing and contract inquiries and requests should be addressed to us at:

<u>http://www.uvvu.com/uv-for-business</u>

Content Metadata Specification Version 2.2

Contents

Content Metadata Specification Version 2.2

1.1 Overview of DECE Metadata

DECE Metadata is used throughout the Ecosystem. It is created as part of the Publishing process, used by Retailers to support sales, DSPs do manage assets, User Interface and Customer Support for displaying Rights information to Users, and Devices to manage assets and display content information.

DECE Metadata is only a portion of the metadata used throughout the Ecosystem. It is anticipated that parties will use metadata from various sources to provide the best possible experience for the User.

DECE Metadata is based on Common Metadata. Descriptive Metadata used in the Coordinator and elsewhere is a specific subset of Common Metadata defined in *Common Metadata Derived Types* below. Container Metadata, information included in Digital Common File Format (CFF) Containers (DCCs) draws upon Common Metadata and is defined in *Container Metadata* below.

1.2 Overview of Common Metadata

Common Metadata [DCMETA] includes elements that cover typical definitions of media, particularly movies and television. Basic Metadata includes descriptions such as title and artists. It describes information about the work independent of encoding. Physical metadata describes information about individual encoded audio, video and subtitle streams, and other media included. Package and File Metadata describes one possible packaging scenario and ties in other metadata types. Ratings information is described.

Common Metadata is designed to provide definitions to be inserted into other metadata systems, so ancillary participants in DECE will ideally be using at least some common elements.

Common Metadata was created to accommodate the common elements of various metadata systems under development, primarily DECE and the Entertainment Merchants Association (EMA). DECE specifies metadata for exchange directly between Content Publishers and the Coordinator, and between the Coordinator, and Retailers, LASPs, DSPs and Devices. Interfaces directly between Content Publishers and Retailers LASPs and DSPs are out of scope. EMA nicely fills this gap by providing elements identical to DECE metadata elements. Elements are identical because they work of the Common Metadata Specification.

Content Metadata Specification Version 2.2

1.3 Document Organization

This document is organized as follows:

1. Introduction—Provides background, scope and conventions

2. Identifiers – References identifiers used by DECE and defined in Common Metadata

3. Common Metadata Derived Type – Defines Common Metadata types used in DECE

4. Container Metadata – Defines metadata for DECE Common File Format (CFF) Containers

1.4 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]. 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.

Notational Conventions are described in Common Metadata [DCMETA].

1.5 Normative References

1.5.1 DECE References

[DMedia]	Common File Format & Media Format Specification
[DDMP]	Media Package Specification
[DCMeta]	Common Metadata Specification, Version 2.3c md-v2.3.xsd
[DCMetaCR]	Common Metadata Ratings Specification

[DCManifest]	Common Media Manifest Metadata Specification, Version 1.4 manifest-v1.4.xsd

1.5.2 Other Normative References

[RFC1738]	T. Berners-Lee, et al, *RFC 1738, Uniform Resource Locators (URL),* December 1994. http://www.ietf.org/rfc/rfc1738.txt
[RFC2141]	URN Syntax, May 1997
[RFC3629]	UTF-8, a transformation format of ISO 10646
[RFC3986]	Uniform Resource Identifiers (URI): Generic Syntax, January 2005
[RFC4647]	Philips, A., et al, *RFC 4647, Matching of Language Tags,* September 2006. http://www.ietf.org/rfc/rfc4647.txt
[RFC5646]	Philips, A, et al, *RFC 5646, Tags for Identifying Languages*, IETF, September, 2009. http://www.ietf.org/rfc/rfc5646.txt
[RFC5891]	Klensin, J., RFC 5891, *Internationalized Domain Names in Applications (IDNA): Protocol*, August 2010. http://www.ietf.org/rfc/rfc5891.txt
[IANA-LANG]	IANA Language Subtag Registry. http://www.iana.org/assignments/language-subtag-registry
[IANA-IMAGE]	IANA Image Media Types, http://www.iana.org/assignments/media-types/image
[IANA-MIME]	IANA MIME Media Types, http://www.iana.org/assignments/media-types
[TTML]	Timed Text Markup Language (TTML) 1.0, W3C Proposed Recommendation 14 September 2010, http://www.w3.org/TR/ttaf1-dfxp/
[ISO-10918-1]	ISO/IEC 10918-1:1994, Information technology -- Digital compression and coding of continuous-tone still images: Requirements and guidelines
[ISO-10918-5]	ISO/IEC 10918-5:2013, Information technology -- Digital compression and coding of continuous-tone still images: JPEG File Interchange Format (JFIF)
[ISO-15948]	ISO/IEC 15948:2004, Information technology -- Computer graphics and image processing -- Portable Network Graphics (PNG): Functional specification.

[XML]	"XML Schema Part 1: Structures", Henry S. Thompson, David Beech, Murray Maloney, Noah Mendelsohn, W3C Recommendation 28 October 2004, http://www.w3.org/TR/xmlschema-1/ "XML Schema Part 2: Datatypes", Paul Biron and Ashok Malhotra, W3C Recommendation 28 October 2004, http://www.w3.org/TR/xmlschema-2/
[ISO-P2H]	ISO/IEC Directives, Part 2, Annex H http://www.iso.org

1.6 Informative References

MovieLabs metadata information may be found at http://www.movielabs.com/md.

EMA metadata information may be found at http://www.entmerch.org/programsinitiatives/the-ema-metadata-structure/index.html.

[TR-META-EMA]	EMA Metadata, TR-META-EMA, v1.2, November 1, 2011, http://www.movielabs.com/md/ema/v1.2/EMA%20Metadata%20v1.2.pdf
[XSD-META-EMA]	XML Schema to accompany [TR-META-EMA], November 1, 2011, http://www.movielabs.com/schema/ema/v1.2/ema.xsd

1.7 Encoding

Metadata SHALL be encoded using UTF-8.

2   Identifiers

Identifiers and metadata are closely linked. In essence, all identifiers have corresponding metadata that describes the object being identified. Just as it is useful to distinguish between different kinds of objects with different kinds of identifiers, it is useful to distinguish the metadata in terms of those same objects.

The primary objects being identified and described in Common Metadata are:

- Logical Asset (an entity to which a Right is granted); Asset Logical ID (ALID)

- Physical Asset (a Container); Asset Physical ID (APID)

- Content Metadata; Content ID (`ContentID`)

- Compound Object (groups logical assets sold together); Compound Object ID (`CompObjID`)

The following XML types describing identifiers are defined in Common Metadata [DCMETA]:

```
md:id-type

md:orgID-type

md:ContentID-type

md:AssetPhysicalID-type

md:AssetLogicalID-type

md:CompObjID-type
```

3   Common Metadata Derived Types

Some of DECE Metadata is based on Common Metadata [DCMETA]. Common Metadata includes:

- Basic Metadata—Information about logical assets that allows basic DECE functions beyond the Coordinator to function.

- Digital Asset Metadata—Information about physical assets (e.g., encoding) that may be required for basic user experience as provided by Coordinator and other Roles

The following XML types describing metadata are defined in Common Metadata [DCMETA]:

```
md:BasicMetadata-type
```

```
md:DigitalAssetMetadata-type
```

3.1 Metadata Constraints

DECE Metadata is a subset of Common Metadata as defined here.

The following defines whether metadata MAY be included (noted as optional) or SHALL BE included (not noted as optional), or otherwise included as noted. Any metadata allowed in [DCMETA] BasicMetadata-type not listed here MAY be included in the Basic Metadata. Any metadata allowed in [DCMETA] DigitalAssetMetadata-type not listed here MAY be included in Digital Content Metadata. Note that elements required in [DCMETA] are also required for DECE as excluding them would result in invalid XML.

- `BasicMetadata-type`
  - `ContentID` attribute
  - `UpdateNum`—SHALL be included if the record is an update (i.e., not the first record distributed)
  - `LocalizedInfo`
    - `TitleDisplay19` MAY be included. See note below.
    - `TitleDisplay60`
    - `TitleDisplayUnlimited` SHOULD be included
    - `TitleSort`
    - `ArtReference` – References to Images SHALL be included in accordance with Image Formats, Section 3.2. Additional images references MAY be included. References SHALL refer to available images.
    - `Summary190`
    - `Summary400`
    - `Summary4000` SHOULD be included
    - `Genre` SHOULD be included
    - `OriginalTitle` SHOULD be included
    - `CopyrightLine`
  - `RunLength`
  - `ReleaseYear`

- o `ReleaseDate` SHOULD include the highest date/time resolution available
- o `WorkType` – `WorkType SHALL` comply with enumeration in [DCMETA], Section 4.1.1.1
- o `PictureColorType`—optional, but it SHOULD be included
- o `PictureFormat`—optional, but it SHOULD be included
- o `AltIdentifier`—optional, but it SHOULD be included for all commonly used identifiers. For example, if ISAN is available, it should be included.
- o `RatingSet`—SHALL be included for all available ratings in the regions where Retailers are authorized to sell this content
- o `SequenceInfo` – SHALL be included if WorkType is: '`Season`' or '`Episode`'. SHALL be included for other WorkTypes when an order applies.
- o `Parent` – SHALL be included if WorkType is: '`Season`', '`Episode`', '`Promotion`', '`Excerpt`' or '`Supplemental`'. SHOULD be included for other WorkTypes if a parent relationship exists.
- • `DigitalAssetMetadata-type`—SHALL be included for each track included in the Container.
  - o `Audio`
    - ▪ `Type`
    - ▪ `Encoding`
      - • `Codec`
      - • `CodecType`—The IANA namespace SHALL be used
      - • `BitrateMax`
      - • `SampleRate`
      - • `SampleBitDepth`
    - ▪ `Language`
    - ▪ `Channels`
  - o `Video:`
    - ▪ `Type`
    - ▪ `Encoding (AVC1)`
      - • `Codec`—SHALL BE '`H.264, MPEG-4 Part 10`'
      - • `CodecType`—SHALL BE '`IANA:h264`'
      - • `BitrateMax`
    - ▪ `Encoding (AVC3)`
      - • `Codec`—SHALL BE '`H.264`'
      - • `CodecType`—SHALL BE '`mpeg4ra:avc3`'
      - • `BitrateMax`
    - ▪ `Encoding (HEVC)`
      - • `Codec`—SHALL BE '`H.265`'
      - • `CodecType`—SHALL BE '`mpeg4ra:hev1`'
      - • `BitrateMax`
    - ▪ `Picture:`
      - • `AspectRatio`
    - ▪ `ColorType`
    - ▪ `SubtitleLanguage`—SHALL be included if the video contains visible subtitles.
  - o `Subtitle` (if applicable)
    - ▪ `Format`
    - ▪ `Type`
    - ▪ `FormatType`—SHALL be '`SMPTE 2052-1 Timed Text`'
    - ▪ `Language`

Note: TitleDisplay19 was required in earlier versions, but is now deprecated.

Content Metadata Specification Version 2.2

If usage permits and TitleDisplayUnlimited is available, TitleDisplayUnlimited is preferred to TitleDisplay60.

OriginalTitle is not to be used in lieu of one of the TitleDisplay elements for display purposes.

BitrateMax is the maximum bitrate that the stream is guaranteed not to exceed. Note that this can be the maximum for the profile, the actual maximum or something in between.

3.2 Image Formats

3.2.1 Encoding

Images SHALL be encoded using the sRGB color model.

Images SHALL be in either JPEG or PNG formats.

JPEG images SHALL be encoded and packaged in accordance with [ISO-10918-1] and [ISO-10918-5].

PNG images SHALL be encoded and packaged in accordance with [ISO-15948] and [DMedia] Section 6.4.

In metadata, MIME types SHALL be in accordance with IANA Media Types Registry found at [IANA-MIME]. MIME types for images are found at [IANA-IMAGE]. For example, the allowable encoding MIME type for JPEG is 'image/jpeg' (i.e., not '/image/jpeg', 'image/jpg', 'jpg' or 'jpeg').

3.2.2 Format

3.2.2.1   Metadata Image Format

The intent is for images to be displayed well against a variety of backgrounds in conjunction with images from various sources.

Active pixels (pixels that contribute to the picture) in JPEG images SHALL fill the image. That is, no padding is allowed.

With the following exception, active pixels in PNG images SHALL fill the image. PNG images MAY be padded in with transparent pixels (i.e., pixels with alpha value of zero as per [ISO-15948], Section 2.4) on one pair of opposite edges with equal padding within one pixel applied to each edge. That is, active pixels must fill the image horizontally, vertically or both. If active pixels do not fill the image in one direction, transparent pixels are to be added in equal proportion, within single-pixel rounding error.

Images SHOULD have the following characteristics

Content Metadata Specification Version 2.2

- Formatted in portrait orientation

- Contain key art and title, as appropriate

- Do not contain release date, film rating, website, or small text

Multiple sizes are recommended to allow appropriate image size to be selected for various screen sizes. For example, a smaller display may use low resolution images as thumbnails in a locker view, and medium resolution images for the detailed display.

The following tables specify image formats and where they apply.

Resolutions are stated in pixels.

In the following table:

- Under Container
  - R means Recommended.
    - At least one image in a Recommended format associated with the Media Profile of the Container SHALL be included in the Container.
    - Images in Recommended formats in each Container Media Profile SHOULD be included in each Container with that Profile.
  - O means Optional
    - Images in Optional formats in each Container Media Profile MAY be included in each Container with that Profile.
- In Basic Metadata
  - R means Recommended
    - At least one image in a Recommended format SHALL be included with Basic Metadata.
    - Images in Recommended formats SHOULD be included with Basic Metadata.
  - O means Optional
    - Images in Optional formats MAY be included in Basic Metadata.
  - Any image format that is not supplied through Basic Metadata to the Coordinator SHALL be generated by the Coordinator as follows
    - Where other images are supplied in the same aspect ratio, missing images SHALL be generated using proportional resizing from the largest of those images.
    - Where other images are not supplied in the same aspect ratio, missing images SHALL be generated from the largest supplied images using proportional resizing

Content Metadata Specification Version 2.2

> to align one dimension and symmetric cropping (i.e., approximately same number of pixels from each side) to align the other dimension.

- Image format (PNG or JPEG) SHALL be retained when generating other images.

Note the specification of required images does not preclude the inclusion of additional images of any aspect ratio and resolution.

Locker view implementations in Nodes are thus guaranteed to have available at least six images, in small, medium, and large resolution in two common aspect ratios, either supplied by the Content Provider or derived by the Coordinator from the supplied images.

Locker view implementations in Devices will find at least one image in one of the two common aspect ratios in the Container. See [DDevice] 9.2.2 for related requirements.

Shape	Resolution (horizontal x vertical)	Container				Basic Metadata
		PD	SD	HD	UltraHD	
Portrait (0.67 aspect ratio)	96x144	O	O	O	O	O
	192x288	R	O	O	O	O
	800x1200	O	R	R	R	R
	1600x2400	O	O	O	O	O
Portrait (0.70 aspect ratio)	112x160	O	O	O	O	O
	224x320	R	O	O	O	O
	840x1200	O	R	R	R	R
	1680x2400	O	O	O	O	O

3.2.2.2 Chapter Image Format

Chapter images SHALL have square pixel aspect ratio.

Chapter images SHALL be of the aspect ratio ranging from 1:1 to the active picture area aspect ratio. For example, a video with a 16:9 picture with 1.85:1 active picture area could have chapter images with aspect ratios ranging from 1:1 to 1.85:1.

All Chapter images in a single Container SHALL be the same size.

Chapter images SHALL be no larger than the video picture size, and with a width no smaller than 1/8 of the video width. For example, if video is 1280x720 with all pixels active, the largest allowable image is 1280x720, and the smallest allowable image has 160 horizontal pixels (160x160 – 160x90). Given a 1280x720 (16:9) frame with 2:35:1 active picture, the smallest allowable image has 160 horizontal pixels and 67 vertical pixels.

3.2.3 Reference (Extensions)

URN image references as per Section 4.3 SHALL use <ext> as follows

- 'png' for PNG images
- 'jpeg' or 'jpg' for JPEG images.

URL image references SHALL have URL <path> as per [RFC1738] Section 3.3, and SHOULD end as follows

- '.png' for PNG images
- '.jpeg' or '.jpg' for JPEG images

3.2.4 Image Size

Images SHALL NOT exceed 10MB ($10x1024^2$ bytes).

4   Container Metadata

This section defines the profile for DECE metadata that will be included in the Digital CFF Container (DCC) further defined in [DMedia]. In a Common Media Package (CMP), including the DECE-specialized DECE Media Package (DMP), metadata resides in its own Part as defined in [DDMP], Section 4.3.4.

All types and elements here are in the 'mddece' namespace unless otherwise specified.

4.1 Required Metadata

DECE Container Required Metadata is a well formed XML document with a `MetadataMovie` root element.

The Required Metadata elements SHALL be as follows:

Element	Attribute	Definition	Type	Card.
MetadataMovie		Movie metadata that is required in a DECE Container.	ContainerMovieMeta data-type	

4.1.1 ContainerMovieMetadata-type

`ContainerMovieMetadata-type` is defined as follows:

Element	Attribute	Definition	Type	Card.
ContainerMovie Metadata-type				
	MetadataVersion Reference	A string that defines the version of the metadata in this element. If the metadata changes, this string SHOULD be included and unique relative to other instances of this attribute.	xs:string	0..1
	PresentationID	Media Presentation ID for Content in the Container.	md:LogialAssetID-type	
ContentMetadata		Mandatory descriptive metadata regarding the media in the Container.	mddece:ContainerC ontentMetadata-type	

RequiredImages		References to Container required images in Compliance with Section 3.2 Image Formats	md:DigitalAssetImageData-type	1..n
TrackMetadata		Descriptions of each track	mddece:ContainerTrackMetadata-type	
Ratings		Content ratings for media in the Container as defined in Common Metadata [DCMETA], Section 7.3.	md:ContentRating-type	0..1
Chapters		Chapter entry points	mddece:ContainerChapterList-type	0..1
OptionalImages		References to Container optional images	md:DigitalAssetImageData-type	0..n
TrackSelections			mddece:ContainerTrackSelectionList-type	0..1
ContainerVersionReference		A string that defines the version of the Container. It can be used as a reference to identify changes in the Container.	xs:string	0..1

4.1.2 ContainerContentMetadata-type

`ContainerInfo-type` contains the following information:

Element	Attribute	Definition	Value	Card.
ContainerContentMetadata-type				
ContentID		Content Identifier as a metadata reference identifier.	md:ContentID-type	

Content Metadata Specification Version 2.2

DECEMediaProfile		Identifier of Media Profile of Container. This element is required for DECE.	mddece:AssetProfile-type	(choice)
ProfileLevelIDC		Profile Level IDC as defined in [DMedia], Annex D.	xs:string	
RunLength		The duration of the primary track(s) in the Container as defined in Common Metadata [DCMETA], Section 4.1.	xs:duration	
Publisher		Content Publisher. This equivalent to DisplayName in the AssociatedOrg element as per [DCMETA], Section 4.1. The Content Publisher chooses which entry goes here.	xs:string	
ReleaseYear		These correspond with elements of the same name in Section 3.1 (including notes on TitleDisplay19, TitleDisplayUnlimited and TitleDisplay60)	xs:gYear	
ReleaseDate			xs:date	
ReleaseDateTime			xs:dateTime	
TitleDisplay19			xs:string	0..1
TitleDisplay60			xs:string	
TitleSort			xs:string	
Summary190			xs:string	
TitleDisplayUnlimited			xs:string	0..1
Summary400			xs:string	0..1
Summary4000			xs:string	0..1
DescriptionLanguage		Language of the Title and summary information in this element.	xs:language	
AlternateLocalizedInfo		Optional additional localized information (title, etc.)	mddece:ContainerLocalizedInfo-type	0..n

AssetProfile-type is a simple type of xs:string enumerated to 'PD', 'SD' and 'HD'. Note that because 'PD', 'SD' and 'HD' are the only allowed profile enumerations, ProfileLevelIDC must be used for other profiles. DECEMediaProfile is deprecated and ProfileLevelIDC is preferred.

4.1.2.1 ContainerLocalizedInfo-type

`ContainerLocalizedInfo-type` allows additional localized descriptions to be included.

Element	Attribute	Definition	Value	Card.
ContainerLocalizedInfo-type				
TitleDisplay19		These correspond with elements of the same name in Section 3.1 (including notes on TitleDisplay19, TitleDisplayUnlimited and TitleDisplay60)	xs:string	
TitleDisplay60			xs:string	
TitleSort			xs:string	
Summary190			xs:string	
TitleDisplayUnlimited			xs:string	0..1
Summary400			xs:string	0..1
Summary4000			xs:string	0..1
DescriptionLanguage		Language of the Title and summary information in this element.	xs:language	

4.1.3 `ContainerTrackMetadata-type`

`ContainerTrackMetadata-type` is defined as follows:

Element	Attribute	Definition	Value	Card.
ContainerTrackMetadata-type				
Track		Track description. One instance for each track.	md:DigitalAssetMetadata-type	1..n
SegmentSize		This value shall be equal to or greater than the number of bytes in the largest DCC Movie Fragment in the track. This element is required for DECE.	xs:int	0..1 (extension to md:DigitalAssetMetadata-type for Track)

In addition to elements and attributes defined in Section 3.1, `Track` element SHALL also include from the elementary streams in the file:

- `DigitalAssetVideoEncoding-type`:

    o `MPEGProfile` —set to profile_idc

    o `MPEGLevel`—SHALL be set to level_idc

- `DigitalAssetVideoPicture-type`:

    o `MasteredColorVolume and ColorEncoding` SHOULD be populated for any video track encoded at more than 8-bit depth. For example, a video track encoded using MPEG Main 10 Profile would qualify.

Content Metadata Specification Version 2.2

- o If `MasteredColorVolume` and `ColorEncoding` elements are included and "Mastering display colour volume SEI" is present in the video track as per [DMedia] Section 4.2.2, these elements' values SHALL correspond with the data in that SEI message.

- • `TrackReference` in `Audio`, `Video`, `Subtitle`, `Image` and `Interactive` as applicable. When present, `TrackReference` SHALL corresponds with `track_ID` in `tkhd' Box, as per [DMedia], Section 2.3.5.

Note that, to allow a device to set a decode buffer size, SegmentSize is set a value equal to or greater than the largest segment, in bytes, in the track.

4.1.4 Chapter Metadata

Chapter metadata identifies the locations within a track where chapters begin. Each chapter has a numerical index and an entry point that defines where the chapter starts.

Note that optional metadata may provide additional information about chapters.

Element	Attribute	Definition	Value	Card.
ContainerChapterLIst-type				
Chapter		Chapter entry point descriptor	mddece:ContainerChapter-type	

Elements SHALL be in chapter order.

Element	Attribute	Definition	Value	Card.
ContainerChapter-type				
	index	Chapter index.	xs:integer	
EntryTimecode		Entry point for chapter start.	xs:string, pattern [0-9]+\.[0-9]+	

DisplayLabel		Displayable text on a per-language basis for the chapter	xs:string	0..n
	language	Language of DisplayLabel. Must be included in all DisplayLabel elements if more than one DisplayLabel element is included. Matching is in accordance with Section 4.1.5.1 Use of Language	xs:language	0..1
ImageReference		Reference to a chapter image in accordance with Section 4.3 Image Reference and Section 3.2 Image Format. ImageReference SHALL only be used when MetadataMovie is contained within a multi-track DCC.	xs:anyURI	0..n
	language	Language of ImageReference. Must be included in all ImageReference elements if more than one ImageReference element is included and there is language-specific text in the image (i.e., burned in text). Matching is in accordance with Section 4.1.5.1 Use of Language	xs:language	0..1

ImageDescription		Information about dimensions and encoding of image. This element is required if either ImageReference or ExternalImageURL is provided. TrackReference has no meaning. TrackIdentifier SHALL be present except when MetadataMovie is contained within a Multi-track DCC.	md:DigitalAssetImageData-type	0..n

The `index` attribute is a number starting with 0 and increasing monotonically for each subsequent chapter.

`EntryTimecode` corresponds with a constrained form of the 'offset-time' syntax (without the metric field) of the media timebase defined in [TTML], Section 10.3.1, and corresponds with the beginning of the chapter in the video and/or audio tracks for which the chapters are identified. The metric is in units of seconds.

In the case of a rounding error that doesn't result in an integer number of frames, the video and/or audio frame(s) `EntryTimecode` refers to shall be the next decodable frame after the time in the media referenced by this value. For example, in a 30fps progressive video track, 0.1 = the 3rd frame. 0.101 = the 4th frame.

A reference to a chapter image is encoded differently when MetadataMovie is contained within a Multi-Track DCC (i.e., a DCC conforming to a Multi-Track Delivery Target as defined in [DMedia], Annex C) than when it used externally. When it is part of a Multi-Track DCC, ImageReference is used. When used external to a Multi-Track DCC—such as part of a Common Media Package (CMP)—ImageDescription/TrackIdentifier provides the reference information. Specific definitions within TrackIdentifier are defined elsewhere; for example, specific conventions are defined for CMP. When not in a multi-track DCC, ImageDescription/Language is interpreted the same as ImageReference/@language.

4.1.5 Track Selection Metadata

The `TrackSelection` element provides grouping information for which tracks belong to the same type, such as normal or commentary track selections. The `TrackSelection` element also provides

information about track priority and which audio and subtitle language pair is preferred based on the language preferences.

These data supplement information in `TrackMetadata/Track/Subtitle` and `TrackMetadata/Track/Audio`.

See Section 0 for information on expected interpretation of these data.

4.1.5.1 Use of Language

Track Select Metadata assumes that Devices have a parameter referred to here as System Language. The System Language is the current setting for the Device's interface language, perhaps set by the User. Users may also make independent language preference selections for audio language and for subtitle language.

Language preferences such as System Language are expressed as at least one language tag as per [RFC5646] and included in [IANA-LANG], possibly prioritized as a Language Priority List as per [RFC4647], Section 2.3. The assumed Priority List consists of at least the following language ranges:

1) The fully enumerated language tag including region, dialect or any other subtag element. For example, this would be a language tag from System Language, Audio User preference or Subtitle User preference.

2) The language tag from the first entry trimmed to the primary language tag, followed by a wildcard '*' subtag.

For example if the language is "en-GB", the Priority List will be "en-GB, en-*".

The best language match between a language preference (e.g., System Language) and one or more languages in a list (e.g., language tags in a list of audio tracks) is to be done in accordance with [RFC4647], Section 3.4 "Lookup".

4.1.5.2 ContainerTrackSelectionList-type

The `ContainerTrackSelectionList-type` provides information on what tracks go together in the `TrackGroup` element and which tracks are preferred.

Element	Attribute	Definition	Value	Card.

ContainerTrack SelectionList- type				
TrackGroup		A prioritized list of video, audio and subtitle track selections associated with each other (e.g., main program, commentary 1, commentary 2, etc.) .	mddece:ContainerTrackGroup -type	1..n

Each `ContainerTrackSelectionList-type` instance SHALL have a `TrackGroup` with `TrackSelectionNumber='0'`.

Each `TrackGroup` element SHALL have a unique value in `TrackSelectionNumber`.

4.1.5.3 ContainerTrackGroup-type

The ContainerTrackGroupType defines which tracks are associated with each other. This allows a Device to determine which tracks should be played together. It also contains LanguagePairs that include information about which tracks language combinations the author recommends for a given a System Language.

Within an element of this type, any audio track is associated with any video track and any subtitle track; and any subtitle track is associated with any video track and any audio track.

For example, all video, audio and subtitle track relating to the main program, regardless of CODEC and language would be in the same element. However, commentary audio and subtitle tracks would be in a separate element. A TrackGroup would not include both a 'primary' audio track and a 'commentary' subtitles track that are not intended to be played together. A Device would know from this structure which subtitle track to play with a commentary audio track.

Element	Attribute	Definition	Value	Card.
ContainerTrackGroup- type				
TrackSelectionNumber		A Track Selection Number assigned to the group of tracks that belong to the same type, such as normal or commentary tracks.	xs:nonNegativeInteger	

VideoTrackReference		Track Reference to a Video track in TrackMetadata.	xs:string	1..n
	priority	Relative priority of this track.	xs:positiveInteger	
AudioTrackReference		Track Reference to an Audio track in TrackMetadata.	xs:string	1..n
	priority	Relative priority of this track.	xs:positiveInteger	
SubtitleTrackReference		Track Reference to a Subtitle track in TrackMetadata.	xs:string	0..n
	priority	Relative priority of this track.	xs:positiveInteger	
LanguagePair		Defines which audio language and subtitle language are paired with a System Language. Each instance SHALL have a SystemLanguage element. With a unique language.	mddece:ContainerLanguagePair-type	0..n
AncillaryTrackReference		Track Reference to Ancillary Track in TrackMetadata.	xs:string	0..n
	priority	Reserved for proprietary, Ancillary Track specific priority	xs:positiveInteger	0..1

Within `VideoTrackReference`, `AudioTrackReference` and `SubtitleTrackReference`, the `priority` attribute is the relative priority of the track. A smaller number is a higher priority, with '1' being the highest priority.

Within a ContainerTrackGroup-type instance, each VideoTrackReference/priority child SHALL be unique.

Within a ContainerTrackGroup-type instance, each AudioTrackReference/priority child SHALL be unique.

Within a ContainerTrackGroup-type instance, each SubtitleTrackReference/priority child SHALL be unique.

Each TrackSelectionNumber represents a selection of tracks that belong to the same type. For example, primary audio tracks and normal subtitle tracks are associated with TrackSelectionNumber ='0',

director's commentary audio tracks and subtitle tracks are associated with TrackSelectionNumber ='1', and so on.

Audio tracks of type 'primary' and subtitle tracks of Type 'normal' SHALL be associated with TrackSelectionNumber='0'.

VideoTrackReference, AudioTrackReference and SubtitleTrackReference elements, lists the track priority order for all video, audio and subtitle tracks associated with the TrackSelectionNumber. All tracks associated with a lower TrackSelectionNumber are higher priority than all tracks associated with a higher TrackSelectionNumber.

The priority attribute can be used to specify priority order amongst equivalent tracks. For example, given multiple AudioTrackReference instances that reference primary English tracks with different CODECs, the preferred order of these tracks would be indicated by the priority attributes, with the most preferred track having priority='1' . If there are multiple instances of SubtitleTrackReference elements for equivalent tracks with different Track/FormatTypes (Text or Image), authors can specify which FormatType has higher priority using the priority attribute. Within a TrackGroup, Priority is unique across all audio tracks and is unique across all subtitle tracks.

Note that CFF currently only allows one video track, so it is not meaningful to have more than one VideoTrackReference (i.e., a cardinality of 1). The schema allows multiple instances to support future growth.

4.1.5.3.1 AncillaryTrackReference

AncillaryTrackReference elements reference Ancillary Tracks associated with tracks within a Presentation.

Ancillary Tracks are tracks that are not in themselves playable, but provide additional information for other tracks. All Ancillary tracks are bound to a specific track audio, video, subtitle track in the same Presentation. Although no use case is currently identified for the use of Ancillary Tracks for images and interactive, its use is not prohibited.

Players that do not know how to interpret an Ancillary Track are expected to ignore that Ancillary Track.

Ancillary Track Metadata is defined in [CM], particularly Section 5.2.12.

4.1.5.4 ContainerLanguagePair-type

ContainerLanguagePair-type allows the author to specify audio and subtitle track pairs based on a User's System Language.

Content Metadata Specification Version 2.2

A User preference for System Language does not always imply audio and subtitle tracks of the same language. For example, in some cases the best choice for a Japanese viewer would be Japanese language audio and no subtitle. In other cases, the best choice would be an English audio track and a Japanese subtitle.

TrackGroup/AudioReference and TrackGroup/SubtitleReference refer to a subset of tracks in TrackMetadata/Track/Audio and TrackMetadata/Track/Subtitle respectively. ContainerLanguagePair-type further constrains the track list by selecting tracks by language. That is, LanguagePair refers only to audio tracks where TrackMetadata/Track/Audio/Language equals AudioLanguage and to subtitle tracks where TrackMetadata/Track/Subtitle/Language equals SubtitleLanguage.

Element	Attribute	Definition	Value	Card.
ContainerLanguagePair -type				
SystemLanguage		The language scope for which the Language Pair applies. For example, if this element is 'en-US' then the Language Pair element applies to English spoken in the United States.	xs:language	
AudioLanguage		Author recommended audio language for given SystemLanguage	xs:language	
SubtitleLanguage		Author recommended subtitle language for given SystemLanguage	xs:language	

Within the set of `LanguagePair` elements, each `LanguagePair` element SHALL have a unique value in `SystemLanguage`.

4.2 Container Optional Metadata

Optionally, detailed metadata can be included in the DECE Container.

Container Optional Metadata MAY include DECE Container Optional Metadata.

Container Optional Metadata MAY include one or more of DECE Alternative Optional Metadata.

If both DECE Container Optional Metadata and DECE Alternative Optional Metadata are included, DECE Container Optional Metadata SHALL be first.

Optional Metadata SHALL not exceed 256×2^{10} (256K) bytes.

4.2.1 DECE Container Optional Metadata

DECE Container Optional Metadata is a well formed XML document with a `MetadataTail` root element.

DECE Container Optional Metadata SHALL be in conformance with Common Metadata Derived Types, Section 3 above. Additional metadata elements MAY be included.

Element	Definition	Value
MetadataTail	Optional metadata that may be included at the end of a DECE Container.	mddece:ContainerSupplementalMetadata-type

`ContainerSupplementalMetadata-type` allows up to one instance of DECE metadata and optionally metadata in other forms. If elements with this type are included, at least one metadata (i.e., `DECE`, `Alternate` or both) SHALL be included.

Element	Attribute	Definition	Value	Card.
ContainerSupplemental Metadata-type				
	MetadataVersion Reference	A string that defines the version of the metadata in this element. If the metadata changes, this string SHOULD be included and unique relative to other instances of this attribute.	xs:string	0..1

DECE		Detailed metadata optionally included in a Container.	mddece:Container OptionalMetadata-type	0..1
Alternate		Detailed non-DECE metadata optionally included in a Container.	mddece:Alternate OptionalMetadata-type	0..n

4.2.1.1 ContainerOptionalMetadata-type

ContainerOptionalMetadata-type is defined as follows. When used for DECE, metadata SHOULD meet the minimum requirements of Section 3.2.

Element	Attribute	Definition	Value	Card.
ContainerOptionalMetadata -type				
Basic		Basic Metadata as defined in Common Metadata, [DCMETA], Section 4.	md:BasicMetadata-type	
DigitalAsset		Digital Asset Metadata as defined in Common Metadata, [DCMETA], Section 5.	md:DigitalAssetMetadata-type	1..n

4.2.2 DECE Container Alternate Metadata

Alternative Optional Metadata takes the form of the of the AlternativeOptionalMetadata element as defined here.

Element	Attribute	Definition	Type	Card.
AlternateOptionalMetadata-type		Other metadata		
Namespace		Namespace to identify the alternative metadata	xs:string	

(any)		Alternate metadata. Structure is not defined by DECE.	xs:any	

Namespace identifies the metadata used. It should clearly identify a metadata scheme such that someone familiar with that scheme will be able to interpret the elements. As guidance, it suggested the namespace be a string, all in lowercase, that constitutes a common name for that metadata; for example, 'ema' or 'pbcore'.

4.3 Image References

Metadata images internal to a DCC SHALL be referenced using a URN, as per [RFC2141] of the form:

    urn:dece:container:metadataimageindex:<index>.<ext>

where

- <index> is the item_ID value as expressed in the 'iloc' Box defined in [DMedia] 2.1.2 referring to the image in question

- <ext> is a file extension associated with the image type (e.g., "png")

Images external to a DCC, but not in a CMP, SHALL be referenced using a URL as per [RFC3986] with the actual location of the image. Typically, the image reference URL will be of scheme 'http' or 'ftp'.

Images in a CMP as per [DDMP] SHALL be referenced using Part Name.

4.4 Base Locations

The BaseLocations element is equivalent to the Base Location Box 'bloc' in [DMedia], Annex D. Note that while 'bloc' must be fixed size, BaseLocations does not require padding. The BaseLocations element is defined as ContainerBaseLocations-type.

Element	Attribute	Definition	Type	Card.
ContainerBaseLocations-type				
BaseLocation		The Base Location defined in Section 8.3.2 of [DSystem].	xs:string	

BasePurlLocation		The Base Purl Location as specified in Section 8.3.3 of [DSystem].	xs:string	0..1

4.5 Experience Media Application

ExperienceMediaApp data is a Media Application for inclusion in a Common Media Package (CMP) as defined in [DDMP]. The element ties together the various components of the CMP and creates an experience as defined in the Common Media Manifest Metadata [DCManifest].

ExperienceMediaApp-type SHALL comply with [DCManifest] except where noted.

Element	Attribute	Definition	Value	Card.
ExperienceMediaApp-type				
	Experience MediaAppID	As defined for ManifestID in [DCManifest].	As defined for ManifestID in [DCManifest]	
	updateNum	As defined in [DCManifest]	As defined in [DCManifest]	
PictureGroups				0..1
AppGroups				0..1
Experiences				

4.5.1 Experience Constraints

This section describes constraints on Experiences/Experience elements.

Experience/Audiovisual elements SHALL include a PresentationID or a PlayableSequence element. That is, PlayableSequenceID references are not allowed.

If there is exactly one Experience/Audiovisual element in the Experience element, Experience/BasicMetadata SHALL be included and Experience/Audiovisual/BasicMetadata SHALL be omitted.

4.5.2 Experience Navigation

Experience navigation starts with the first Experience instance in the ExperienceMediaApp/Experiences element.

4.5.3 Playable Sequence

Annex B defines the assumed media encoding and playback when the @seamless attribute of Experience/Audiovisual/Presentation/PlayableSequence equals 'true'.

4.6 Experience Media Stream Application

ExperienceMediaStream data is an Application for use in Common Streaming [DStream]. The element ties together the various components of the CMP and creates an experience as defined in the Common Media Manifest Metadata [DCManifest]. Note that the streaming version requires Presentations definitions because that is where Adaptation Sets are defined. It also includes information for Default Track Selection.

ExperienceMediaApp-type SHALL comply with [DCManifest] except where noted.

Element	Attribute	Definition	Value	Card.
ExperienceStreamApp -type				
	Experience MediaAppID	As defined for ManifestID in [DCManifest]	As defined for ManifestID in [DCManifest]	
	updateNum	As defined in [DCManifest]	As defined in [DCManifest]	
Presentations				
PictureGroups				0..1
AppGroups				0..1
Experiences				

4.6.1 Experience Constraints

This section describes constraints on Experiences/Experience elements.

Experience/Audiovisual elements SHALL include a PresentationID or a PlayableSequence element. That is, PlayableSequenceID references are not allowed.

If there is exactly one Experience/Audiovisual element in the Experience element, Experience/BasicMetadata SHALL be included and Experience/Audiovisual/BasicMetadata SHALL be omitted.

4.6.2 Experience Navigation

Experience navigation starts with the first Experience instance in the ExperienceMediaApp/Experiences element.

This section defines metadata structures describing an Original Common Media Package as defined in [DDMP].

An OCMP is self-defining in the sense that all data necessary to understand that OCMP is within the OCMP's data. However, it is sometimes necessary to understand the structure of an OCMP prior to gaining access to that OCMP. The structures in this section carry those data.

5.1 OCMP-type

Top-level definition of the OCMP. It identifies the OCMP and describes its structure.

Element	Attribute	Definition	Value	Card.
OCMP-type		Physical metadata for an asset		
	CMPID	CMP Identifier associated with the described CMP.	md:AssetPhysicalID	
	Version	OCMP Version as per [SMPTE2053]	xs:nonNegativeInteger	
Presentation		A description of a Presentation within CMP. An instance SHALL exist for each Presentation within the CMP.	mddece:OCMPPresentation-type	1..n
Application		A description of a Media Application within CMP. An instance SHALL exist for each Application within the CMP.	mddece:OCMPApplication-type	0..n
OtherPart		LocalSource of other Parts within the CMP that are not associated with Presentations referenced by Presentation element, or Applications reference by the Application element. There is one instance for each Part.	xs:string	0..n
	isPresent	Indicates whether this Part is present in the OCMP.	xs:boolean	0..1
	forceDownload	Indicates whether the download manager SHALL download the Part with LocalSource immediately following OCMP download.	xs:boolean	0..1
OtherContainer		Any other Containers within the CMP.	md:ContainerMetadata-type	0..n

Content Metadata Specification Version 2.2

5.1.1 OCMPPresentation-type

Describes Presentations within an OCMP.

Element	Attribute	Definition	Value	Card.
OCMPPresentation-type		Definition of Presentations within an OCMP		
	PresentationID	Presentation ID associated with this Presentation	md:AssetLogicalID-type	
	Version	Presentation Version as per [SMPTE2053]	xs:nonNegativeInteger	
MediaProfile		The DECE Media Profile associated with this Presentation	mddece:AssetProfile-type	
APID		APID for each DCC associated with CMP, whether or not that DCC is in the OCMP.	md:AssetPhysicalID-type	1..n
	isPresent	Indicates whether the DCC associated with this APID is present in the OCMP.	xs:boolean	
	forceDownload	Indicates whether the download manager SHALL download the DCC associated with this APID immediately following OCMP download.	xs:boolean	0..1
DependentPart		LocalSource of other Parts other than APIDs associated with this Presentation.	xs:string	0..n
	isPresent	Indicates whether the this Part is present in the OCMP.	xs:boolean	0..1
	forceDownload	Indicates whether the download manager SHALL download the Part with LocalSource immediately following OCMP download.	xs:boolean	0..1

5.1.2 OCMPApplication-type

Describes Media Applications within an OCMP.

Element	Attribute	Definition	Value	Card.
OCMPApplication-type		Definition of Media Applications within an OCMP		
	Version	Media Application Version as per [SMPTE2053]	xs:nonNegativeInteger	

Content Metadata Specification Version 2.2

Element	Attribute	Definition	Value	Card.
	ApplicationID	ApplicationID for Media Application, whether or not that Media Application element is in the OCMP.	md:AssetPhysicalID-type	
	isPresent	Indicates whether the Media Application is present in the OCMP.	xs:boolean	
	forceDownload	Indicates whether the download manager SHALL download the Media Application immediately following OCMP download.	xs:boolean	0..1
DependentPart		LocalSource of Parts other than Media Application (Media Application components) associated with this Application.	xs:string	1..n
	isPresent	Indicates whether this Part is present in the OCMP. This attribute is deprecated and should not be included.	xs:boolean	0..1
	forceDownload	Indicates whether the download manager SHALL download the Part with LocalSource immediately following OCMP download. This attribute is deprecated and should not be included.	xs:boolean	0..1
PresentationID		Each Presentation associated with this Media Application. If absent, Media Application is associated with all Presentations in the CMP.	md:AssetLogicalID-type	0..n

DependentPart/@isPresent and DependentPart/@forceDownload are redundant with @isPresent and @forceDownload and therefore have been deprecated. These should not be used. The correct behavior is for all dependent parts to inherit the @isPresent and @forceDownload from the OCMPApplication-type.

Content Metadata Specification Version 2.2

Annex A   Track Selection Process

This section describes the intended use of Track Selection Data as described in Section 4.1.5.

The following stages occur in track selection:

1. The Device assigns a default System Language

2. A User optionally changes System Language; and may selects preferences such as audio and subtitle languages, and subtitle type

3. The Device selects default audio track and subtitle track (Primary Subtitling Presentation Track), if applicable

4. A User may optionally select specific audio track or subtitle track (Primary Subtitling Presentation Track)

5. The Device selects subtitle tracks for forced subtitles (Alternate Subtitling Presentation Track), if applicable

6. Playback can begin. User selections may require repeating some steps above. For example, changing tracks (Step 4) would require performing Step 5.

This Annex uses the following terminology:

- The following subtitle definitions are used to describe what is in a subtitle track

    - Forced Subtitle: A subtitle with only one instance of MetadataMovie/TrackMetadata/Track/Subtitle/Type where that instance equals 'forced'.

    - Other Subtitle: A subtitle with no instances of MetadataMovie/TrackMetadata/Track/Subtitle/Type equal to "forced"

    - Mixed Subtitle: A subtitle with at least one instance of MetadataMovie/TrackMetadata/Track/Subtitle/Type equal to 'forced'; and at least one instance of Metadata/TrackMetadata/Track/Subtitle/Type not equal to "forced"

        - Within a Mixed Subtitle track, subtext and subpicture elements that are to be displayed as forced subtitles are referred to as 'forced elements' and elements that are not to be displayed as forced elements are referred to as 'non-forced elements'

- From a User's perspective, subtitles are either "on" or "off", however, in both cases subtitle elements may be displayed. The following definitions indicate what subtitles elements are presented when subtitles are off and on, what tracks contain those elements, and what audio track contains audio for playback

    o Primary Subtitling Presentation Mode: corresponds to subtitles are "on". When in Primary Subtitling Presentation Mode, the Primary Subtitling Presentation Track will be presented.

    o Primary Subtitling Presentation Track: The subtitle track that is to be presented during Primary Subtitling Presentation. An Other Subtitle track or a Mixed Subtitle track will be decoded and presented during Primary Subtitling Presentation.

    o Alternate Subtitling Presentation Mode: corresponds to subtitles are "off". When in Alternate Subtitling Presentation Mode, only forced elements within the Alternate Subtitling Presentation Track will be presented (if any). An Alternate Subtitle can be forced subtitle elements within a Mixed Subtitle track or a Forced Subtitle track.

    o Alternate Subtitling Presentation Track: The subtitle track that includes the forced subtitle elements to be presented during Alternate Subtitling Presentation. Forced subtitle elements within a Mixed Subtitle track or all elements in a Forced Subtitle track will be presented during Alternate Subtitle Presentation. Note that for a Mixed Track, the Selected Primary Subtitle Track and the Selected Alternate Subtitle Track might be the same track.

- The following definition indicates what audio track contains audio for playback

    o Selected Audio Track: The audio track selected for play.

A.1. Defined Preferences

The following are Input Variables to default track selection and must be selected prior to default track selection.

- System Language (required)

- User Preferred Audio Type. The type of audio preferred by the user. Type enumeration is as per md:DigitalAssetAudioData-type/Type. By default this should be "primary"

- User Preferred Audio Language (optional) – User preference for audio language which applies to all DCCs

- User Preferred Subtitle Language (optional) – User preference for subtitle language which applies to all DCCs

- User Preferred Subtitle Type (optional) – The type of subtitle preferred by the User for the purposes of selecting default audio and subtitle tracks. Type enumeration is as per md:DigitalAssetSubtitleData-type/Type. By default this should be 'normal'.

Devices are assumed to have the following capabilities

- Allow a User to override Input Variables

- Allow a User to select a specific audio track

- Allow a User to select a specific subtitle track for Primary Subtitling Presentation

- Allow a User to turn "on" and "off" subtitles

  - When "On", decode and present the Primary Subtitling Presentation Track and display all forced and non-forced elements.

  - When "Off": decode and present the Alternate Subtitling Presentation Track and only display forced elements

A.2. Default Audio and Subtitle Track Selection

This section defines algorithms for selecting default audio track and default subtitle track.

Default tracks are selected prior to initial playback and prior to User's making specific tracks selections.

The following rules apply to the decision flow:

- When matching and selecting tracks, only tracks that are playable on the Device should be considered. Tracks that are not playable should be ignored. For example, a track with a CODEC not supported by the Device would never be selected.

- When multiple elements match equivalently

  - If there are additional User preference and at least one element matches this preference, filter elements based on the User preferences. For example, if the user prefers original audio tracks, and an original audio track matches other criteria, select that track.

 - Then, If elements are prioritized, return the element with the highest priority;

- o   Otherwise, return the element that appears first in the metadata. For example, if a language lookup matches two LanguagePairs equally well, the first LanguagePair to appear in the TrackGroup would be selected.

- If more than one `TrackGroup` element is present, the `TrackGroup` element with `TrackSelectionNumber` equal to 0 is referenced for automatic default track selection.

- In the diagrams, when an audio track is "selected" it is selected as the Selected Audio Track. When a subtitle track is selected, it is selected as a Selected Primary Subtitle Track, unless otherwise noted.

- In conditions referring to matching tracks of a given language, TrackMetadata/Track/Audio/Language is used for audio language matching and TrackMetadata/Track/Subtitle/Language is used for subtitle language matching.

- In conditions referring to matching tracks of a given type TrackMetadata/Track/Audio/Type is used for audio Type matching, and TrackMetadata/Track/Subtitle/Type is used for subtitle Type matching.

- When referring to Tracks referenced by LanguagePair this refers to all tracks referenced by TrackGroup/AudioTrackReference that match TrackMetadata/Track/Audio/Language in union with tracks referenced by TrackGroup/SubtitleTrackReference that match TrackMetadata/Track/Subtitle/Language.

A.2.1.   Default Audio Track Selection

This flow describes the assumed algorithm for selecting a Default Audio Track.
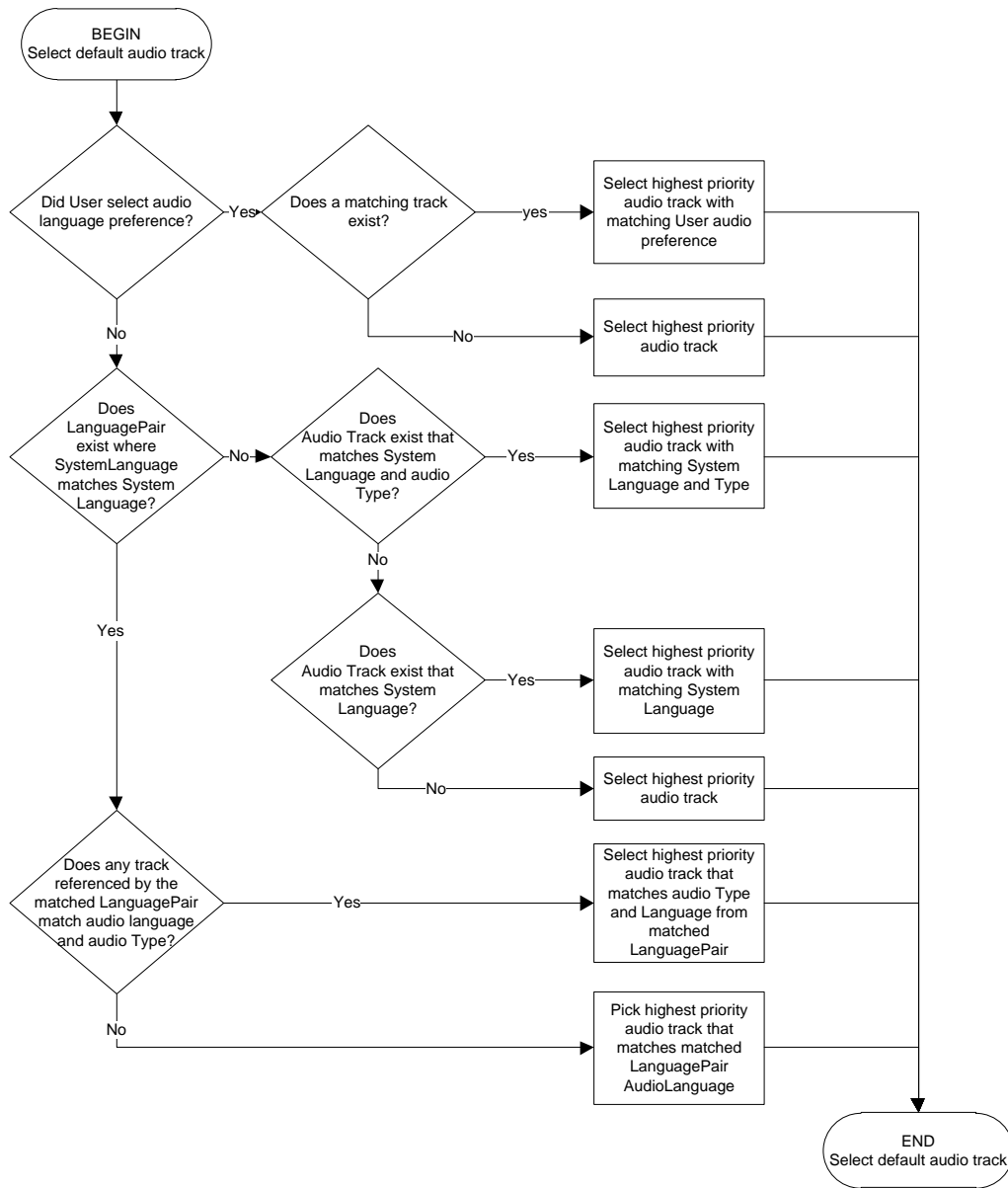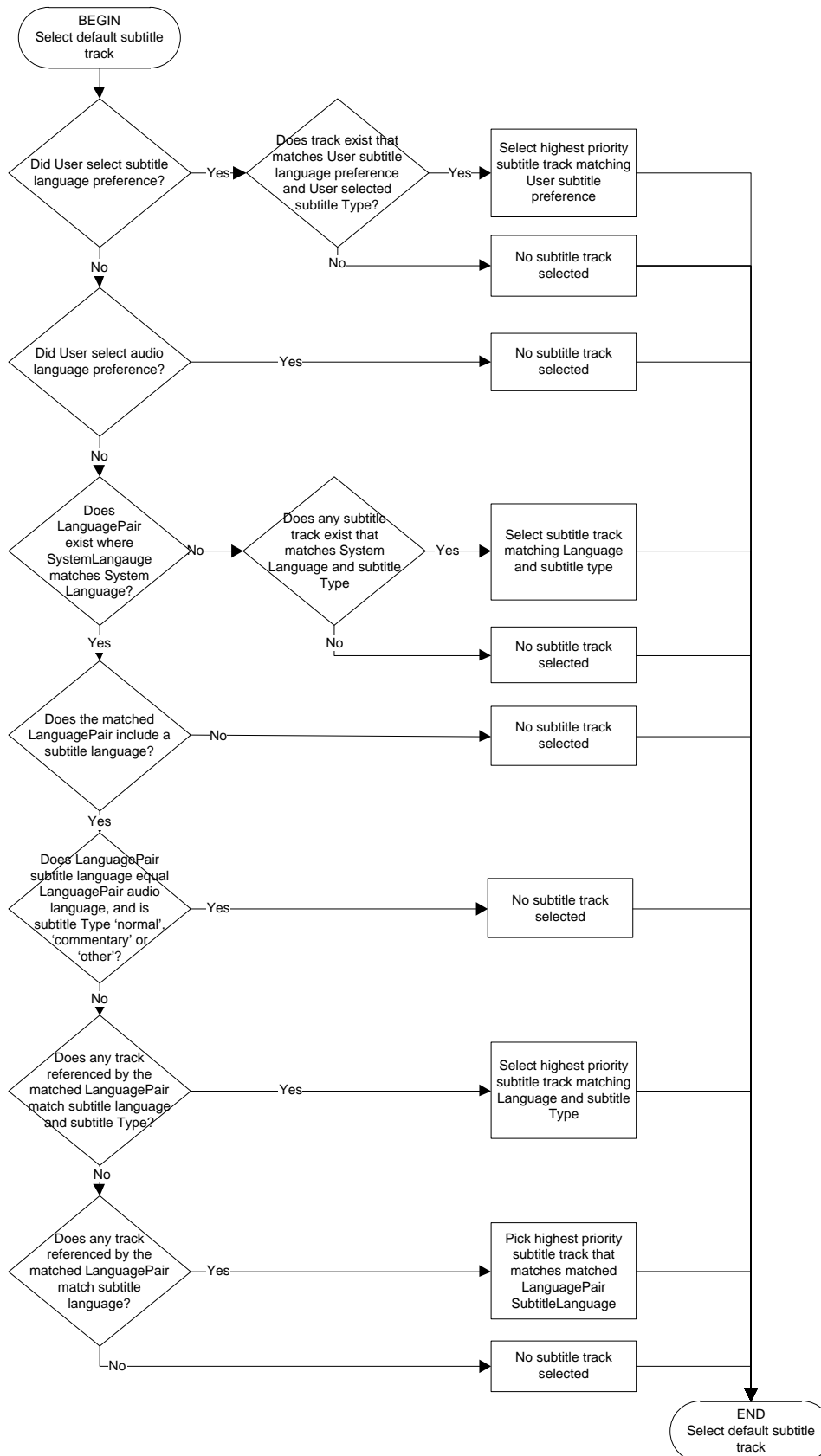
A.2.2.   Default Primary Subtitling Presentation Track Selection

This flow describes the assumed algorithm for selecting a Default Subtitle Track.

Content Metadata Specification Version 2.2

```
                    BEGIN
              Select default subtitle
                    track
                       │
                       ▼
```

Did User select subtitle language preference? ──Yes──▶ **Does track exist that matches User subtitle language preference and User selected subtitle Type?** ──Yes──▶ Select highest priority subtitle track matching User subtitle preference

│No   │No ──▶ No subtitle track selected

Did User select audio language preference? ──Yes──▶ No subtitle track selected

│No

Does LanguagePair exist where SystemLangauge matches System Language? ──No──▶ **Does any subtitle track exist that matches System Language and subtitle Type** ──Yes──▶ Select subtitle track matching Language and subtitle type

│Yes   │No ──▶ No subtitle track selected

Does the matched LanguagePair include a subtitle language? ──No──▶ No subtitle track selected

│Yes

Does LanguagePair subtitle language equal LanguagePair audio language, and is subtitle Type 'normal', 'commentary' or 'other'? ──Yes──▶ No subtitle track selected

│No

Does any track referenced by the matched LanguagePair match subtitle language and subtitle Type? ──Yes──▶ Select highest priority subtitle track matching Language and subtitle Type

│No

Does any track referenced by the matched LanguagePair match subtitle language? ──Yes──▶ Pick highest priority subtitle track that matches matched LanguagePair SubtitleLanguage

│No ──▶ No subtitle track selected

```
                    END
              Select default subtitle
                    track
```

A.3. Alternate Subtitling Presentation Track Selection

An Alternate Subtitle Track is used for Forced Subtitles.

Forced subtitles are displayed either in conjunction with other subtitles, or when subtitles are turned off.   That is, if subtitle is off and a suitable forced subtitle track (i.e., either a Forced Subtitle track or a Mixed Subtitle Track) is present, it will be displayed.

A forced subtitle track is expected to match the language of a selected audio track.

If a subtitle track contains information that allows differentiation between elements that are forced and not forced, then the forced subtitle track should be interpreted as the mixed track with only forced elements presented.

A.3.1. Select Alternate Subtitle Track

This flow describes the assumed algorithm for selecting the Alternate Subtitle Track.
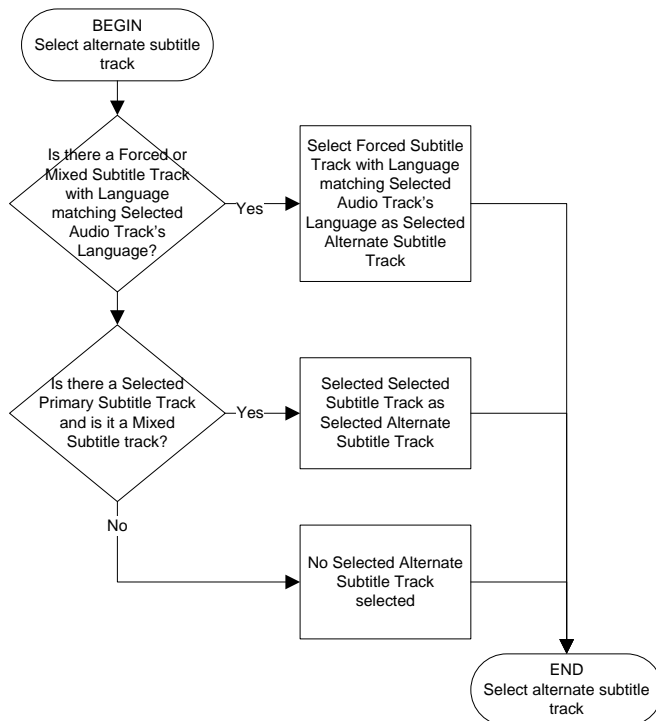
Annex B   Seamless Sequential Playback Model

This section describes the seamless playback model.

Seamless sequential playback allows two or more Presentations or sections of a Presentation (Clip) to play in sequence without interruption.

Although some models require the simultaneous decoding of two audio tracks to ensure seamless playback, this model does not.

If properly implemented, this model will maintain A/V sync to within the duration of an audio frame over the course of the entire seamless playback portion of a Playable Sequence. Audio will also never precede video, a phenomenon that is disconcerting to a viewer.

B.1. Summary

The approach described here allows Clips within a Playable Sequence to be played seamlessly with audio never playing before the associated video and never more than one audio frame late. This approach depends on properly authored clips.

B.2. Definitions

The terms *Playable Sequence*, *Presentation* and *Clip* are defined in Media Manifest, found in [DCManifest].

This section addresses Clips within a Playable Sequence that are flagged for seamless playback with the Clip/@seamless attribute set to 'true'.

In this model, the following diagram shows nomenclature related to a Clip.

Audio Frame	A
Video Frame	V
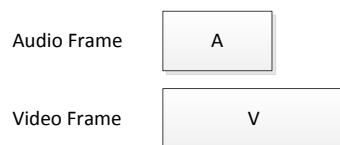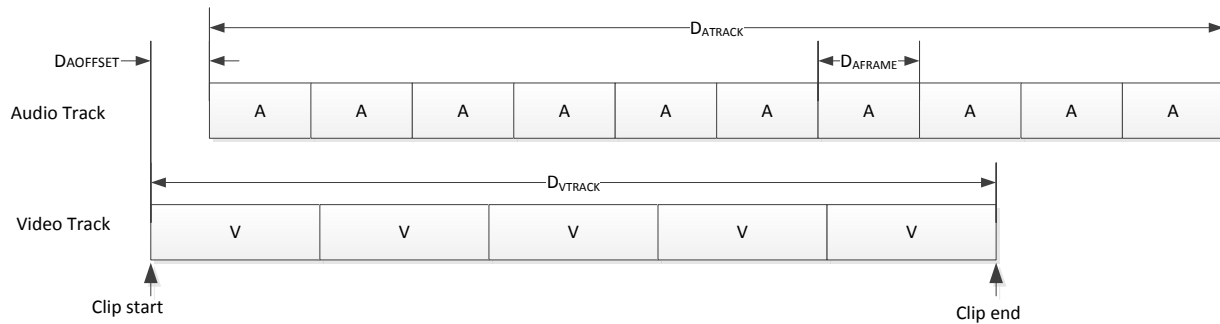
Parameters are expressed as units of time, although the exact unit is not relevant.

The following parameters are used

- D_{ATRACK} is the duration of the entire audio track
- D_{VTRACK} is the duration of the entire video track. Note that the start and end times of the Clip are defined by the first and last video frames respectively.
- D_{AFRAME} is the duration of each audio frame in the audio track. Assumed to be constant.

$D_{AOFFSET}$ is the initial offset of audio in a Clip. If Clips start simultaneously with video, $D_{AOFFSET} = 0$. Although not shown in the illustration, $D_{AOFFSET}$ can be longer than D_{AFRAME}. A Fragment contains one or more sequential Frames. An Audio Fragment is a Fragment consisting of Audio Frames. A Video Fragment is a Fragment consisting of Video Frames. We use the following definition:

- $D_{AFRAGMENTMAX}$ is the maximum possible duration of an Audio Fragment. It is assumed that maximum possible length is known.

The following diagram defines nomenclature related to playing Clips in sequence. The following shows two Clips (1 and 2) playing in sequence seamlessly. The *Clip Boundary* is the point at which the one Clip's video ends and the next Clip's video begins.
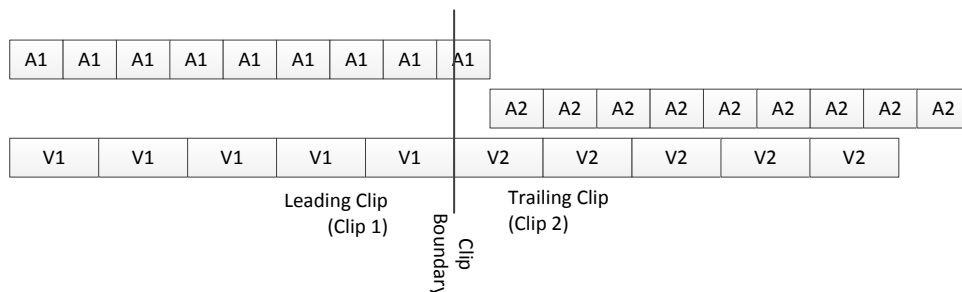
The first of successive Clips in a seamless portion of a Playable Sequence is called the *Leading Clip*. The second of successive Clips in a seamless portion of a Playable Sequence is called the *Trailing Clip*. In the diagram, Clip 1 is the Leading Clip and Clip 2 is the Trailing Clip.

Content Metadata Specification Version 2.2

A *Splice* is the point at which two Clips meet in a Playable Sequence. This refers both the Clip Boundary and the point where the Leading Clip audio meets the Trailing Clip audio.

If Content is authored such that $D_{AOFFSET}$ is less than D_{AFRAME}, the Splice is said to be *Conditioned*. Otherwise, the Splice is said to be *Unconditioned*.

For this discussion, the following parameters are defined here:

- D_{EXCESS} is the duration a Leading Clip audio must play past the start of the Trailing Clip audio to finish at the end of a frame without a gap.
    - If playback of an audio frame of the Leading Clip ends at the Trailing Clip's $D_{AOFFSET}$, D_{EXCESS} is 0.
    - If playback of an audio frame of the Leading Clip ends less than one frame duration (D_{AFRAME}) following the Trailing Clip's $D_{AOFFSET}$, D_{EXCESS} is the time between $D_{AOFFSET}$ and the end time of that frame.
    - Note
        - Encoding rules require that the Leading Clip has enough audio frames to never end before $D_{AOFFSET}$ of the Trailing Clip.
        - There is no implication that the Leading Clip needs to be played past the time associated with $D_{AOFFSET}$ of the Trailing Clip.

Note that in the following illustrations, labels such as 'Clip 1', 'Clip 2', etc. are just names for those clips, and many additional frames would be included in each Clip.

The following illustrations show two cases of Conditioned Splices. In the first clip, called 'Clip 1', D_{EXCESS} is 0. In the second, Clip 3 has $D_{EXCESS}>0$ meaning there is some audio that will play past the end of Clip 3.
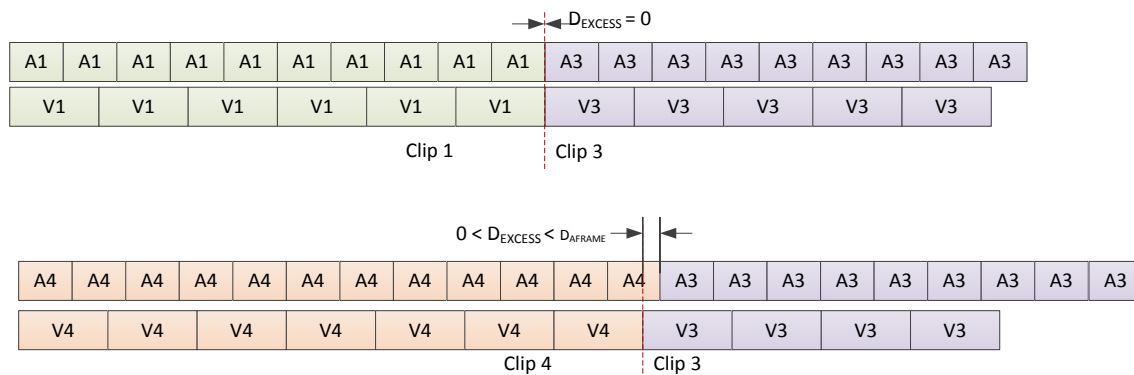
The following illustrations show two cases of Unconditioned Splices. $D_{AOFFSET}$ is greater than 0. In the first illustration, the Clip 4 audio frame ends exactly when Clip 5 audio starts (after $D_{AOFFSET}$ delay), so D_{EXCESS} is 0.
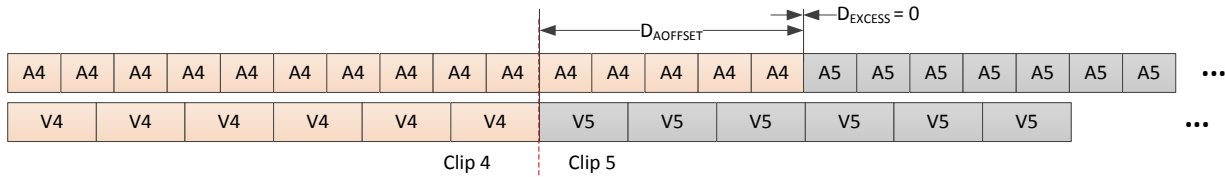
In the following case, the start of Clip 5 does not align with the audio frame boundary for Clip 4. Therefore, $D_{EXCESS} > 0$. Note that only the partial A4 frame contributes to the excess as extra full frames will be dropped. Additional details on this scenario are provided in Examples (Section B.5).
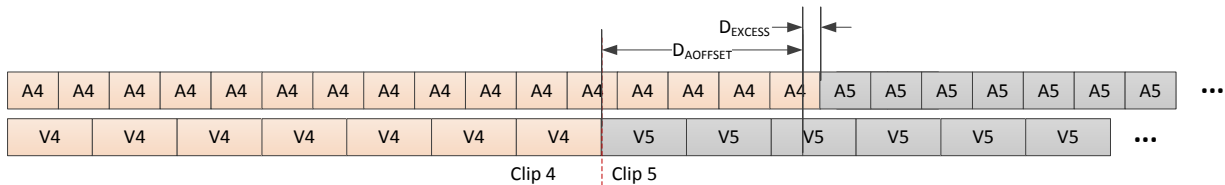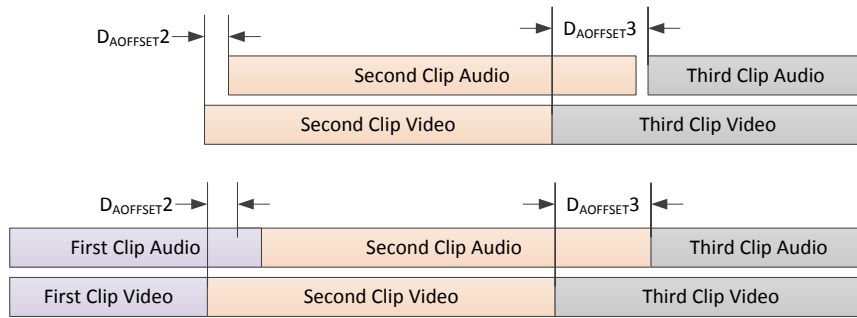

B.3.   Track Constraints

This section describes Content authoring constraints.

Audio and Video Tracks in all Clips in a seamless portion of a Playable Sequence SHALL be constructed such that the end of the audio track in a Leading Clip never ends before $D_{AOFFSET}$ of the Trailing Clip. The audio track could need to be as much as $D_{AFRAGMENTMAX}$ longer than D_{VTRACK}. The duration additional audio that must be authored into the Leading Clip depends on how the Splice is authored. If the Trailing Clip audio is Conditioned, the Leading Clip audio need not be more than one audio frame longer than the Leading Clip video.

If the Trailing Clip is Unconditioned, the exact length calculation depends on $D_{AOFFSET}$ of both the Leading and Trailing Clips, and possibly the delay introduce by the previous Leading Clip ($<D_{AFRAME}$). In the following illustrations, "Second Clip Audio" does not appear to be long enough. However, when played in sequence "First Clip Audio" shifts "Second Clip Audio". In this illustration, that was sufficient to fill the gap.

Audio track length of $D_{VTRACK} + D_{AFRAGMENTMAX}$ will always satisfy this requirement. This is a worst-case calculation.

Note that this constraint is true regardless of the start time of the audio relative to the video and could cause there to be audio frames that play after the end of video. It is desirable to keep $D_{AOFFSET}$ small, but it is possible that the beginning of clip the beginning of the first full Audio Fragment follows the Video Fragment by more than D_{AFRAME}. Consequently, when authoring a Leading Clip, it is essential that the track contains enough audio to play out long enough for the Trailing Clip audio to begin on or after the point where the Trailing Clip's audio is in sync with the video. For example, if the Trailing Clip's audio begins 1 second after the video, the Leading Clip must have at least one second of audio following the Clip Boundary. Audio and Video Tracks in all Clips in a seamless portion of a Playable Sequence SHALL be of the same Media Profile and Delivery Target as defined in [DMedia] Annex A-C; and SHALL have all the same ecosystem-specific constraints, such as defined in [DMedia], Annex E.

All tracks of a given type (i.e., audio, video, timed text) in a seamless portion of a Playable Sequence SHALL have functionally equivalent DCC Headers. That is, even when tracks for Clips are extracted from different Presentations, the sequence of track fragments for a track use same codec parameters, same track_ID and so on.

The audio track within a Clip SHALL start less than one audio fragment duration ($D_{AFRAGMENTMAX}$) after the correct time to be synchronized with video (i.e., $D_{AOFFSET}$ after video start). Audio and video can start simultaneously. Audio cannot start before video.

The audio track within a Clip SHOULD start less than one audio frame duration after video start within the same clip. A clip meeting this requirement results in a Conditioned Splice. Otherwise, it results in an Unconditioned Splice.

When defining a Clip in Media Manifest,

- Clip/EntryPointTimecode for an Audiovisual Clip SHALL be the start time of the first DCC Movie Fragment of the video track DCC for the Clip.

- Clip/ExitPointTimecode for an Audio Visual Clip SHALL be the end time of the last DCC Movie Fragment of the video track DCC for the Clip.

Note that Timed Text tracks are tied to video frames so events will continue to start and end with the associated video frame.

Note that some authors have found it beneficial to fade audio to silence at the end of each clip to avoid possible audible artifacts. Authors should select clip boundaries with discretion.

B.4. Player Behavior

The player behavior is as follows.

A player SHALL begin playback of the Trailing Clip video with the first video frame of the Trailing Clip, starting immediately following the Leading Clip video.

A player SHALL continue playback of the Leading Clip audio before starting playback of the Trailing Clip audio, for the duration of $D_{AOFFSET}$ of the Trailing Clip after the Clip Boundary.

A player SHALL begin playback of the Trailing Clip audio with the first audio frame in the first audio fragment that begins immediately following the first video frame of the Trailing Clip (in the source Presentation).

Note that the player is assumed to synchronize timed text and other synchronized behavior with the Trailing Clip video.

A player SHALL synchronize audio and video such that audio is not advanced on video and audio trails video by no more than D_{AFRAME}.

Best results are achieved when a player synchronizes audio and video as close as possible to the authored timing.

At the start of playback of a Trailing Clip, the player SHOULD synchronize audio and video. That is, synchronization of audio and video in the Trailing Clip SHOULD maintain authored timing regardless of Leading Clip characteristics. However, the player MAY start Trailing Clip audio immediately following the last Leading Clip audio, subject to other synchronization constraints. Note that although the playback algorithm maintains audio-video synchronization be less than or equal to the duration of an audio frame ($D_{EXCESS} \le D_{AFRAME}$) in situations of overlapped timeframes, better synchronization can be maintained by simultaneously decoding audio in the previous clip with audio of following clip.

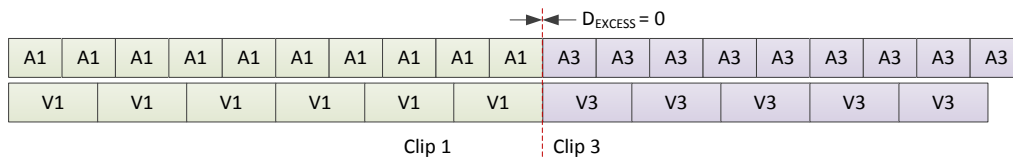The player SHALL calculate D_{EXCESS} in accordance with the definitions.

At the end of each Clip, the player SHALL discard (not play) all audio frames that exceed end of video playback plus $D_{AOFFSET}$ of the Trailing Clip by one frame or more.
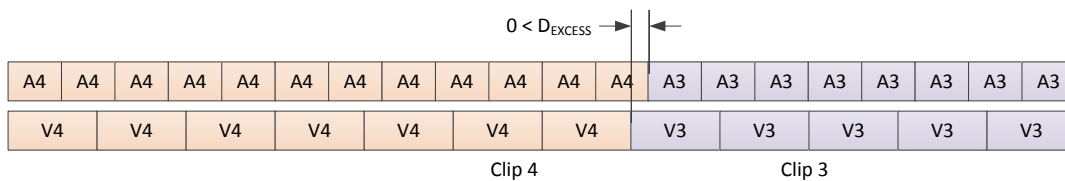
B.5. Examples

The following illustrates three cases of Clip boundary in a seamless portion of a Playable Sequence.
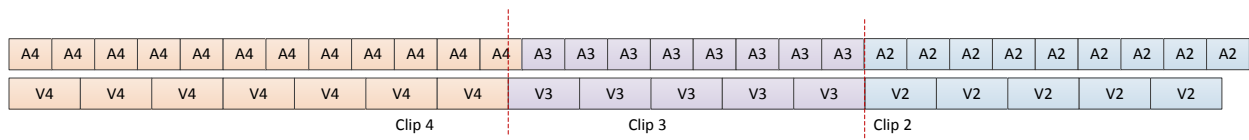
In the following case, an audio frame ends simultaneously with the last video frame. All following audio tracks are discarded. At the end of Clip 3, $D_{EXCESS} < D_{AFRAME}$, so the last frame shown is played, and subsequent frames are discarded.

$D_{EXCESS} = 0$

| A1 | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A1 | A3 | A3 | A3 | A3 | A3 | A3 | A3 | A3 | A3 |

| V1 | V1 | V1 | V1 | V1 | V1 | V3 | V3 | V3 | V3 | V3 |

Clip 1 | Clip 3

In the following case, because $D_{EXCESS} < D_{AFRAME}$ the frame of Clip 4 is played as shown and any additional frames are discarded. At the end of Clip 3, $D_{EXCESS}) >= D_{AFRAME}$ the last audio frame of Clip 3 shown in the illustration and any following frames must be discarded before the next Clip can be played.
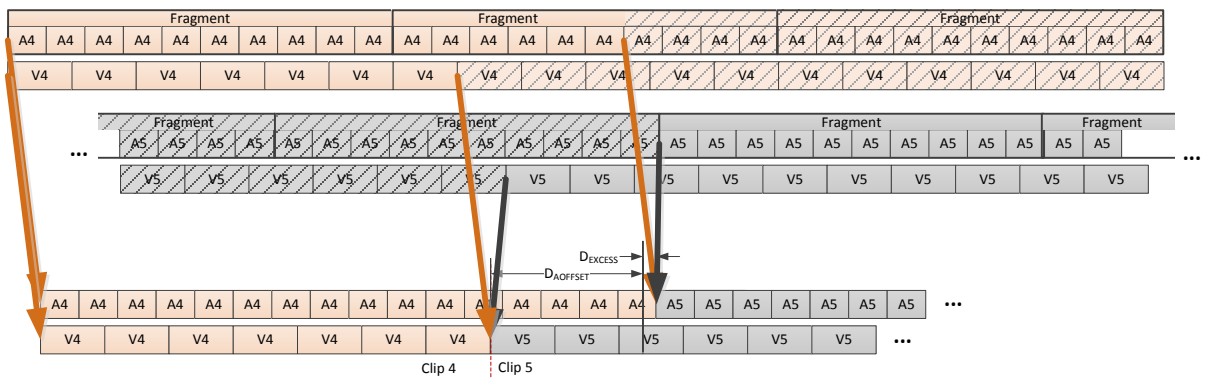
$0 < D_{EXCESS}$

| A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A3 | A3 | A3 | A3 | A3 | A3 | A3 | A3 |

| V4 | V4 | V4 | V4 | V4 | V4 | V4 | V3 | V3 | V3 | V3 | V3 |

Clip 4 | Clip 3

The following example is similar to the example above, but includes another clip:

| A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A4 | A3 | A3 | A3 | A3 | A3 | A3 | A3 | A3 | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2 | A2 |

| V4 | V4 | V4 | V4 | V4 | V4 | V4 | V3 | V3 | V3 | V3 | V3 | V2 | V2 | V2 | V2 | V2 |

Clip 4 | Clip 3 | Clip 2

Following shows additional details on an illustration under definitions. This example shows Clips 4 and 5 in the context of their original streams. Although not shown in the illustration, note that video portions of each Clip are one or more complete DCC Movie Fragments. The portions of the audio streams that are discarded are illustrated with hashed backgrounds. Arrows show points in the original streams and the corresponding points in the final Playable Sequence. Video portions of each Clip are, by definition, one or more complete movie fragments. Notice that A5 begins on a fragment boundary. Notice also that playback of A4 ceases mid Fragment, although the last frame plays. It is preferable for the player to play the transition from A4 to A5 such that A5 is not delayed, although it is recognized this might be necessary. Illustration in this document are not intended to imply the number of frames in a fragment.

END