SCTE | STANDARDS

Digital Video Subcommittee

SCTE STANDARD

SCTE 214-1 2024

MPEG DASH for IP-Based Cable Services
Part 1: MPD Constraints and Extensions

NOTICE

The Society of Cable Telecommunications Engineers (SCTE) Standards and Operational Practices (hereafter called "documents") are intended to serve the public interest by providing specifications, test methods and procedures that promote uniformity of product, interoperability, interchangeability, best practices, and the long term reliability of broadband communications facilities. These documents shall not in any way preclude any member or non-member of SCTE from manufacturing or selling products not conforming to such documents, nor shall the existence of such standards preclude their voluntary use by those other than SCTE members.

SCTE assumes no obligations or liability whatsoever to any party who may adopt the documents. Such adopting party assumes all risks associated with adoption of these documents and accepts full responsibility for any damage and/or claims arising from the adoption of such documents.

NOTE: The user's attention is called to the possibility that compliance with this document may require the use of an invention covered by patent rights. By publication of this document, no position is taken with respect to the validity of any such claim(s) or of any patent rights in connection therewith. If a patent holder has filed a statement of willingness to grant a license under these rights on reasonable and nondiscriminatory terms and conditions to applicants desiring to obtain such a license, then details may be obtained from the standards developer. SCTE shall not be responsible for identifying patents for which a license may be required or for conducting inquiries into the legal validity or scope of those patents that are brought to its attention.

Patent holders who believe that they hold patents which are essential to the implementation of this document have been requested to provide information about those patents and any related licensing terms and conditions. Any such declarations made before or after publication of this document are available on the SCTE web site at https://scte.org.

140 Philips Road
Exton, PA 19341

Document Tags

☒ Specification	☐ Checklist	☐ Facility
☐ Test or Measurement	☐ Metric	☒ Access Network
☐ Architecture or Framework	☒ Cloud	☐ Customer Premises
☐ Procedure, Process or Method		

Document Release History

Release	Date
SCTE 214-1 2015	*8/10/2015*
SCTE 214-1 2016	*2/22/2016*
SCTE 214-1 2022	*4/11/2022*
SCTE 214-1 2024	*4/29/2024*

Note: Standards that are released multiple times in the same year use: a, b, c, etc. to indicate normative balloted updates and/or r1, r2, r3, etc. to indicate editorial changes to a released document after the year.

Table of Contents

List of Figures

Title **Page Number**

List of Tables

Title **Page Number**

1. Introduction

1.1. Executive Summary

This document is part of a suite of documents related to MPEG DASH for IP-Based cable services and is referred to SCTE 214. Part 1 describes general MPD constraints and common features supported by both the DASH TS profile and DASH ISOBMFF profile. Part 2 contains further constraints for the DASH TS profile. Part 3 (*deprecated*) contains further constraints for the DASH ISOBMFF profile. Part 4 provides an instance MPD template for the DASH TS profile. Part 5 which replaces Part 3 (*deprecated*) extends the features and constraints but only for ISOBMFF DASH Profiles.

1.2. Scope

This document describes general media presentation description (MPD) constraints and common features supported by both the DASH TS profile and DASH ISOBMFF profile. This will allow a common feature parity between DASH Profile and ISOBMFF Profile versions of the service and includes multiplexed segments. SCTE 214-1 and [SCTE 214-2] are used together to support DASH TS Profile delivery which is beneficial while transitioning from traditional broadcast MPEG-2 TS delivery structures using an ATS structured stream. Additional features developed in later DASH editions and needed for CABLE IP Services will be supported in SCTE 214-5 but only for constrained DASH ISOBMFF Profiles with non-multiplexed segments. Profile URNs for DASH/TS and DASH/FF appear in SCTE 214-2 and SCTE 214-5.

1.3. Benefits

This document assists in the transition between a broadcast cable infrastructure to a unicast IP based cable services while allowing for a gradual transition of backend systems supporting these services.

1.4. Intended Audience

This document is intended for the development, operations, and maintenance of IP based cable services. It is useful for those creating, ingesting, and delivering content for adaptive streaming services.

1.5. Areas for Further Investigation or to be Added in Future Versions

Additional developments in this document are limited to including additional features that can co-exist with features and constructs define in [DASH] edition 5 that enable parity between DASH MPEG-TS and DASH ISOBMFF [ISOBMFF] profiles which includes the use of multiplexed segments.

2. Normative References

The following documents contain provisions which, through reference in this text, constitute provisions of this document. The editions indicated were valid at the time of subcommittee approval. All documents are subject to revision and, while parties to any agreement based on this document are encouraged to investigate the possibility of applying the most recent editions of the documents listed below, they are reminded that newer editions of those documents might not be compatible with the referenced version.

2.1. SCTE References

[SCTE 35] ANSI/SCTE 35 2023, Digital Program Insertion Cueing Message for Cable

[SCTE 128-1] ANSI/SCTE 128-1 2020, AVC Video Constraints for Cable Television Part 1: Coding.

[SCTE 130-10] ANSI/SCTE 130-10 2020, Digital Program Insertion – Advertising Systems Interfaces, Part 10 – Stream Restriction Data Model (SRDM).

[SCTE 193-1] ANSI/SCTE 193-1 2020, MPEG-4 AAC Family Audio System – Part 1: Coding Constraints for Cable Television.

[SCTE 193-2] ANSI/SCTE 193-2 2020, MPEG-4 AAC Family Audio System – Part 2: Constraints for Carriage over MPEG-2 Transport.

[SCTE 194-1] ANSI/SCTE 194-1 2018, DTS-HD Audio System – Part 1: Coding Constraints for Cable Television.

[SCTE 194-2] ANSI/SCTE 194-2 2018, DTS-HD Audio System – Part 2: Constraints for Carriage over MPEG-2 Transport.

[SCTE 215-1] ANSI/SCTE 215-1 202x, HEVC Video Constraints for Cable Television, Part 1 – Coding.

[SCTE 223] ANSI/SCTE 223 2018, Adaptive Transport Stream.

[SCTE 281-1] SCTE 281-1 2023, VVC Video Constraints for Cable Television Part 1- Coding.

2.2. Standards from Other Organizations

[ATSC A/52] ATSC A/52: 2018 Digital Audio Compression (AC-3) (E-AC-3) Standard.

[ATSC A/53] ATSC A/53 Part 5: 2014 ATSC Digital Television Standard.

[ATSC A/65] ATSC A/65:2013 ATSC Standard: Program and System Information Protocol for Terrestrial Broadcast and Cable.

[DASH] ISO/IEC 23009-1:2021 5th Ed., Information technology – Dynamic adaptive streaming over HTTP (DASH) – Part 1: Media presentation description and segment formats.

[CMAF] ISO/IEC 23000-19: 2020, Common media application format (CMAF) for segmented media, AMENDMENT 1: Additional CMAF HEVC media profiles.

[AVC] ITU-T Recommendation H.264: 2020 "Advanced video coding for generic audio-visual services" | ISO/IEC 14496-10: "Information technology – Coding of audio-visual objects – Part 10: Advanced Video Coding."

[ISOBMFF] ISO/IEC 14496-12: 2020, Information technology – Coding of audio-visual objects – Part 12: ISO base media file format.

[ISOBMFF-NAL] ISO/IEC 14496-15: 2019, Information technology – Coding of audio-visual objects – Part 15: Carriage of network abstraction layer (NAL) unit structured video in ISO base media file format.

[HEVC] ITU-T Recommendation H.265: 2021, "Advanced video coding for generic audio-visual services" | ISO/IEC 23008-2: " High Efficiency Coding and Media Delivery in Heterogeneous Environments – Part 2: High Efficiency Video Coding."

[VVC] ITU-T Rec. H.266 | ISO/IEC 23090-3:2020 – MPEG-I Part 3: Versatile Video Coding.

[CICP-Video] ISO/IEC 23091-2: 2019, “Information technology – MPEG systems technologies – Part 8: Coding-independent code points.”

[CTA 608-E] ANSI/CTA-608-E S-2019: Line 21 Data Services. Note that for backwards compatibility this standard is referred to as a CEA document even though the official name is now CTA.

[CTA 708-E] ANSI/CTA-708-E R-2018: Digital Television (DTV) Closed Captioning. Note that for backwards compatibility this standard is referred to as a CEA document even though the official name is now CTA.

[CTA 708.1-E] ANSI/CTA-708-1 R-2017: Digital Television (DTV) Closed Captioning: 3D Extensions. Note that for backwards compatibility this standard is referred to as a CEA document even though the official name is now CTA.

[ISO 639-3] ISO 639-3:2007, Codes for the representation of names of languages-Part3: Alpha-3 code for comprehensive coverage of languages – First Edition.

[RFC 7826] IETF RFC 7826, Real Time Streaming Protocol (RTSP) Version 2.0, December 2016.

[RFC 3339] IETF RFC 3339, Date and Time on the Internet: Timestamps, July 2002.

[RFC 8141] IETF RFC 8141, Uniform Resource Names (URNs), April 2017.

[RFC 6874] IETF RFC 6874, Representing IPV6 Zone Identifiers in Address Literals and Uniform Resource Identifiers, February 2013.

[RFC 4648] IETF RFC 4648, The Base16, Base32, and Base64 Data Encodings, October 2006.

[RFC 7405] IETF RFC 7405, Case-Sensitive String Support in ABNF, December 2014.

[RFC 6381] IETF RFC 6381, The ‘Codecs’ and ‘Profiles’ Parameters for ‘Bucket’ Media Types, August 2011.

[RFC 9110] IETF RFC 9110, HTTP Semantics, June 2022.

[RFC 9111] IETF RFC 9111, HTTP Caching, June 2022.

[RFC 9112] IETF RFC 9112, HTTP/1.1, June 2022.

[RFC 5646] IETF RFC 5646, Tags for Identifying Languages, September 2009.

[RFC 4647] IETF RFC 4647, Matching of Language Tags, September 2006.

[BCP 47] IETF BCP 47, Best Common Practice of Language Tags, IETF RFC 5646, IETF 4647.

[DASH-IF IOP] DASH-IF Implementation Guidelines: Interoperability Points; Version 4.3, 2018, http://dashif.org/wp-content/uploads/2015/10/DASH-IF-IOP-v4.3.pdf.

[XML] Extensible Markup Language (XML) 1.0 (Fifth Edition), W3C Recommendation 26 November 2008, available at http://www.w3.org/TR/REC-xml/.

[XLINK] XML Linking Language (XLink) Version 1.0, W3C Recommendation 27 June 2001, available at http://www.w3.org/TR/xlink/.

[XML Schema] XML Schema Part 2: Datatypes Second Edition, W3C Recommendation 28 October 2004, available at http://www.w3.org/TR/xmlschema-2/.

[ISOBMFF-TT] ISO/IEC 14496-30: 2018, Information technology- Coding of audio-visual objects- Part 30: Timed text and other visual overlays in ISO base media file format- Second Edition.

[IMSC 1] W3C TTML Profiles for Internet Media Subtitles and Captions 1.0 (IMSC1), 2020.

2.3. Other Published Materials

No normative references are applicable

[MP4RA] MP4RA – Official Registration Authority for the ISOBMFF family of standards, mp4ra.org.

3. Informative References

The following documents might provide valuable information to the reader but are not required when complying with this document.

3.1. SCTE References

[SCTE 243-1] SCTE 243-1 Next Generation Audio Carriage Constraints for Cable Systems: Part-1 _Common Transport Signaling.

[SCTE 243-2] SCTE 243-2 Next Generation Audio Carriage Constraints for Cable Systems: Part-2 _AC-4 Audio Carriage Constraints.

[SCTE 243-3] SCTE 243-3 Next Generation Audio Carriage Constraints for Cable Systems: Part-3 _MPEG-H Audio Carriage Constraints.

[SCTE 243-4] SCTE 243-4 Next Generation Audio Carriage Constraints for Cable Systems: Part-4 DTS-UHD Audio Carriage Constraints.

3.2. Standards from Other Organizations

[ATSC T/300] ATSC Technology Group Report: ATSC 3.0 Launch – DASH Timeline and IMSC1.

[ETSI TS 103 285] ETSI TS 103 285 V1.3.1 (2020-02): "MPEG-DASH Profile for Transport of ISO BMFF Based DVB Services over IP Based Networks."

[ETSI 103 190-1] ETSI TS 103 190-1 V1.3.1 (2018-02): Digital Audio Compression (AC-4) Standard; Part 2: Channel based coding.

[ETSI 103 190-2] ETSI TS 103 190-2 V1.2.1 (2018-02): Digital Audio Compression (AC-4) Standard; Part 2: Immersive and personalized audio.

[ETSI 103 491] ETSI TS 103 491 V1.1.1 (2017-04), DTS-UHD Audio Format; Delivery of Channels, Object and Ambisonic Sound Fields.

[MPEG-H] ISO/IEC: “Information technology – High efficiency coding and media delivery in heterogeneous environments – Part 3: 3D audio,” Doc. 23008-3:2022.

[MPEG-D] ISO/IEC: “MPEG audio technologies – Part 4: Dynamic Range Control,” Doc. 23003- 4:2020.

[NorDig] NorDig Unified Requirements for Integrated Receiver Decoders for use in cable, satellite, terrestrial and managed IPTV based networks, NorDig.

3.3. Other Published Materials

[HLS I-D] IETF RFC 8216bis, HTTP Live Streaming.

[DV-DASH] Dolby Vision: Streams within the MPEG-DASH format, https://professionalsupport.dolby.com/s/article/How-to-signal-Dolby-Vision-in-MPEG-DASH?language=en_US.

[DV-HLS] Dolby Vision: Streams within the HTTP Live Streaming format, https://professionalsupport.dolby.com/s/article/How-to-signal-Dolby-Vision-in-HLS?language=en_US.

4. Compliance Notation

shall	This word or the adjective "***required***" means that the item is an absolute requirement of this document.
shall not	This phrase means that the item is an absolute prohibition of this document.
forbidden	This word means the value specified ***shall*** never be used.
should	This word or the adjective "*recommended*" means that there *may* exist valid reasons in particular circumstances to ignore this item, but the full implications *should* be understood and the case carefully weighed before choosing a different course.
should not	This phrase means that there *may* exist valid reasons in particular circumstances when the listed behavior is acceptable or even useful, but the full implications *should* be understood and the case carefully weighed before implementing any behavior described with this label.
may	This word or the adjective "*optional*" indicate a course of action permissible within the limits of the document.
deprecated	Use is permissible for legacy purposes only. Deprecated features *may* be removed from future versions of this document. Implementations *should* avoid use of deprecated features.

5. Abbreviations and Definitions

5.1. Abbreviations

AAC	advanced audio coding
ABNF	Augmented Backus-Naur Form
AC-3	Audio Codec 3 or Advanced Codec 3 (also Dolby Digital)
AES-CBC	Advanced Encryption Standard cipher block chaining
ANSI	American National Standards Institute
ATS	Adaptive Transport Stream
ATSC	Advanced Television Systems Committee
AVC	advanced video coding
BMFF	base media file format
CC	closed captioning
CEA	Consumer Electronics Association, former name of the Consumer Technology Association
CMAF	Common Media Application Format
CPB	coded picture buffer
CTA	Consumer Technology Association
DASH	[MPEG] dynamic adaptive streaming over HTTP
DRM	Digital Rights Management
DTS	trademark for DTS, Inc. audio (originally Digital Theater Systems, Inc.)
DTV	digital television
DVB	Digital Video Broadcasting [Project]
DVS	[SCTE] Digital Video Subcommittee
e.g.	for example (*exempli gratia*)
ETSI	European Telecommunications Standards Institute

FF	file format
HD	high definition
HDR	high dynamic range
HEVC	high efficiency video coding
HI	hearing impaired
HLS	HTTP live streaming
HRD	hypothetical reference decoder
HTTP	hypertext transfer protocol
i.e.	that is (*id est*)
IEC	International Electrotechnical Commission
IP	Internet protocol
ISO	International Organization for Standardization
ISOBMFF	ISO Base Media File Format
LCEVC	low complexity enhancement video coding
MBT	minimum buffer time
MPD	media presentation description
MPEG	Moving Picture Experts Group
MPEG-2 TS	MPEG-2 transport stream
NAL	network abstraction layer
PMT	program map table
SAP	stream access point(s)
SCTE	Society of Cable Telecommunications Engineers
SD	standard definition
SEI	supplemental enhancement information
TS	MPEG-2 transport stream
UHD	ultra-high definition
UPID	unique program identifier
URI	uniform resource identifier
URL	Uniform resource locator
URN	universal resource name
UTC	coordination universal time
VI	visually impaired (aka audio description or AD)
XLink	external link
XML	extensible markup language

5.2. Definitions

Definitions of terms used in this document are provided in this section. Defined terms that have specific meanings are capitalized. When the capitalized term is used in this document, the term has the specific meaning as defined in this section.

EPOCH Time	Referenceable Point in Time- 00:00:00 Thu 01 Jan 1970 UTC.
SCTE DASH Event	A DASH Event containing the payload of an SCTE message. It could be a converted SCTE Message directly from a linear stream or originated in the adaptive stream l.

5.3. Notation

This document uses notation similar to the one of [DASH].

Extensible markup language (XML) elements are written in bold face, e.g. **`Element1`**.

Child XML elements are separated from parent elements by a dot ('.'), e.g. **`Element2.Element1.`**

XML attributes are prefixed by an at-sign ('@'), e.g. `@attribute`. Attributes of an element are separated from the name of the containing element by at-sign, e.g. **`Element`**`@attribute.`

ISOBMFF boxes are written as box names enclosed in backquote ('`') signs, e.g. `` `box0` ``

Fields in ISOBMFF boxes are separated from box names by a dot ('.'), e.g. `` `box0`.field0 ``

In cases where an element has the same name as a concept it describes, when the name is written in bold face, it refers to the syntactic element. For example, **`Representation`** refers to an XML element named "Representation", while "representation" refers to the concept of a representation as defined in [DASH].

XML elements and attributes defined in SCTE 214 are prefixed with `scte214:`

URLs that do not fit into a single line are escaped with `\` character

6. SCTE 214 Notational Conventions

6.1. Normative XML schema

Descriptions of elements and attributes are normative and, when combined with the normative XML schema document (provided separately), comprise the full normative schema specification. Unless otherwise specified, the normative text and values assigned to elements or attributes in this specification ***shall*** be constrained by the bit stream equivalent field.

Non-normative schema illustrations and instance examples are included herein for informational purposes only. Any real or implied usage, semantics, or structure indicated by the schema illustrations and examples ***shall not*** be considered part of the specification.

No XML documents representing the structures defined in the schema are considered conformant unless they are valid according to the schema document. Additionally, other SCTE 214 standard normative parts *may* impose additional rules or restrictions that ***shall*** be adhered to in order for XML documents to be considered conformant to those parts.

In the case where this document and the normative schema document (i.e., the separately provided XML ‘xsd’ file) conflict, this document ***shall*** take precedence over the XML schema document.

The inclusion of a normative XML schema document does not require or imply the specific use of the schema nor a requirement that an XML document be validated.

If the SCTE 214 schema is used in combination with other schemas, it is recommended to utilize the namespace prefix of “scte214”. For example, `scte214:ContentIdentifier` to reference an SCTE 214 ContentIdentifier Type.

6.1.1. XML namespace and version

The following XML namespaces and versions are defined in the normative XML schema for this specification:

Prefix	Namespace	Version
scte214	http://www.scte.org/schemas/214	YYYYMMDD

6.2. Unknown/Unrecognized/Unsupported XML elements and attributes

Generally, unknown, unrecognized or unsupported XML elements and attributes contained within SCTE 214 elements *should* be ignored during XML document processing. Specifically, these are elements or attributes which the implementation does not understand or expect. XML parsers that encounter elements or attributes which are prohibited by a namespace *should* include exception handling.

6.3. Element order

Element order is constrained by the schemas and ***shall*** be preserved throughout processing of the XML document. In particular, the order of elements affects the end result of the processing. Consequently, an implementation failing to preserve the order *may* cause incorrect processing results. Subsequently, the process of producing an abstract XML Information Set (InfoSet) from a concrete XML document, e.g., by parsing it, ***shall*** always result in the same abstract InfoSet, with the same element order per XML InfoSet (see [XML InfoSet] for additional information). Any intermediary processing *may* enhance the XML document but it ***shall not*** alter the abstract InfoSet element order (i.e., the XML elements comprising the document ***shall*** stay in document order).

7. MPD Restrictions

7.1. Restrictions on MPD elements

1. **`MPD`**`@id` ***shall*** be present. Consecutive updates to the `MPD` ***shall*** have the same `MPD@id`.

2. **`MPD`**`@minBufferTime` ***shall*** be present. Its value *should* be equal or larger than maximum segment (live profile) or subsegment (on demand profile) duration.

3. If the **`MPD`**`@type` is `"dynamic"`:

   a. **`MPD`**`@minimumUpdatePeriod` ***shall*** be present.

   b. **`MPD`**`@maxSegmentDuration` ***shall*** be present.

Note: It is unsafe to base player buffer allocation on the attributes above whenever XLink is used, or MPD type is `"dynamic"`. See sec. 9.2.3 for more details.

   c. **`MPD`**`@availabilityStartTime` ***shall*** be present and *should* satisfy the conditions below:

      i. **`MPD`**`@availabilityStartTime` *should* not change across MPD updates

      ii. **`MPD`**`@availabilityStartTime` *should* be set to a fixed time in the past (e.g. the 00:00.00 GMT January 1, 1970, a.k.a. the Epoch time)

d. **MPD**@suggestedPresentationDelay *should* be present.

e. If it is expected that at some point in the future a media segment *may* become unavailable, then the @timeShiftBufferDepth attribute ***shall*** be present, or @timeShiftBufferDepth ***shall*** be present as a part of segment information.

f. **MPD.Location** element *should* be present.

g. **MPD.PatchLocation** *may* be present, and **MPD.Location** ***shall*** be present in this case. The patch URL *should* contain the value of **MPD**@publishTime of the original MPD (verbatim) as a query parameter named publishTime.

7.2. Restrictions on Period elements

1. The **Subset** element ***shall not*** be present.

2. The **Period.SegmentList** element ***shall not*** be present.

3. At least one audio and one video **AdaptationSet** element ***shall*** contain a **Role** element with @schemeIdUri="urn:mpeg:dash:role:2011" and @value="main" Perceptually equivalent audio adaptation sets differing by non-linguistic characteristics such as codecs or number of channels *should* be prioritized for the players using the @selectionPriority attribute across the equivalent adaptation sets with a higher value indicating higher preference. The **Role** descriptor ***shall*** be used only to express linguistic differences.

4. If a **Period** element represents a part of a multi-period asset, this **Period** *should* contain an **AssetIdentifier** element.

5. **Period**@start ***shall*** be present.

6. **Period**@id ***shall*** be present.

7.3. Restrictions on Adaptation Set elements

1. Every adaptation set ***shall*** use consistent addressing. Exactly one of the following restrictions ***shall*** be met:

   a. **AdaptationSet.SegmentTemplate** element is present, and no **Representation.SegmentTemplate** elements are present.

   b. Every **Representation** element within this Adaptation Set consists of a single segment and its URL is stated in the **Representation.BaseURL** element. The MPEG DASH ISOBMFF On Demand profile satisfies this requirement.

2. All Representations within an Adaptation Set ***shall*** use the same codecs, but not necessarily the same profiles and levels. Therefore, exactly one of the following restrictions ***shall*** be met:

   a. The **AdaptationSet**@codecs attribute is present and signals the maximum profile, level, and – when applicable tier of any representation contained in the adaptation set, and **Representation** elements within this **AdaptationSet** *may* contain the full @codecs

attribute including profile, level, or and – when applicable tier pertaining to that representation if these differ from the one signaled in **AdaptationSet**@codecs.

Note: As an example, no adaptation set is permitted to contain both `avc1` (AVC) and `hev1` (HEVC) video, however avc1.64Y01F (Progressive High@L3.1) and avc1.64Y028 (Progressive High@L4.0) can be in the same AVC adaptation set.

b. **AdaptationSet**@codecs is present as code point (4cc) only, and **Representation** elements within this **AdaptationSet** contain the full @codecs attribute including profile, level, and – when applicable – tier pertaining to that representation.

3. **AdaptationSet**@segmentAlignment attribute ***shall*** be present and have a value of `true`.

Note: Older versions of this specification allowed value of "1", which has been *deprecated* in the latest editions of [DASH].

4. **AdaptationSet**@startWithSAP ***shall*** be present and its value ***shall*** be 1 or 2.

5. If indexing (`sidx`) is used, then

   a. @subsegmentAlignment ***shall*** be present and have value of `true`.

   b. **AdaptationSet**@subsegmentStartsWithSAP ***shall*** be present and ***shall*** have a value of `1` or `2`. Values larger than 2 are allowed if **AdaptationSet**@segmentAlignment is also present.

   Note: The above *may* happen in cases such as low-latency deployment relying on CMAF chunks (subsegments).

6. For any adaptation set that contains video the following conditions ***shall*** apply:

   a. @maxWidth (or @width if all Representations have the same width) ***shall*** be present.

   b. @maxHeight (or @height if all Representations have the same height) ***shall*** be present.

   c. @maxFrameRate ***shall*** be present and *should* be an integer multiple of each @frameRate in this adaptation set. If all representations have the same frame rate, @frameRate rather than @maxFrameRate *should* be present.

   d. @scanType ***shall*** be present only if one or more of the video frames are coded as interlaced. In this case @scanType ***shall*** have value of "interlaced" .

   Note: The above implies that if an adaptation set is interlaced, all representations in it are interlaced. There is no need to signal progressive content – this is the default value of the @scanType attribute when not present.

   e. @sar ***shall*** be present and, consequently, all representations within this adaptation set ***shall*** have the same aspect ratio.

f. `@lang` attribute ***shall*** be used with video sign language media components to indicate which type of sign language is used. These code points are defined in [BCP-47].

g. If a representation contains at least one interlaced picture, this representation is considered interlaced. Interlaced and non-interlaced representations ***shall not*** be mixed in the same adaptation set.

Note: Alignment between interlaced and non-interlaced adaptation sets can be expressed by using adaptation set switching

h. Colorimetric properties of video representations in an adaptation set ***shall*** be the same. When these properties are known (e.g. can be derived from the VUI as defined in AVC [AVC], HEVC [HEVC], and VVC [VVC]), they *should* be signaled as defined in sec 11.1.5.

7. Supplementary video media components (e.g. video sign language) ***shall*** be in separate adaptation sets. They ***shall*** have the **`Role`** descriptor with value other than "main".

8. For any adaptation set containing audio, the following ***shall*** be true (and thus ***shall*** apply to all representations):

i. `@lang` ***shall*** be present. Language described through a language tag ***shall*** use the IETF BCP 47 format [RFC 9110] IETF RFC 9110, HTTP Semantics, June 2022.

[RFC 9111] IETF RFC 9111, HTTP Caching, June 2022.

[RFC 9112] IETF RFC 9112, HTTP/1.1, June 2022.

ii. [RFC 5646][BCP 47]. Subtags except the primary language subtag, e.g. region subtags such as "es-US", *should not* be used unless essential for linguistic disambiguation (e.g. "es-US" vs "es-CL") .

Note 1: Per IETF BCP 47, 2-character codes are to be used whenever possible, i. e. 3-character codes are not to be used when there is an equivalent 2-character code.

Note 2: the above signaling complies with both BCP-47 and `xs:lang` syntax and restricts the number of possible options for encoding a single language.

Note 3: BCP-47 extensions *may* be needed in cases such as regional dialects.

iii. When language can change and timing of such a change is unknown, the `` `mul` `` language code ***shall*** be used.

iv. When multiple audio services with unknown languages exist for a single video service, codes in the range `qaa` - `qtz` may be used.

v. When no language applies, then `` `zxx` `` language code *should* be used.

vi. When the language has no code, then `` `mis` `` language code *should* be used.

b. `@codecs` ***shall*** be present and contain sub-parameters as defined in RFC 6381.

Note: implies that only the option described in 2.b above is acceptable for audio adaptation sets.

c. `@audioSamplingRate`

d. **`AudioChannelConfiguration`** ***shall*** be present and ***shall*** be at **`AdaptationSet`** level. This implies that all audio representations within an adaptation set shall have same channel configuration.

e. There ***shall*** be same number of audio and video segments.

f. Audio and video segment duration ***shall*** be very close and the earliest presentation time of any i^{th} audio and i^{th} video segment *should* be within 70ms from each other.

Note: when **`SegmentTimeline`** and `$Time$`-based addressing are used for video, the above makes it possible to use **`SegmentTemplate`** with `$Number$`-based addressing and **`SegmentTimeline`** for audio. This *may* significantly reduce the overall MPD size. In order to keep timing tractable, it is recommended to use the **`SegmentTemplate`**`@tolerance` attribute and set it to a very low percentage of the segment duration (e.g., a sample duration). The above attribute was introduced in ISO/IEC 23009-1 5th Ed. AMD2.

9. If media segments contain CTA 608/708 closed captioning carried in video elementary stream (as defined in [SCTE 128-1] and [SCTE 215-1]), this ***shall*** be reflected in the MPD using **`AdaptationSet.Accessibility`**, as described in 8.2.

10. For any adaptation set carrying text (such as closed captioning or subtitles) the following ***shall*** hold:

    a. IMSC1 text profile of TTML ***shall*** be used;

    b. The above provide a continuous timeline and avoid the creation of sparse tracks by using empty segments to create a continuous set of segments. Please refer to [ATSC T/300] for recommended practices.

    c. Text adaptation sets *should* be aligned to video adaptation sets, and the earliest presentation time of any i^{th} text and i^{th} video segment *should* be within 100ms from each other.

    d. `@lang` ***shall*** be present. Language described through a language tag ***shall*** use the IETF BCP 47 format [BCP 47]. Other subtags except the primary language subtag, e.g. region subtags such as used in "es-US", should not be used unless essential in language disambiguation. If the information expressed in the code is insufficient to determine the linguistic content, BCP-47 extensions ***shall*** be used for the purpose.

        Note 1: Per IETF BCP 47, 2-character codes are to be used whenever possible, i. e. 3-character codes are not to be used when there is an equivalent 2-character code.

        Note 2: the above signaling complies with both BCP-47 and `xs:lang` syntax and restricts the number of possible options for encoding a single language

        Note 3: BCP-47 extensions *may* be needed in cases such as regional dialects or different scripts used for the same language.

11. If segments comply with one or more CMAF profiles and structural brands defined in [CMAF] or in CTA-5001, these ***shall*** be signaled either in the **`AdaptationSet`**`@containerProfiles recommended)` or @segmentProfiles attribute using the ISOBMFF brands defined in the above standards.

12. Segments *should* be referenced from one and only one **`SegmentTemplate`** element.

13. Timescale values for ISOBMFF representations ***shall be*** selected such that any sample duration would be an integer. **Note** that the MPEG-2 Systems 90KHz timescale does not satisfy this requirement in cases of video frame rates such as 24000/1001 or 60000/1001 fps. One of the two practices below are recommended:

    a. Use different timescales in different adaptation sets. In particular, use frame rate (video) or sampling rate (audio) as timescale; For example, timescale of 24 can be used for 24fps content, while value of 60000 can be used for 60000/1001 (i.e., each sample duration is 1001).

    b. Use a common timescale across adaptation sets, allowing for integer sample duration for all adaptation sets. In particular, 240KHz timescale satisfies these requirements in most cases. The resolution of the timescale shall be such that the timestamp value will not exceed 25.

14. **`AdaptationSet`** attributes for supplemental information such as codecs ***shall*** be used for enhanced media experiences such as [SCTE 215-1]. @supplementalCodecs and @supplementalProfiles attributes are not essential attributes of the content but can be used to enhance the media experience upon playout but are not essential to the playout of the content.

    Note: By signalling enhanced viewing properties in the form of **`AdaptationSet`** attributes can be used to avoid separate **`AdaptationSet`** elements pointing to the same content segments.

    Supplemental media properties defined for these purposes are:

    a. **`AdaptationSet`**`@scte214:supplementalProfiles`
    Values for this list are ISOBMFF optional brands (registered in mp4ra.org [MP4RA]) with whitespace separated multiple values being listed in no particular order.

    b. **`AdaptationSet`**`@scte214:supplementalCodecs`

    Values for this list of codec strings for “enhancement” codecs with whitespace-separated multiple codec strings being listed in no particular order.

7.4. Restrictions on ContentComponent elements

`ContentComponent` element ***shall not*** be used with segment formats other than MPEG-2 TS, i.e., unless the **`AdaptationSet`**`@mimeType` equals video/mp2t, audio/mp2t, or application/mp2t.

For MPEG-2 TS segments, the below conditions apply:

1. **`AdaptationSet`** elements ***shall*** contain a single **`ContentComponent`** element per each media component in a multiplexed representation.

2. If more than one audio content component is present, each one of them ***shall*** be signaled using a separate **`ContentComponent`** element. `@lang` attribute ***shall*** be present for each audio component.

3. **ContentComponent**@contentType attribute ***shall*** be present in any **ContentComponent** element.

7.5. Restrictions on Representation elements

1. The following attributes and elements ***shall not*** appear at **Representation** level within an adaptation set containing audio:

    a. **AudioChannelConfiguration**;

    b. @audioSamplingRate;

    c. @lang;

    d. @codecs

    Note: the above implies that audio representations within an adaptation set always have identical codec, sampling rate and number of channels.

2. For any **Representation** element within an adaptation set containing video the following attributes ***shall*** be present:

    a. @width, if and only if **AdaptationSet**@width is not present in this adaptation set

    b. @height, if and only if **AdaptationSet**@height is not present in this adaptation set

    c. @frameRate, if and only if **AdaptationSet**@frameRate is not present in this adaptation set

    d. @codecs, which ***shall*** contain complete sub-parameter string as defined in [ISOBMFF-NAL] Annex E. **Note** that this means that 2.a in 6.3 applies to video adaptation sets.

3. **Representation**@id value ***shall*** be unique within the scope of the **Period** to which it belongs.

4. **Representation**@bandwidth value ***shall*** be unique within its parent **AdaptationSet** element.

5. **Representation.ContentProtection** element ***shall not*** be used.

6. **Representation.SegmentList** element ***shall not*** be used.

7. **Representation**@scte214:maxSegmentRate ***should*** be used, as defined in sec. 10.3.2.

8. In case of variable bitrate content, **Representation.ExtendedBandwidth** element *should* be used to express average bandwidth.

    a. **ExtendedBandwidth**@vbr ***shall*** be true.

    b. For assets conforming to the live profile, **ModelPair**@bufferTime *may* be infinite if the model describes target bitrate as set in the encoder and not all segments are available. It is

recommended to use the value `P999D` for the purpose, for compatibility purposes, as infinity cannot be properly expressed with `xsd:duration`. If all segments are available, **`ModelPair`**`@bufferTime` ***shall*** be set to the total duration of the period.

Note: For assets conforming to the VOD profile, all bitrate information can be derived from the `sidx` box, so extended signaling is redundant. With that said, there *may* be operational circumstances where its use is needed.

7.6. Restrictions on use of XLink

The use of the XLink (only the subset defined in [DASH]) is supported in SCTE profiles with the following restrictions:

1. The `@xlink:href` attribute *may* appear only in **`Period`** elements;

2. If the **`Period`**`@xlink:href` attribute is present, the value of **`Period`**`@xlink:actuate` ***shall*** be `'onLoad'`.

7.7. Use of events

7.7.1. Inband events

1. Event message boxes with same event scheme ***shall*** be aligned, per definition of event alignment in [DASH].

2. Version 0 of the EventMessageBox (`emsg`) ***shall not*** be used.

7.7.1.1. Declaring events

1. **`InbandEventStream`** element ***shall not*** be present either at **`Representation`** or at **`SubRepresentation`** level.

2. If inband events are used, their presence ***shall*** be signaled in **`AdaptationSet.InbandEventStream`** element. A client cannot be expected to process undeclared events, though this specification does not disallow processing them.

7.7.1.2. DASH events

MPD Validity Expiration events ***shall*** only have the value of 1 (i.e., not contain MPD or MPD patch).

7.7.2. MPD Events

7.7.2.1. Carriage of SCTE 35 as MPD events

SCTE 35 cue messages can be encapsulated into a DASH **`Event`** element hereby called in this document as ‘SCTE DASH Event’**.** SCTE DASH Events can be published in an **`MPD`** at the Period level when SCTE 35 messages in the stream are ingested and processed by systems responsible for MPD creation. SCTE DASH Events are contained in one or more **`EventStream`** elements which are hereby called in this document as ‘SCTE DASH Event Stream’.

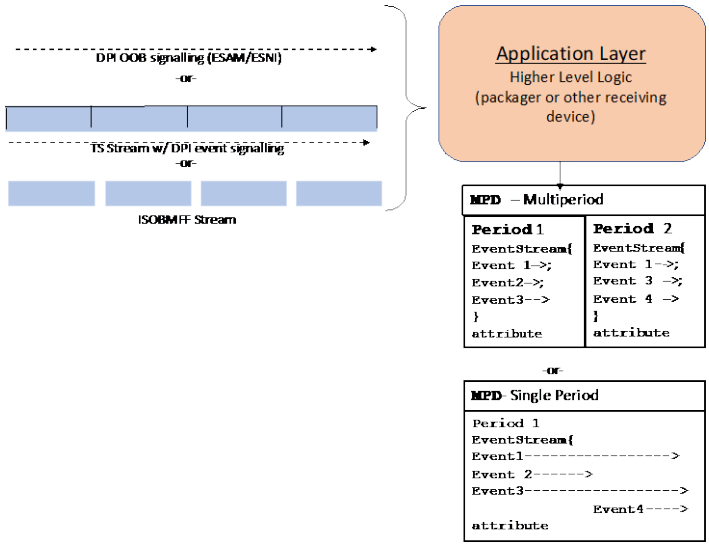

Figure 1 - Handling of SCTE 35 information

SCTE DASH Events ***shall*** operate in an "on-receive" dispatch mode. Events will be dispatched at or before the event time to the Application [as defined in DASH 5th edition and DASH-IF 5.0]. This implies that the message will be passed to the application upon arrival with devices subscribing to the DASH Events.

SCTE DASH Event elements contained in an SCTE DASH Event Stream element with `@schemeIdUri="urn:scte:scte35:2013:xml"` ***shall*** contain an XML representation of an SCTE 35 cue message as either a **`Signal.SpliceInfoSection`** element or a **`Signal.Binary`** element as defined in SCTE 35.

SCTE DASH Event elements contained in an SCTE DASH Event Stream element with `@schemeIdUri ="urn:scte:scte35:2014:xml+bin"` ***shall*** contain an XML representation of an SCTE 35 cue message a **`Signal.Binary`** element as defined in SCTE 35.

For both `"urn:scte:scte35:2013:xml"` and `="urn:scte:scte35:2014:xml+bin"` schemes, SCTE DASH Event Stream elements ***shall*** satisfy the following constraints,

1. If more than one SCTE DASH Event Stream element with identical value of **`EventStream`**`@schemeIdUri` is present, then the **`EventStream`**`@value` attribute ***shall*** be present and *should* be set to the decimal PID value of the elementary stream or to a URI.

2. **`EventStream`**`@timescale` is an integer value for a common timescale that is granular enough to accommodate or allow at least all frame-accurate access and possibly frame-accurate access to audio. To achieve the former video timescale can be used, to achieve the latter a common time scale is needed. A timescale of 240000 is known to work with typical video and audio configurations used in US cable systems, but this standard does not require its use.

3. **`EventStream`**`@presentationTimeOffset` *should* be present. When **`EventStream`**`@presentationTimeOffset` is present and its value *should be* set such that **`Event`**`@presentationTime` of an event starting before the start of the current period would still be a non-negative number.

Note 1: the above allows expressing events happening in the past without keeping periods for this purpose. As a result, the receiver *may* receive the same SCTE DASH Event element twice, in different periods, and is expected to handle this.

Note 2: It is possible to do "epoch locking" and start the event timing from the POSIX Epoch time, January 1, 1970 00:00.00Z. This implies that **`EventStream`**`@presentationTimeOffset` set to the number of ticks since the Epoch till the start of the period, and **`EventStream`**`@presentationTime` is set to the number of ticks from the Epoch till the event start as calculated from the SCTE 35 cue message. Note that the condition **`EventStream`**`@presentationTimeOffset > ` **`Event`**`@presentationTime` is legal and indicates an event starting prior to the start of the current period.

4. SCTE DASH Event are considered essential, per definition of event essentiality in ISO/IEC 23009-1 [DASH].
   Note: SCTE DASH Event***s*** ***shall*** be received immediately and sent to the application.

For both `"urn:scte:scte35:2013:xml"` and `="urn:scte:scte35:2014:xml+bin"` schemes, SCTE DASH Event elements ***shall*** satisfy the following constraints,

1. Each SCTE DASH Event carrying an SCTE 35 Event ***shall*** contain only one SCTE 35 SpliceInfoSection. The SCTE DASH Event ***shall*** include a reference to either the Splice Insert splice_event_id or Segmentation Descriptor segmenation_event_id that is applicable to the SCTE DASH Event. See SCTE 35 on the specific constraints on which descriptors are applicable to each splice command.

2. **`Event`**`@duration` *should* be the expected duration of the SCTE cue message. If the event is "closed" by a different event that event *should* have a duration of zero and ***shall not*** be indefinite.

   Note 1: Duration of zero implies that the application subscribed to SCTE 35 messages will get it immediately on its reception. If there is an expectation for a DASH client joining after the event start (e.g. during an ad break) to still reliably receive the event **`Event`**`@duration` *should* be sufficient for any such clients to join during its duration. For example, a duration can be set to an expected duration of the ad break.

   Note 2: An SCTE 35 cue message can be terminated by receiving a paired SCTE 35 event end message for the event, going beyond the assigned duration in an SCTE 35 event start message. Event extending beyond the period boundary will be ignored by clients starting from the next period unless copied into that period. Such copied event would have `presentationTime` smaller than **`EventStream`**`@presentationTimeOffset.`

3. The **`Event`**`@messageData` attribute ***shall not*** be used.

4. SCTE DASH Event elements *should* be retained as long as one or more media segments containing media with presentation times during the event are available.

5. There *should* not be more than one SCTE DASH Event in the SCTE DASH Event Stream with the same **`Event`**`@id` having identical values for the **`Event`**`@presentationTime` attribute. Note that subsequent events with same **`Event`**`@id` value *may* be dropped by the DASH client.

7.7.3. Carriage of SCTE 35 as inband event messages

1. SCTE 35 events ***shall*** be used with emsg v1 and use scheme `"urn:scte:scte35:2013:bin".`

2. For events with the above scheme, entire SCTE 35 `splice_info_section` starting at the `table_id` and ending with the `CRC_32` ***shall*** be carried in `` `emsg`.message_data[]. ``

3. `` `emsg`.presentation_time `` ***shall*** equal the time distance between the splice time and *PeriodStart* `as` adjusted by `@presentationTimeOffset` in the time units specified in `` `emsg`.timescale ``. These units ***shall*** be same as the timescale of the segment carrying the `` `emsg` `` box.

   Note: epoch locking allows expressing events in the past. Since both `` `emsg` `` and the media segments carrying it share the value of `@presentationTimeOffset`, epoch locking will be needed for the media segments as well.

4. `` `emsg`.event_duration `` ***shall*** be a translation of `segmentation_duration` or `break_duration` into time units specified in `` `emsg`.timescale ``. If not known, the value ***shall*** be 0xFFFFFFFF (not known).

5. `` `emsg`.value `` ***shall*** be the value of either the SCTE 35 PID, or a URI, or may be left empty.

   Note: `` `emsg`.value `` *may* be used by the application to distinguish between cue messages from different sources.

6. `` `emsg`.id `` ***shall*** be a unique integer within the scope of the Period (within the scope of events defined in the **`InbandEventStream`** element with same value of **`InbandEventStream`**`@value`). This can be achieved using approaches as random 64-bit numbers, truncated hashes or CRC of the descriptor, or/and lower bits of the current time. Event messages with the same `` `emsg`.id `` values are regarded as repeated messages carrying no new information. The DASH client will handle the first occurrence of a specific event message identified by an `` `emsg`.id ``. Other occurrences of the event message with the same `` `emsg`.id `` can be discarded by the DASH client since it contains no new information.

   Note: Please consult with ISO/IEC 23009-1 5th ed. Annex A for a detailed description of the DASH event handling model

7. Presence of inband SCTE 35 events in media segments ***shall*** always be signaled using **`AdaptationSet.InbandEventStream`** element with `@schemeIdUri` value of `"urn:scte:scte35:2013:bin".` SCTE 35 events are essential as defined in SCTE 214-1 6.8.3

   Note: As `` `emsg` `` boxes are aligned, all `` `emsg` `` boxes will be read if media from an adaptation set is played out irrespective of representation selections.

Figure 2 below shows the content of an `emsg` box at the beginning of a segment with earliest presentation time *T*. There is a 6-sec warning of an upcoming splice – delta to splice time is indicated as 6 seconds – and duration is given as 1 minute. This means that an ad will start playing at time *T + 6* till *T + 66*.

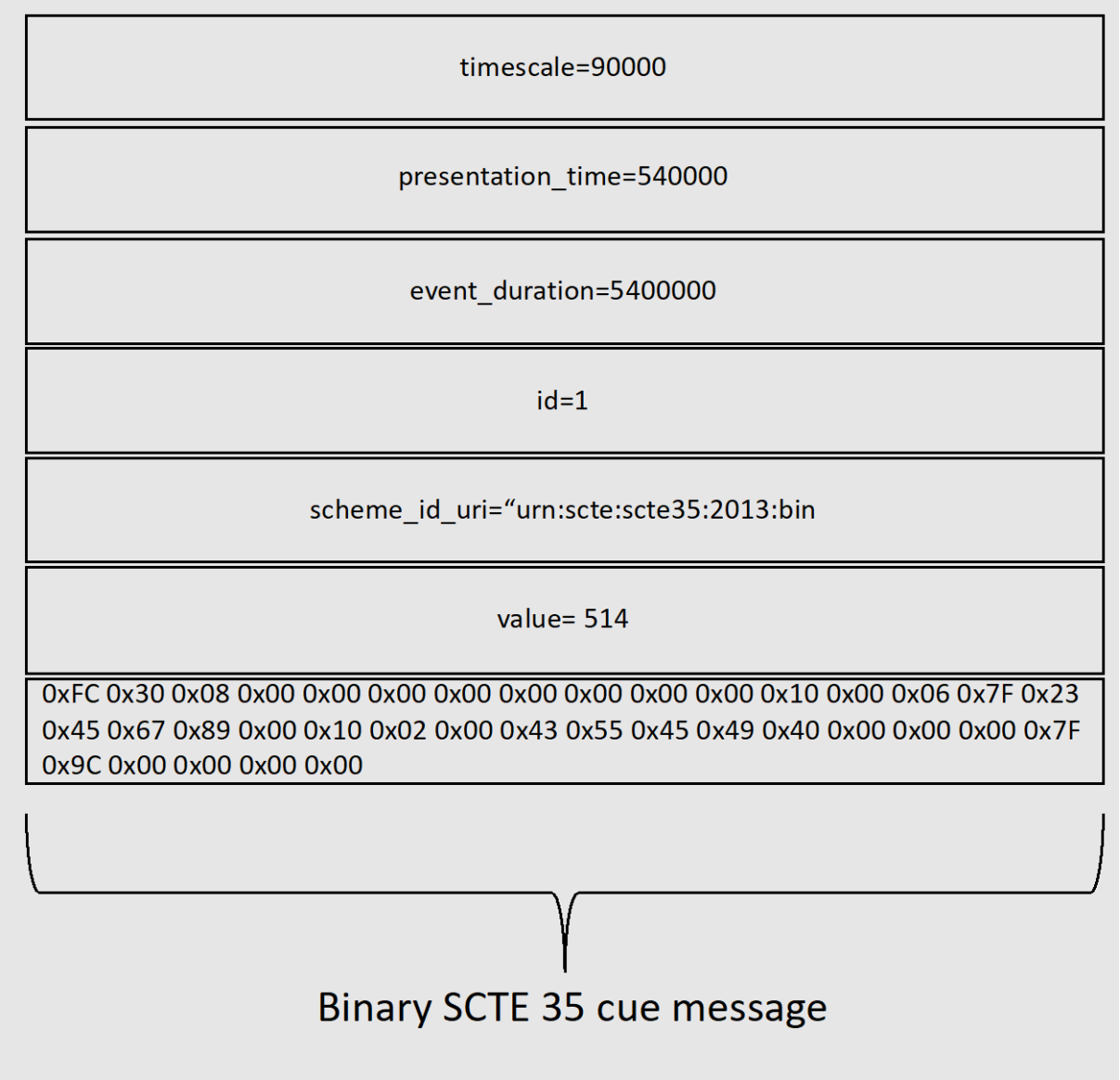

Figure 2 - Inband carriage of SCTE 35 cue message

7.8. MPD Updates

MPD updates *may* only extend the timeline, unless this is an MPD reset. This means that information provided in a previous version of the MPD *should not* be invalidated in an updated MPD. Hence the expected changes are typically addition or removal of **`Period`** elements, addition or removal of **`Event`** elements, or modification of **`SegmentTemplate`** (e.g., of **`S`** elements in **`SegmentTimeline)`**.

In live scenarios MPD updates can add new MPD events but ***shall not*** remove existing MPD events in order to provide system consistency. Consequently, a cancellation of a previous event *should* be done via MPD update adding a new event. With that said, preserving events is not the reason to keep an old period when its segments are no longer available – relevant (e.g. "open") SCTE 35 events *should* be copied to the current **`Period`**, and **`Period`** element removed via an MPD update. A "closed" event should not be copied into the current **`Period`** if no media segments under the event time are available.

Note: MPD Patches as defined in [DASH] 5th edition are typically a more efficient mechanism then frequent MPD updates. Use of patches reduces the traffic and parsing overhead of MPD updates. The restrictions above apply to changes introduced either by a full MPD update or by a patch.

7.9. MPD handling of missing segments

`FailoverContent` element *should* be used for missing segments at the time of MPD creation or updates. If missing segments are later found, then MPD update can remove the indicated segment from **`FailoverContent`** element at the next MPD update or through MPD patches.

8. Signaling service metadata

8.1. Associated services

8.1.1. General

In many cases an audio component is not intended for a general presentation, but for a more specialized purpose (e.g., audio description for the visually impaired). Moreover, in some cases (known as "receiver mix"), two audio elementary streams need to be combined for the same service.

This section defines signaling for such services. If signaling is present both in the media segments and in MPD, the two ***shall not*** contradict each other.

This section uses the **`Role`** descriptor for two different purposes. In 8.1.3 **`Role`** is used to express the purpose of the audio component, while in 8.1.48.1.4 the value of the **`Role`** descriptor indicates whether an audio component represents a full or partial service.

8.1.2. Media language defaults

In some cases, it is unclear which language *should* the player select by in absence of explicit user selection. One or more **`SupplementalProperty`** elements defined below can be used to explicitly signal this.

The value of the `@schemeIdURI` attribute ***shall*** be `urn:scte:dash:default-language<#fragment>` where the optional fragment argument can take the values "video", "audio", "text", and "commentary". If no fragment parameter is supplied, the language applies to all content components within the scope of the parent element.

The value of the `@value` attribute in these elements ***shall*** be a language as described in sec 7.3 above. Example:

```
<SupplementalProperty schemeIdUri="urn:scte:dash:default-language#audio" value="en"/>
```

```
 <SupplementalProperty schemeIdUri="urn:scte:dash:default-
language#text" value="en"/>
 <SupplementalProperty schemeIdUri="urn:scte:dash:default-
language#video" value="ase" />
<SupplementalProperty schemeIdUri="urn:scte:dash:default-
language#commentary" value="es"/>
```

These supplemental properties represent the default language of the media or content for audio, text, video signing language, commentary from the channel or content perspective. The client ***shall*** use the specified language in absence of explicit user preferences. In cases where program language may shift from program to program but all programs are in its native language, the channel language of "qaa" can be used to indicate the channel language is native but can vary in the channel. See [NorDig] see Table 12.6.

In applications at the MPD level, it can represent the default language of the channel and can be used to align the language choices for a program, ad, or promo that starts at the new period boundary.

In applications at the Period level, it can represent an override of the MPD level default language information if available. An example of this may be a forced English commercial on a Spanish language channel.

8.1.3. Roles

Personalized audio services ***shall*** be signaled using the **`Role`** descriptor with `@schemeIdUri ="urn:mpeg:dash:role:2011"`. The role descriptor can differentiate multiple audio streams associated with a video media adaptation set or video content component. This allows the client player to select the appropriate audio adaptation set or content component for the video component adding some personalization factor to the content experience. It can assist with creating a smooth content experience with channels or programs that have inserted ads to align the audio experience of the ad with the main channel or program.

Associated services, such as visually impaired (VI) (aka audio description) and hearing impaired (HI), ***shall*** be signaled using the **`Accessibility`** descriptor with `@schemeIdUri="urn:mpeg:dash:role:2011"` or **`Role`** descriptor with `@schemeIdUri="urn:scte:dash:associated-service:2015".`

Note: the earlier versions of this specification required **`Role`**, rather than **`Accessibility`** descriptor. The 2021 edition of this specification requires use of **`Accessibility`** descriptor in order to achieve better alignment with DASH-IF and DVB approaches

8.1.3.1. Derivation from MPEG-2 TS

The role values *should* be derived from MPEG-2 TS stream using the ISO_639_ language descriptor in the program map table (PMT)[1]. In absence of ISO_639_ language descriptor, Role / Accessibility can be derived from the audio elementary streams as discussed below.

[1] Note that the ISO_639_language descriptor could contain more than one pair of language/audio_type values.

The recommended process for deriving Role/Accessibility for audio elementary stream is as follows. Let service_type *ST* be an integer value instantiated as follows:

- For AC-3 and E-AC-3 elementary streams, *ST* takes the value of the `bsmod` field. The possible values for `bsmod` are defined in [ATSC A/52] Table 5.7
- For advanced audio coding (AAC) elementary streams, *ST* takes the value of `AAC_service_type`, as defined in [SCTE 193-2] Table 4.
- For DTS elementary streams, *ST* value is derived from `component_type` bit values `b3`, `b4` and `b5`, as follows: *ST* = `b5 << 2 + b4 << 1 + b3`. The values `b3`, `b4` and `b5` are defined in [SCTE 194-2] Table 6.

8.1.3.2. DASH signaling

The value of the **`Role`**`@value` attribute ***shall*** be derived from equivalent audio type value as described in table below.

Table 1 - Audio Services

Type	**`Role`**`@value`	**`Accessibility`**`@value`
Audio default (audio_type = 0x00 | service_type [bsmod or equivalent] = 000)	main[2]	N/A
Clean effects (audio_type = 0x01 | service_type [bsmod or equivalent] = 001)	SCTE: Music & Effects	N/A
Primary Audio (audio_type = 0x80)	main[3]	N/A
Native Audio (audio_type = 0x81)	absence of dub[4]	N/A
Emergency (audio_type = 0x82 | service_type [bsmod or equivalent] = 110)	emergency	N/A
Primary Commentary	main[5],	N/A

[2] The role of "main" can occur across multiple adaptationSets of the period. This can apply across several dimensions of the media such as audio channel order, or audio codec format. Selection between multiple adaptation sets with the same language and role should be done according to the value of `@selectionPriority`.

[3] The supplemental property for media language default for audio *should* be set at the **`MPD`** or **`Period`** Level

[4] In DASH, native audio is not signaled explicitly but can be determined be the absence of "**`role`** = dub" to describe the track. In cases where program language may shift from program to program but all programs are in its native language, the channel language of "qaa" can be used to indicate the channel language is native but can vary in the channel. See [NorDig] see Table 12.6.

[5] The supplemental property for media language default for commentary *should* be set at the **`MPD`** or **`Period`** Level

(audio_type = 0x83 | service_type [bsmod or equivalent] = 101)	commentary	
Alternate Commentary (audio_type = 0x84)	alternate, commentary	N/A
ST = 100 or 111	TBD	N/A

Table 2 - Accessibility Associated Services

Type	**Role**@value	**Accessibility**@value
Audio description (audio_type = 0x03 | service_type [bsmod or equivalent] = 010)	alternate	description
Clean audio (audio_type = 0x02 | service_type [bsmod or equivalent] = 011)	alternate	enhanced-audio-intelligibility
Closed Captions[6]	main	captions
Sign Language[7]	supplementary	sign

The expected practice in North America is that an audio adaptation set having `@contentType="audio"` and **`Role`**`@value = "main"` is equivalent to the audio service "Complete Main," which is defined for audio standards such as AAC and DTS. In North America, the "Complete Main" audio service is an audio component that contains a complete audio program (which typically includes dialog, music, silence, and effects).
For any two audio adaptation sets belonging to the same period element with the role of main, the values of `@lang` ***shall*** be identical.

The expected practice in North America is that audio adaptation sets having **`Role`**`@value = "commentary"` are equivalent to the audio service "commentary", which is defined for audio standards such as AAC and DTS.

DASH **`Role`** scheme `"urn:mpeg:dash:role:2011"` ***shall*** be used if there is more than one audio component a client can select (i.e., multiple audio services within a single multiplex or multiple audio adaptation sets).

In case of *ST=0* and multiple audio content components (as described above), **`Role`** descriptor with `@schemeIdUri="urn:mpeg:dash:role:2011"` and `@value="main"` ***shall*** be used.

In case of *ST > 1*, **`Role`** descriptor with `@schemeIdUri="urn:mpeg:dash:role:2011"` and `@value="alternate"` ***shall*** be used in case of full service, and `"supplementary"` ***shall*** be used otherwise ("receiver mix").

[6] Closed captioning is an accessibility component for a video or text track indicated by the caption service descriptor in the PSI. The equivalent audio_type value would be 0x00

[7] Sign language is identified through the @lang attribute (e.g. "ase" or "bfi"). The equivalent audio_type value would be 0x00.

8.1.4. Full and partial audio services

An audio service *may* be a full service suitable for presentation, or only a partial service which *should* be combined with another audio service before presentation ("receiver mix"). In case the partial and the full services are in different adaptation sets, it is necessary to signal such dependence in order to indicate to the client that two adaptation sets need to be downloaded prior to the presentation.

Note: There is no need to signal this for a multiplex containing both – inband signaling in this multiplex is sufficient.

Let *F* be a boolean value, which indicates whether a service is a full service ('true'), or the client will need to combine it with a different audio service ('false').
For AC-3 and E-AC-3 elementary streams, *F* is true if and only if the `full_svc` bit in the `AC-3_audio_stream_descriptor` is set to `1`.

For AAC, *F* is true if and only if `receiver_mix_rqd` is set to '0' (see [SCTE 193-2] Table 1).

For DTS, *F* is true if and only if `full_service_flag` bit in `component_type` field is set to `1` (see [SCTE 194-2] tables 6 and 7).

If neither signaled nor known by other means, *F* is assumed to be true.

In case *F* is false for an audio service in adaptation set A, and it needs to be combined with a different audio service in a different adaptation set B, this will be signaled in adaptation set A using an **`EssentialProperty`** descriptor with `@schemeIdUri` attribute value of `urn:mpeg:dash:audio-receiver-mix:2015` . The `@value` attribute ***shall*** the value of **`AdaptationSet`**`@id` of B.

Note 1: this signalling is defined in sec. 5.8.5.7 of [DASH] and was introduced in ISO/IEC 23009-1 3rd edtion.

Note 2: AC-3, E-AC-3 and AAC full service is signalled in PMT descriptors, hence when [ISOBMFF] segments are generated from an MPEG-2 TS source, such signalling is expected to be translated into signalling defined in this section by the entity performing the container format conversion.

8.2. Caption service metadata in video media components

8.2.1. Introduction

CTA-608 and CTA-708 caption services are carried embedded in the elementary streams. Carriage of CTA-608 and CTA-708 in supplemental enhancement information (SEI) messages is defined in [SCTE 128-1] and [SCTE 215-1]. This section describes MPD signaling of caption service metadata for and applies to content with both MPEG-2 TS and ISOBMFF segments.
Signaling is done using the **`Accessibility`** descriptors, one per each standard. The value string of each descriptor can be either list of languages or a complete map of services (or CC channels, in CTA-608 terminology).

Listing languages without service/channel information is strongly discouraged if more than one caption service is present. At any time language-channel (CTA-608) or language-service (CTA-708) is known at content generation time, it ***shall*** be used, as opposed to signaling mere presence or presence and language.

Note: Signaling described in this section is identical to [DASH-IF IOP].

8.2.2. Signaling CEA-708 caption service metadata

If CEA-708 closed caption service is carried in the video elementary stream, the relevant metadata per CEA-708 sec. 4.5 will be expressed using **`ContentComponent.Accessibility`** or, if the latter is not used, **`AdaptationSet.Accessibility`** with `@schemeIdURI` set to `urn:scte:dash:cc:cea-708:2015`.

The `@value` attribute ***shall*** contain the Caption Service Metadata as provided in CEA-708 section 4.5, as a semicolon-separated string of service descriptions. Each service description is either a single language code or a list of colon-separated name-value pairs

```
@value            = service *15 [";" service]
service           = language / ( service-number "=" param )
service-number    = (%d1 - %d63) ; decimal numbers 1 through 63
param             = language[",”easy-reader][",”aspect-ratio] [",”3d]
language          = "lang" ":" 3ALPHA; language code per ISO 639-3
easy-reader       = "er" ":" BIT ; default value 0
aspect-ratio      = "war" ":" BIT / "?"
                     ; default value is 1 (16:9),
                     ; value '0' indicates 4:3,
                     ; value '?' if unknown
3d                = "3D" ":" BIT
                     ; 1 if caption disparity data is present (CEA-708.1)
                     ; default value 0 (no 3d support).
```

Note: `ALPHA` and `BIT` are as defined by IETF RFC 5234, Appendix B.1.

Each of the `service` parameters (except for language) *may* be present or not present. Default values can be assumed where specified. If more than one language is present, the `service-number` ***shall*** be present. If `service-number` is known to the entity generating the MPD, it *should* be present even when a single language is signaled.

The CEA-708 information supplied in the **`Accessibility`** descriptor ***shall not*** contradict information supplied in the `caption_service_descriptor` in the PMT. See 0 below for derivation.

8.2.3. Signaling CTA-608 caption service metadata

If CTA-608 closed caption service is carried in the video elementary stream, language metadata will be expressed using **`AdaptationSet.Accessibility`** with `@schemeIdURI` set to `urn:scte:dash:cc:cea-608:2015`.

The `@value` attribute ***shall*** contain description of caption service(s) provided in the stream, as either a semicolon-separated list of languages or of colon-separated channel-language pairs. The `@value` syntax ***shall*** be as described in the Augmented Backnus-Naur Form (ABNF) below.

```
@value            = language / channel 0*3 [";" channel]
channel           = channel-number "=" language
```

```
channel-number     = "CC1" / "CC2" / "CC3" / "CC4"
language           = 3ALPHA ; language code per ISO 639-3
```

If language is known more than one language is present, the `channel-number` ***shall*** be present. If `channel-number` is known to the entity generating the MPD, it ***shall*** be present even when a single language is signaled.

8.2.4. Caption service examples

Table 3 - Caption Service Examples

```
<!-- Simple signaling of presence of CTA-608 closed caption service -->
<!-- NOTE: not signaling languages is a discouraged practice        -->
<Accessibility
    schemeIdUri="urn:scte:dash:cc:cea-608:2015"/>


<!-- Signaling of presence of CTA-608 closed caption service
<!-- in English and German, with channel assignments -->
<Accessibility
    schemeIdUri="urn:scte:dash:cc:cea608:2015" value="CC1=eng;CC3=deu"/>

<!-- Signaling of presence of CTA-708 closed caption service -->
<!-- in English and easy reader English                      -->
<Accessibility
    schemeIdUri="urn:scte:dash:cc:cea708:2015"
    value="1=lang:eng;2=lang:eng,war:1,er:1"/>
```

Note: For captioning, ABNF channel syntax is preferred and channel-less syntax is *deprecated*.

8.2.5. Derivation of caption service metadata from MPEG-2 TS

When MPD and media segments are generated from MPEG-2 transport stream, the PMT *may* contain the `caption_service_descriptor()` descriptor, as defined in Sec. 6.9.2 of ATSC A/65. If this descriptor is present, MPD signaling of caption service ***shall*** be generated using the procedure described below.
If there is a service for which `cc_data.digital_cc` bit is '0', then **`Accessibility`** with URI `urn:scte:dash:cea-608:2015` ***shall*** be used to signal it. If languages or channel-language association is known (from any source), it *should* be provided, using syntax from 8.2.3.

If there is at least one service with `cc_data.digital_cc` bit set to '1', then **`Accessibility`** with URI `urn:scte:dash:cea-708:2015` ***shall*** be used to signal it. For each such service syntax defined in **8.2.2** ***shall*** be used, and at least service number and language ***shall*** be provided.

Note 1: Descriptors for both CTA 608 and CEA 708 often appear in the same scope.

Note 2: PSI, and, consequently, caption service descriptors *may* change at splice points. In case of a splice we expect a new period to be started and the process above will be applied to the new period.

Note 3: Use of CTA 708.1 is not (as of 2015) reflected in `caption_service_descriptor()`

9. Signaling Asset Identification

9.1. General

AssetIdentifier elements are used in [DASH] to uniquely identify content in periods. This section identifies schemes that *may* be used in content compliant to this specification.

There *may* be several alternative identifiers applicable to the same content. **AssetIdentifier** elements are used for grouping periods with the same content and indicates to DASH client that one is continuation of the other. Thus, choice of main identifier (carried in **AssetIdentifier**) needs to reflect continuity in terms of user interface, random access, and trick modes. Alternate identifiers (carried in **SupplementalProperty**) are informational and may be used for reporting purposes.

As an example, when the same advertisement is shown multiple times, its instances will have same value of Ad-ID, but different AiringID. Use of Ad-ID as **AssetIdentifier** will result in DASH client considering the second instance of the advertisement a continuation of the first. In this case, e.g. the UI *may* show at the start of the second instance progress bar at 50% and advertisement duration of twice its actual duration. Thus AiringID *should* be used as **AssetIdentifier**, while Ad-ID *should* be used as an alternative identifier carried in **SupplementalProperty** with the same value of `@schemeIdUri`. If Ad-ID is to be used as the only identifier see section 8.3 for describing this.

In the opposite case, when two periods carry same in-network content which has the same EIDR and a period between them is an advertisement, the two in-network periods are continuation of each other and thus EIDR *should* be used in **AssetIdentifier**, while AiringID of these periods will be different and, if used, it *should* be used in **SupplementalProperty** . If EIDR is to be used as the only identifier for the period see section 8.3 for describing this.

9.2. UPID Content Identification scheme

The value of **AssetIdentifier**`@schemeIdUri` or **SupplementalProperty**`@schemeIdUri` for this scheme ***shall*** have the value "`urn:scte:dash:asset-id:upid:2015`". The content of this **AssetIdentifier** or **SupplementalProperty** descriptor ***shall*** contain one or more **ContentIdentifier** elements defined below.

Only **ContentIdentifier** elements with the same scope can appear in the same **AssetIdentifier** or **SupplementalProperty** element.

9.2.1. ContentIdentifier element semantics

Table 4 - UPID ContentIdentifier Element Semantics

Element or Attribute Name	Use	Description
ContentIdentifier		
`@type`	M	Type corresponding to SCTE 35 UPID type as specified in table 9-7. The value of this attribute ***shall*** be same as the value of `segmentation_upid()` (i.e., 3rd column of the table).

Element or Attribute Name	Use	Description
		MID ***shall not*** be used – the structure ***shall*** be translated into multiple UPID elements.
`@value`	M	Textual representation of the UPID value. It ***shall*** correspond to the description in the Description column (i.e., 4th column) of table 9-7 in SCTE 35. In case of the UPID contains binary encoding (e.g., ADI and ISAN), and a full textual representation is specified by the applicable standard, this textual representation ***shall*** be used. Otherwise, binary encoding is represented as a byte string in hexadecimal format.
Legend: For attributes: M=Mandatory, O=Optional, OD=Optional with Default Value, CM=Conditionally Mandatory. For elements: <minOccurs>...<maxOccurs> (N=unbounded) Elements are **`bold`**; attributes are non-bold and preceded with and @.		

9.2.2. XML syntax

Table 5 - UPID Content Identification XML Syntax

```
<xs:complexType name="UPID">
    <xs:sequence>
        <xs:any namespace="##other" processContents="lax"
                minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="type" type="xs:string" use="required"/>
    <xs:attribute name="value" type="xs:string" use="required"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
  </xs:complexType>
```

9.2.3. Example

Table 6 - UPID Content Identification Example

```
<AssetIdentifier schemeIDUri="urn:scte:dash:asset-id:upid:2015">
    <!-- ADI of the asset -->
    <scte214:ContentIdentifier type="ADI" value="cablelabs.com/MOVE1234567890123456>
    <!-- Alternative ID using an opaque provider-specific scheme -->
    <scte214:ContentIdentifier type="MPU" value="CSP1DE12AB327FE312AF"/>
</AssetIdentifier>

<SupplementalProperty schemeIDUri="urn:scte:dash:asset-id:upid:2015">
    <!-- AiringID of the asset -->
    <scte214:ContentIdentifier type="AiringID" value="0xDEADBEEF"/>
</AssetIdentifier>
```

9.3. Use of EIDR and Ad-ID

In case only a single identifier is needed, it is recommended to use EIDR or Ad-ID.

[DASH-IF IOP] v4.3 sec 5.5.1 normatively describes use of Ad-ID and EIDR in **`AssetIdentifier`** descriptors. The text below describes that approach.

EIDR: The value of `@schemeIdUri` is set to `"urn:eidr"` to signal the use of EIDR. The value of the `@value` attribute ***shall*** be a valid canonical EIDR entry.

Ad-ID: Use of Ad-ID for asset identification is signaled by setting the value of `@schemeIdUri` to "`urn:smpte:ul:060E2B34.01040101.01200900.00000000`" . The value of `@value` attribute ***shall*** be a canonical full 22 Ad-ID identifier as defined in SMPTE 2092-1.

10. Generic restrictions on media segments

10.1. General

10.1.1. SCTE Segment Model

This section defines a model restricting segment sizes and durations for improved predictability and resiliency of client implementation. Conformance to this model is signaled by a **`SupplementalProperty`** element with `@schemeIdURI` value of "`urn:scte:dash:segment-model:2024`"

10.1.2. Terminology

For the purpose of this section the following variables are defined for any segment *S(n)* and its k^{th} subsegment *n[k]*:

EPT(n) := earliest presentation time of segment *n*. *EPT(0) = 0*;

EPT(n[k]) := earliest presentation time of its subsegment *n[k]*. *EPT(n[0]) = EPT(n)*.

SD(R) := signaled segment duration for representation R, as expressed e.g. in the `@duration` attribute or the **`S`**`@d` in the **`SegmentTimeline`**. While this applies to a specific representation *R*, segment alignment requirement in this specification requires this value to be identical for all representations in an adaptation set.

MSD := maximum segment duration, as indicated in **`MPD`**`@maximumSegmentDuration`

MSSD := maximum subsegment duration, as indicated in **`MPD`**`@maximumSubsegmentDuration`

D(n) := "real" presentation duration of segment *n*, i.e. *EPT(n+1) – EPT(n)*.

SD(n[k]) , the signaled subsegment duration, is same as *D(n[k])*, presentation duration of segment *k*, and is provided in `` `sidx`.subsegment_duration[k] `` of the `sidx` box indexing segment *n*.

BW[R] := value of **`Representation`**`@bandwidth` of a representation *R* to which segment n belongs. **Note:** that this is the peak and not the average bandwidth.

MBT := value of **`MPD`**`@minBufferTime`

MSR[R] := maximum segment bitrate, defined in sec 10.3.2

τ : segment duration tolerance, defined as follows:

$$\tau := min\left(0.5, \frac{\textbf{SegmentTemplate}@\text{tolerance}}{100.0}\right)$$

All durations are in seconds, and bandwidth is given in bits per second.

All durations are in seconds, and bandwidth is given in bits per second.

10.2. Duration

10.2.1. Segments

If representation contains more than one segment or segment sequence and is used for normal playback, the following restrictions ***shall*** be met:

1. Unless the **`SegmentTimeline`** element is used to express precise timing, segments or segment sequences ***shall*** have almost equal "real" duration. The maximum tolerance of "real" segment duration *D(n)* ***shall*** be the value of *SD(R)*×τ, and the accumulated drift ***shall not*** exceed the value of *SD(R)*×τ.

$$\boldsymbol{abs}\left(\sum_{\boldsymbol{i=0}}^{\boldsymbol{n-1}} \boldsymbol{D(i)} - \boldsymbol{(n+1)} * \boldsymbol{SD(R)}\right) \leq \boldsymbol{SD(R)} * \boldsymbol{(1+\tau)}$$

   The last segment of the Period may be shorter than $\boldsymbol{SD(R)} * \boldsymbol{(1-\tau)}$

   Note 1: This is done so that if seeking is done using stated duration, correct segment will be identified despite the accumulating drift.

   Note 2: drift *may* develop due to mismatch between *D* and *SD* due to imprecision of the clock used to state SD. For example, if *SD*=2 sec and segments are 2002 ms each, a ±50% drift allowance will be exceeded in less than 10 minutes. Hence this recommendation recommends using sample-accurate timescales where feasible.

3. The above restrictions apply to segments or segment sequences, but not to partial segments or individual ISO-BMFF fragments comprising a single segment (e.g., CMAF chunks).

10.2.2. Subsegments

For representation used for normal playback and containing subsegments, the "real" subsegment duration ***shall*** be less than the defined value for MSSD.

$$D(n[k]) \leq SSD_{max} = MSSD;$$

10.2.3. Segment duration patterns

Audio content segments following the video content segment following a time-based addressing aligning to an audio sampling time can lead to the creation of alternative durations of some audio segments. This can result in a pattern of longer audio content segments followed by shorter audio content segments. This can lead to the creation of larger MPDs to represent this case.

One approach that can mitigate the **`SegmentTimeline`** growth due to patterning is “mixed addressing mode”. This mode employs `$Number$`-based addressing for audio and text segments while using addressing based on **`SegmentTimeline`** for video.

In order to maintain more predictable timing of audio segments, it is recommended to explicitly state the maximum tolerance using the **`SegmentTemplate`**`@tolerance` attribute. This will limit the difference between the time value used for segment retrieval and time value of the first sample of the segment. The above allows a good precision (e.g., ±20ms for AAC).

If sample-accurate precision is needed, the **`SegmentTemplate.Pattern`** element can be used to express a precise pattern.

In both precise pattern and mixed addressing mode, the mismatch between any audio and video segments $S_A(i)$ and $S_V(i)$ ***shall*** be within the following range:

$$-125ms \leq EPT(S_A(i)) - EPT(S_V(i)) \leq 45ms$$

Tolerance ***shall*** be explicitly signaled in the MPD.

10.3. Bandwidth, size, and buffering

10.3.1. Introduction

This section formalizes the relationship between the declared bandwidth *BW[R]*, *MBT*, *MSR[R]*, and segment sizes.

The derivations below are a straightforward, albeit lengthy, translation of the requirement in [DASH] that if segments of representation *R* are delivered over a constant bitrate channel with bitrate equal to *BW[R]* attribute, then each sample with decoding time *DT* is available for decoding at the media engine by time *DT* + *MBT*.

Note: In many cases the latter may differ from the limitations stated in the sections below, as the discussion below applies to complete (sub)segments, rather than samples. From the MPEG DASH standpoint, the maximum rate as defined in the HRD fulfills the requirement.

While MBT does specify minimum time sufficient for ensuring continuous playout of a representation, it describes content encoding properties, rather than expected network behavior. Hence a player implementation has to account for realistic network conditions, and this specification provides neither restrictions nor any guidance on these issues.

10.3.2. Segments

The maximum segment bitrate *MSR[R]* may be explicitly signaled in the MPD as the **`Representation`**`@scte214:maxSegmentRate` attribute. MSR[R] is defined as follows:

Let segment buffer SB_i be comprised of the shortest sequence of one or more consecutive segments or segment sequences *S(i)..S(i+k)* with combined media duration between *0.5×MSD* *1.5×MSD*. *Instantaneous segment bitrate* is defined as the sum of the sizes of the above segments (or segment sequences) in bytes divided by their combined duration. MSR[R] is defined as the maximum value of the instantaneous segment bitrate over the Period.

MSR[R] can be signaled explicitly as the **`Representation`**`@scte214:maxSegmentRate` attribute. In case *MSR[R]* value cannot be computed precisely, an expectation should be signaled. In this case *MSR[R]* ≤ 1.1×**`Representation`**`@scte214:maxSegmentRate`

If *MSR[R]* is not signaled, it is assumed to equal *BW[R]*.

Note 1: the above addresses the issue of very short video segments (smaller than half a segment duration) which inherently have higher bitrate (due to higher proportion of much larger I frames to P/B frames) and would have skew the rate calculation.

Note 2: the above definition is equivalent to the definition of `EXT-X-BITRATE` in section 4.4.4.8 of RFC 8216 [HLS].

Element or Attribute Name	Use	Description
`RepresentationType`		See ISO/IEC 23009-1
`@maxSegmentRate`	OD	specifies the maximum instantaneous bitrate of the current representation.
Legend: For attributes: M=Mandatory, O=Optional, OD=Optional with Default Value, CM=Conditionally Mandatory. For elements: <minOccurs>...<maxOccurs> (N=unbounded) Elements are **`bold`**; attributes are non-bold and preceded with and @.		

Let SD_{max} be the maximum segment duration. For representations containing more than one segment it is defined as follows:

$$SD_{max} = min(SD(R) * (1 + \tau), MSD)$$

Let $SZ_R(n)$ be the size (in bits) of segment *n* from representation *R*.

Note 1: $SZ_R(n)$ is the size of the complete segment including all headers, `sidx` and `ssix` boxes (for [ISOBMFF]) and inband events.

Let MBT_s be the minimum buffer time in units of segments, defined as $MBT_s = ceil\left(\frac{MBT}{SD_{max}}\right)$

Note 2: Buffer size of $1.1 * MSR[R] * SD_{max}$ is sufficient for playback of representation R under idealized network conditions (i.e., assuming constant download rate).

For any representation that contains *N>1* segments and is used for normal playback, the following restrictions ***shall*** be met:

1. Any segment *n* ***shall not*** exceed the buffer size; hence the following ***shall*** hold:
$$SZ_R(n) \leq 1.1 * MSR[R] * SD_{max}$$
2. Combined size of any MBT_s consecutive segments ***shall not*** exceed the buffer size, hence for any $0 \leq k \leq N - MBT_s$, the following ***shall*** hold:
$$\sum_{i=k}^{MBT_s} SZ_R(i) \leq 1.1 * MSR[R] * \sum_{i=k}^{MBT_s} D(i)$$

Note 1: In case of inband events care *should* be taken to keep events small enough in order not to break the model above.

Note 2: For representations without subsegments it is often useful to set *MBT* to SD_{max}. For representations containing subsegments SSD_{max} *may* be a better alternative. This agrees with the recommendation in [DASH-IF IOP] 4.3.

10.3.3. Video aspects

*MBT **shall** not* be smaller than coded picture buffer (CPB) removal delay.
*MBT*BW[R] **shall*** equal or exceed the size of CPB unless *MSR[R]* is explicitly signaled.

11. Codec-Specific Aspects

11.1. Video

11.1.1. Supported video codecs

The following video codecs ***shall be*** supported in SCTE DASH profiles:

1. AVC (ISO/IEC 14496-10, restrictions in [SCTE 128-1])
2. HEVC (ISO/IEC 23008-2, restrictions in [SCTE 215-1]) at least for UHD and HDR content.

Note: VVC (ISO/IEC 23090-3, restrictions in [SCTE 281-1]) support *may* be required in future versions of this specification.

11.1.2. Resolutions and frame rates

This specification neither specifies nor requires support for specific operating points (i.e., combination of resolution, frame rate and aspect ratio).

11.1.3. SAP values

11.1.3.1. AVC video

Segments starting from an IDR picture in decoding order have SAP value of 1, unless this IDR picture is followed by a picture which precedes it in presentation order. In the latter case the segment has SAP value of 2.

11.1.3.2. HEVC video

Segments starting from pictures with nal_unit_type equal to IDR_N_LP or BLA_N_LP have SAP value of 1.

Segments starting from IDR_W_RADL or BLA_W_RADL have SAP value of 2.

11.1.3.3. VVC video

Segments starting from pictures with nal_unit_type equal to IDR_N_LP or CRA_NUT have SAP value of 1.

Segments starting from IDR_W_RADL have SAP value of 2.

11.1.4. Multiplexed MPEG-2 TS segments

When a segment contains video and one or more audio elementary streams, its SAP value is the SAP value of the video elementary stream.

11.1.5. Colorimetry

`AdaptationSet.SupplementalProperty or AdaptationSet.EssentialProperty` descriptors ***shall*** be used to signal source signal information in the manifest to indicate color primaries, optoelectronic transfer characteristics, and matrix coefficients properties for derivation of luma and chroma signals. If all video adaptation sets contain only BT.709 colorimetry, the consistent usage of either descriptor is recommended.

If DVB compatibility is needed, **`AdaptationSet.EssentialProperty`** ***should*** be used. Please see [ETSI TS 103 285] sec 5.2.6 for backwards-compatible HLG signaling.

Note 1: In absence of the colorimetry signaling due in the high level syntax of the encoded stream, BT.709 SDR colorimetry *may* be assumed by the receiver.

The URNs and corresponding values are defined in [CICP-Video] and are informatively provided in the table below.

Table 7 – Colorimetry

`@schemeIdUri`	`@value`
`urn:mpeg:mpegB:cicp:ColourPrimaries`	See ISO/IEC 23091-2 sec. 8.1
`urn:mpeg:mpegB:cicp:TransferCharacteristics`	See ISO/IEC 23091-2 sec. 8.2
`urn:mpeg:mpegB:cicp:MatrixCoefficients`	See ISO/IEC 23091-2 sec. 8.3

11.1.6. Supplemental Codecs and Supplemental Profiles

Supplemental codecs are defined as backwards-compatible codecs enhancing the experience of a base layer codec as described as **`AdaptationSet`** attributes in part 15 of section 6.3. Annexes A and B of [SCTE 215-1] specify such codecs, DolbyVision profile 8.1, HDR10+. In both HDR cases, the "base layer" is HEVC HDR10, support for which is essential for rendering. If supported, use of the optional DolbyVision or HDR10+ supplemental codec or profile enhances the viewers' experience.

The use of Supplemental codecs or profiles can also be used for other purposes such as indicating single track LCEVC media content.

This specification defines two additional XML attributes used within the RepresentationBaseType element defined in ISO/IEC 23009-1 [DASH].

Table 8 – Supplemental XML Attrributes

Attribute Name	Use	Description
Common attributes and elements		See ISO/IEC 23009-1

Attribute Name	Use	Description
`@scte214:supplementalCodecs`	O	specifies a space-separated list of codec strings for supplemental codecs used in the associated Representation(s). These codec strings ***shall*** comply with the syntax defined in IETF RFC 6381 When multiple codec strings are present, the order of codec appearance is immaterial.
`@scte214:supplementalProfiles`	O	specifies a space-separated list of supplemental segment profiles (expressed as ISOBMFF brands or URIs) to which the associated Representation(s) conform.

NOTE 1: These attributes are MP4RA-registered code points [MP4RA]. The difference between `supplementalCodecs` and `supplementalProfiles` is that the former includes profiles and levels as codec code points per IETF RFC 6381[RFC 6381] while the latter specifies brands as brand code points.

NOTE 2: The concept of supplemental codecs is not limited to HDR dynamic metadata or LC-EVC – this same approach may work for NAL-based optional features existing within the same media content segment.

NOTE 3: In case the optional codec strings differ across representations within an adaptation set, the `supplementalCodecs` attribute will appear within **`Representation`** elements. In general, the logic of determining the optional codec string is same as the one used for the codecs in the `@codecs` attribute defined in ISO/IEC 23009-1. The examples in this section show the case the optional codec strings are shared across all representations and hence appear within the **`AdaptationSet`** element.

11.1.7. Signaling Examples for supplementalCodecs and supplementalProfiles

This section provides examples for the use of the `supplementalCodecs` and `supplementalProfiles` attributes.

11.1.7.1. HDR10-based formats

HDR10 formats can be HDR10 plus optional dynamic metadata information. The types of combinations described in this standard are HDR10, HDR10 +DM App#1, or HDR10 +DM App#4 used for segments that comply to CMAF profiles for HEVC as defined in [CMAF] Annex B or for [VVC] as defined in [CMAF] Annex M.

For the nominal use case, content providers ***shall*** include ‘chd1’ (HDR10) for HEVC and ‘cvvc’ for VVC [VVC] (Baseline Media Profile) in the list of compatible brands in the ‘ftyp’ box if the underlying video complies with the above. This is to be placed in the `@containerProfiles` attribute along with all applicable DM `compatible_brands` as defined in [CMAF] Table 7 column 2. HEVC Media Profiles and Track Brands *should* conform to clause [CMAF] Section B.5 except for [CMAF] Table 6 if the underlying video conforms to them.

Note: the above does not preclude usage of any other MP4RA-registered brands.

Conformance and playback requirements on the CMAF tracks are specified by the HEVC media profile HDR10, including conformance requirements specified for HDR10 in [CMAF] , conformance requirements for HEVC specified in ISO/IEC 23008-8 [HEVC].

Conformance and playback requirements on the CMAF tracks are specified by the VVC Baseline media profile HDR10, including conformance requirements specified for Baseline VVC in [CMAF] , conformance requirements for VVC specified in ISO/IEC 23090-3 [VVC] .

11.1.7.2. Backwards-compatible dynamic metadata examples

The following informative examples indicate how to signal different variations of HDR formats. There may be other values or additional values that could be added to the `@containerProfiles`, `@segmentProfiles`, `@supplementalProfiles`, or `@supplementalCodecs` attributes.

Example 1: HEVC/HDR10
The example below shows a partial **`AdaptationSet`** element describing an adaptation set containing an HDR10 stream encoded using HEVC Main10 profile Main tier Level 5.1 HEVC Codec. An optional `chd1` brand indicates compliance with the `chd1` media profile.

```
<AdaptationSet …
 codecs="hev1.2.4.L153.B0" containerProfiles="cmfc chd1">
<EssentialProperty schemeURI="urn:mpeg:mpegB:cicp:MatrixCoefficients"
value="9"/>
<EssentialProperty schemeURI="urn:mpeg:mpegB:cicp:ColourPrimaries"
value="9"/>
<EssentialProperty schemeURI="urn:mpeg:mpegB:cicp:TransferCharacteristics"
value="16"/>
…
</AdaptationSet>
```

Example 2: Dolby Vision Profile 8.1
The example below shows the signaling for Dolby Vision Profile 8.1(using non-standard syntax). In this profile the HEVC Main10/Main/L5.1 base layer is mandatory while the dynamic metadata is optional and serves to enhance the viewers' experience.

For simplicity, we omit the colorimetry signaling which is identical to Example 1. The supplemental Dolby Vision 8.1 signaling includes both a codec string (indicating profile Dolby Vision profile 8) and an additional brand db1p indicating that the underlying base layer is PQ.

```
<AdaptationSet …
codecs="hev1.2.4.L153.B0" containerProfiles="cmfc chd1"
scte214:supplementalCodecs="dvh1.08.09" scte214:supplementProfiles="db1p">
…
</AdaptationSet>
```

Note: This example matches the practice recommended since v2.3 of the Dolby recommendation available at

https://dolby.my.salesforce.com/sfc/p/#700000009YuG/a/4u00000l6FV/TqYPD0f0c3Zm40JJrFKFn1m29QdUxuVB.U_GzZzbXgE

Example 3: HDR10+
The example below shows signaling of HDR10+. In this case the HEVC Main10/Main/L5.1 base layer is mandatory while the dynamic metadata is optional (and present as SEIs) and serves to enhance the viewers' experience. The format is defined in CTA WAVE [reference goes here] as `cdm4`, which is indicated in the @supplementalProfiles attribute.

```
<AdaptationSet …
codecs="hev1.2.4.L153.B0" containerProfiles="cmfc chd1"
scte214:supplementalProfiles="cdm4">
…
</AdaptationSet>
```

Example 4: Combination of Dolby Vision 8.1 and HDR10+
This example shows a stream where both dynamic metadata approaches are used. In this case the HEVC Main10/Main/L5.1 base layer is mandatory while the dynamic metadata is optional and both Dolby Vision and HDR10+ metadata is present. In this case the receiver capable of dynamic metadata will select any of the systems it is capable of playing.

```
<AdaptationSet …
codecs="hev1.2.4.L153.B0" containerProfiles="cmfc chd1"
scte214:supplementalCodecs="dvh1.08.09"
scte214:supplementProfiles="db1p cdm4">
…
</AdaptationSet>
```

11.1.7.3. Additional uses

Optional features contained within the same media segment contained through separate NAL units can be signaled using the `@supplementalCodecs` or/and `@supplementProfiles` attribute.

Example 1: HEVC with LCEVC enhancement layer
This signaling indicates that both mandatory HEVC Main10/Main/L5.1 base layer and the optional LCEVC enhancement NAL units are present in the same sample.

```
<AdaptationSet …
codecs="hev1.2.4.L153.B0" containerProfiles="cmfc chh1"
scte214:supplementalCodecs="lvc1.04.01">
…
</AdaptationSet>
```

11.2. Audio

11.2.1. Supported codecs

The following audio codecs are supported in SCTE DASH profiles:

1. (E-)AC-3 (ATSC A/52 [ATSC A/52] restrictions in A/53 [ATSC A/53] Parts 5-6), AC-3 is only supported for MPEG-2 TS
2. AAC ([ISOBMFF-TT] restrictions in [SCTE 193-1])
3. DTS-HD (ETSI TS 102 114) , restrictions in [SCTE 194-1])

The following next generation audio codecs could be supported in the SCTE DASH Profiles:

1. AC-4 (restrictions in [SCTE 243-2]
2. MPEG-H (restrictions in [SCTE 243-3])
3. DTS-UHD (restrictions in [SCTE 243-4])

11.2.2. SAP values

For AC-3, E-AC-3, DTS and AAC, all segments ***shall*** have SAP value of 1.

AAC segments ***shall*** be start with a RAP AU (as defined in [SCTE 193-1]) and *should* be encoded according to the MPEG DASH Implementation Guidelines sec. 5.1.2 in order to ensure seamless bitstream switching.

For AC-4, all segments ***shall*** have an SAP.

I-frames ***shall*** be placed temporally aligned with the I-frame of the video to enable synchronous switching. AC-4 segments shall be encoded temporally aligned with the video segments from the corresponding video to ensure continuous alignment of video and audio access units in order to utilize features of A/V alignment.

MPEG-H Audio segments ***shall*** start with a RAP AU (as defined in [SCTE 243-3]). In order to ensure seamless bitstream switching alignment of video and audio access units at the segment boundary, the last sample of the MPEG-H Audio segment may contain a truncation message (PACTYP_AUDIOTRUNCATION).

11.3. Trick Modes

11.3.1. Introduction

Playback of media content at speed and / or direction other than the ones intended for normal playback of this asset is referred to as *trick modes*. Trick modes include modes like fast forward, slow motion, and rewind; and are used to emulate visual experience of rewinding analog videotapes.
Trick modes can be implemented in multiple ways, starting from fetching segments at a different speed, to maintaining special trick mode representations, to bringing only specific frames from the segment. This standard does not prescribe a particular implementation strategy or combination of strategies. [ETSI TS 103 285] sec. 6.2 provides a long discussion about ways of implementing trick modes in DASH, while encoding techniques discussed in [SCTE 128-1] provide a content preparation perspective.

Note: [SCTE 128-1] and [SCTE 215-1] discuss trick modes based on extraction of identifiable pictures that result in respective decodable sub-bitstreams, or conversely, on discarding identifiable pictures to obtain respective decodable sub-bitstreams. This functionality can be implemented using Subsegment Index ('ssix') boxes.

Trick modes are not necessarily permitted in all content – sometimes certain modes will be disallowed. This restriction model is described in [SCTE 130-10], and sec. 12.3 defines its integration into DASH MPD.

11.3.2. Trick mode representations

`Periods` *may* contain adaptation sets with representations intended for use in trick modes (e.g., representations with low frame rate). Such adaptation sets ***shall*** employ signaling as defined in [DASH-IF IOP] 4.3 with additional signalling defined below.

Trick mode adaptation sets ***shall*** be marked with a **`SupplementalProperty`** or **`EssentialProperty`** element with `@schemeIdUri` value of "`http://dashif.org/guidelines/trickmode`" and the `@value` the value of the **`AdaptationSet`**`@id` attribute of the adaptation set(s) to containing "normal" (non-trick-mode) representations of the same content.

11.3.3. I-Frame track representations

Fixed frame rate trick mode adaptation sets *may* create segment boundary mismatches unless all segments in the "normal" representation have precisely the same number of frames.

In this case it is possible to have a trick mode adaptation set aligned with "normal" adaptation set(s), where the segment boundaries of the trick mode representations match the segment boundaries of the "normal" representations. As a result, the frame rate *may* be variable due to variable duration of the "normal" representation segments.

The I-Frame representations defined in this clause are a subset of DASH-IF trick modes maintaining segment alignment for variable-duration segments and compatible with IETF RFC 8261 (HLS).

The following ***shall*** hold for I-Frame track representations:

1. Each media segment ***shall*** consist of a single IDR frame, and its EPT ***shall*** match the EPT of the corresponding "normal" media segment;

NOTE: The above implies that there can be more single-frame segments than "normal" segments – an I-frame track may be used for functionality such as "fast join". The above is achieved using Segment Sequence Representations [currently defined in ISO/IEC 23009-1 5th ed ., AMD3], and restricted SCTE 214-6. (TBP)

2. DASH-IF trick mode signaling ***shall*** be present as describe in 11.3.2 above.

3. There ***shall*** be a **`EssentialProperty`** element with `@schemeIdUri` value of "`urn:scte:dash:i-frame-track:2021`".

4. The value of the `@frameRate` attribute ***shall*** be a long-term approximation of the frame rate (i.e., average frame rate per period or a reasonable expectation of it in case of live content). For example, for segments with target duration of 2 seconds and occasional short segments due to ad insertion, this value *should* be 1/2.

12. Multi-period assets

12.1. Period continuity

If multi-period content is offered (e.g., when some of the periods represent placement opportunities), periods with identical **`AssetIdentifier`** elements are considered as contiguous parts of the same asset.

If an asset spans over more than one period, **`Period.AssetIdentifier`** element *should* be present in each such period.

`Periods` with identical asset identifiers ***shall*** be *period-continuous* as specified in [DASH].

12.2. Asset boundaries

If multi-period content is offered in a dynamic MPD, periods can be removed and/or added during the presentation. In these cases, the author *may* want to preserve the information regarding the playback location in time in order to allow e.g. correct display of time in UI.

If a period is the last period of a given asset, this *may* be signaled using **`Period.SupplementalProperty`** with `@schemeIdUri="urn:scte:dash:asset-end".` Correspondence of *PeriodStart* to the time of the asset *may* be signaled using **`Period.SupplementalProperty`** with `@schemeIdUri="urn:scte:dash:asset-time".` The value of `@value` attribute ***shall*** be the timestamp corresponding to *PeriodStart*, as NPT or SMPTE relative timestamp, as defined in RFC 2326.

Correspondence of *PeriodStart* to UTC time *may* be signaled using **`Period.SupplementalProperty`** with `@schemeIdUri="urn:scte:dash:utc-time".` The value of `@value` attribute ***shall*** be the timestamp corresponding to *PeriodStart*, in format defined in [RFC 3339].

Note: The difference between the asset time and UTC time is that asset time is relative to the asset start, while UTC time is the UTC time corresponding to the acquisition time of the first sample of the period. Thus, asset time will show that a period starts at 42nd minute of an asset, while UTC time will show that the period starts with content sent to origin on October 21, 2015 at 4:29am.

12.3. Stream restrictions

Period elements used in multi-period assets *may* contain a **`SupplementalProperty`** element with **`SupplementalProperty`**`@schemeIdUri` value of `"urn:scte:scte130-10:2014 "`. The content of the descriptor is the SCTE 130-10 **`StreamRestrictionList`** element.

`NptRange` in this descriptor ***shall*** be relative to *PeriodStart* and the restrictions ***shall*** be valid only for the duration of the period in which the **`SupplementalProperty`** element appears.
Note: Given `@nptstart` value of N_s, `@nptend` value of N_e, and period duration D, the restrictions in the **`StreamRestrictionList`** element are valid in the range [max(0:00.00, N_s), min(D,N_e)].

13. URL query parameters for XLink

13.1. Introduction

The only interface between an entity dereferencing XLink and the DASH client is the XLink URL. Passing avail metadata such as SCTE 35 cue messages *may* be useful when remote periods are used to signal upcoming avails. This usage is described in [DASH-IF IOP] and referred to as "server-based ad insertion".

The section below defines standardized names and values for parameters that can be used in XLink URLs. It defines the parameter name and the derivation of its value. Name and value ***shall*** contain only characters permitted by RFC 3986 and *may* need to be percent-coded.

The parameters defined in this section are not mandatory – i.e., the author is not required to use them. On the other hand, query parameter with the name defined in this section appears in a URL, its value ***shall*** be derived in a way defined in this section. The author ***shall not*** use query parameter names defined in this section with any different syntax and semantics.

The parameters in this section ***shall not*** be used in elements other than Period.

13.2. Notation

The standardized parameters introduced in this section are <name>=<value> pairs embedded in a URL query string (as defined in RFC 3986). The <field> string in the above construct will be replaced with parameter name For example, for parameter named `param` and having value `42`, the URL will be http://example.com?param=42

Note: <name> is often referred to as "field" in different descriptions of URL query parameter syntax.

13.3. Carriage of SCTE 35 in a query parameter

Parameters defined in this section carry the complete SCTE 35 binary cue message, as well as message timing.

Table 9 - Carriage of SCTE 35 in a Query Parameter

<name>	<value>	format
`scte35-cue`	`splice_info_section()`	base64url-encoded string with padding. Pad character ('=') replaced with "%3D" See Note 1 and 2 below.
`scte35-time`	Offset of splice start from presentation start time. This is a translation of splice_time() into presentation timeline. The time *should* be provided with millisecond accuracy if feasible. See note 3 below	`xsd:duration` (defined in XML Schema Part 2, based on ISO 8601)

Note 1: Percent encoding is required to escape reserved characters in query string. Thus '=' character used as a padding character in base64 and base64url thus needs to be replaced with %3D. While splice_info_section() size is provided within the section, skipping padding per RFC 4648 sec 3.2 is explicitly disallowed in the definition above for simplicity and interoperability reasons.

Note 2: The definition in this section does not preclude use of SCTE 35 in DASH events (either inband or MPD), as defined in this standard.

Example

Table 10 - Remote Period element with SCTE 35 embedded in XLink URL

```
...
<Period duration="PT60.0S" id="ad break #1"
         xlink:href="https://adsrus.com/avail.mpd?scte35-cue=
                     DAIAAAAAAAAAAAQAAZ_I0VniQAQAgBDVUVJQAAAAH+cAAAAAA%3D%3D"
         xlink:actuate="onLoad" >

     <!-- Default content, replaced by elements from remote entity -->
     <AdaptationSet mimeType="video/mp4" codecs="avc1.640828"
                    frameRate="30000/1001"
                    segmentAlignment="true" startWithSAP="1">
        <BaseURL availabilityTimeOffset="INF">default_ad/</BaseURL>
        <SegmentTemplate timescale="90000" initialization="$Bandwidth%/init.mp4v"
                         media="$Bandwidth%/$Time$.mp4v"/>
            <Representation id="v0" width="320" height="240" bandwidth="250000"/>
            <Representation id="v1" width="640" height="480" bandwidth="500000"/>
            <Representation id="v2" width="960" height="720" bandwidth="1000000"/>
     </AdaptationSet>
</Period>
```

13.4. Carriage of geographical information in a query parameter

This section lists parameters can be used for carriage of geographical information in URLs.
Note: that these parameters may have privacy implications, hence it is recommended to use HTTP over TLS (e.g., HTTPS) if the HTTP GET with these parameters is expected to be sent via an insecure channel.

Table 11 - Carriage of Geographical Information in a Query Parameter

<name>	<value>	format
`scte-dash-syscode`	`syscode`	Decimal number, assigned by NCC Media

14. Annotation

14.1. Generator identification

It is often necessary for debug purposes to indicate which software generated an MPD or responded to an XLink dereferencing request. In order to allow such information, the following schemes can be used in **`MPD.SupplementalProperty`** and/or in **`Period.SupplementalProperty`** elements:

Table 12 - Generator identification

@schemeIdUri	@value
urn:scte:dash:powered-by:2016	Human-readable string containing software name and version that generated this MPD or responded to the XLink dereferencing request
urn:scte:dash:generation-info:2016	Comma-separated list of name=value pairs with one or more of the following values: 1. location=<location>: same as location header in HTTP (RFC 7213), as an absolute URI 2. date=[time]: generation time (UTC) of the body of the HTTP response containing the , in format specified by ISO 8601. The time *should* be provided with millisecond accuracy if feasible.
urn:scte:dash:generation-request:2016	Request target string (per [RFC 9112]sec 3.2) from the HTTP request which resulted in generating the current response (MPD or Period). Example: value of syscode, assuming scte-dash-syscode query parameter was used

Example

For remote period from example in Table 10, the remote entity (i.e., the contents of the HTTP response) containing generator identification information is described in Table 13 below.

In the example below the XLink URL from in Table 10 (sans host name) is reflected in the generation request information, while the arrival time of the XLink request and the server name of the responding server are reflected in the generation info.

Table 13 - Remote entity with generator information

```
<Period duration="PT60.0S" id="inserted ad #1" >

    <SupplementalProperty schemeIdUri="urn:scte:dash:generation-info:2016"
        value="location=adsrus.com date=2016-01-07T15:22:16-07:00"/>

    <SupplementalProperty schemeIdUri="urn:scte:dash:generation-request:2016"
        value="/avail.mpd?scte-dash-syscode=123456789&&scte35-
        cue=DAIAAAAAAAAAAAQAAZ_I0VniQAQAgBDVUVJQAAAAH+cAAAAAA%3D%3D"/>

    <!-- Replaced ad content, replacing by elements from remote entity -->
    <AdaptationSet mimeType="video/mp4" codecs="avc1.640828"
        frameRate="30000/1001"
        segmentAlignment="true" startWithSAP="1">
        <BaseURL availabilityTimeOffset="INF">ad/</BaseURL>
        <SegmentTemplate timescale="90000" initialization="$Bandwidth%/init.mp4v"
            media="$Bandwidth%/$Time$.mp4v"/>
        <Representation id="v0" width="320" height="240" bandwidth="250000"/>
        <Representation id="v1" width="640" height="480" bandwidth="500000"/>
        <Representation id="v2" width="960" height="720" bandwidth="1000000"/>
    </AdaptationSet>
</Period>
```


15. Content Protection

15.1. Multi-key encryption

There *may* be cases in which there is a need to encrypt content of the same type in the same MPD with different keys. This requirement often arises when the output protection requirements differ across different video resolutions.

In this case, subsets of representations sharing the same key will be combined into adaptation sets. For example, a 4K video ladder, *may* be broken into three adaptation sets, SD (up to 540p resolution), HD (720p and 1080p resolution) and UHD (above 1080p resolutions). These adaptation sets are aligned and are seamlessly switchable.

`AdaptationSet``@profiles` value of `scte:dash:multi-key:2021` *should* appear in all adaptation sets following the constraints in this clause.

Note: This implementation conforms to the [DASH-IF IOP] guidelines. Other implementations of multi-key encryption *may* be legal but discouraged unless needed for legacy devices.

1. All representations in an **`AdaptationSet`** ***shall*** share the same key;

2. Adaptation sets containing differently encrypted parts of perceptually identical content ***shall*** be aligned and switchable per definition of adaptation set switching in [DASH].

3. All adaptation sets in the above switchable adaptation sets ***shall*** have identical values of the following elements and attributes:

    a. **`AudioChannelConfiguration`**

    b. **`Accessibility`**

    c. **`Role`**;

    d. `@audioSamplingRate`;

    e. `@lang`;

    f. `@codecs`, unless codecs differ by profile/level.

4. If the use of different keys is due to robustness or/and output protection, then these ***should*** be explicitly stated:
    a. If a particular level of DRM robustness (e.g. hardware decryption and decoding) is required, this needs to be specified in the **`ContentProtection`**`@robustness` attribute. The value of the attribute is DRM-specific and is typically described in the DRM documentation.

    b. If a particular HDCP level is required for playback of a specific adaptation set, then the **`OutputProtection`** descriptor ***shall*** be used

Note: Use of **`ContentProtection`**@`robustness` and **`OutputProtection`** signaling is needed to prevent download of segments which are expected to be undecodable by the client. The source of truth is the DRM, and not the above attributes. This means that it in the end is a proprietary DRM decision whether a client is or is not allowed to decode specific representations.

Annex A. URNs

The following URNs are defined in this specification:

urn:scte:dash:scte214-extensions
XML namespace for this specification (to be used post 2021)

urn:scte:dash:2021
XML namespace for this specification (for 2021 and prior editions)

urn:scte:dash:essential-event:2015
Event essentiality, see 7.7.3

urn:scte:scte35:2013:xml
XML representation of SCTE 35, see 7.7.2.1

urn:scte:scte35:2014:xml+bin
base64-coded representation of SCTE 35 wrapped in XML, see 7.7.2.1

urn:scte:scte35:2013:bin
inband SCTE 35 event, see section 7.7.3

urn:scte:dash:associated-service:2015
Roles for non-accessibility associated audio services, see 8.1

urn:scte:dash:cc:cea-608:2015
Signaling of CTA-608 closed captions, see 8.2.3

urn:scte:dash:cc:cea-708:2015
Signaling of CTA-708 closed captions, see 8.2.2

urn:scte:dash:asset-id:upid:2015
Asset identifier scheme based on SCTE 35 UPIDs, see 9.2

urn:scte:dash:asset-end
End of last period of an asset, see 12.2

urn:scte:dash:asset-time
NPT corresponding to asset time, see 12.2

urn:scte:dash:utc-time
UTC time corresponding to asset time, see 12.2

urn:scte:scte130-10:2014
Carriage of [SCTE 130-10], see 12.3

urn:scte:dash:powered-by:2016
Information on MPD / XLink authoring software

urn:scte:dash:generation-info:2016
Time and place of MPD or XLink remote entity generation

urn:scte:dash:generation-query:2016
Time and place of MPD or XLink remote entity generation

urn:scte:dash:default-language
Information on default language in MPD see section 7.1.1.

urn:scte:dash:i-frame-track:2021
Information on I-Frame representation

urn:scte:dash:segment-model:2024
Conformance to segment model as specified in sec 10

