



Information technology — Dynamic adaptive streaming over HTTP (DASH) —

Part 1: Media presentation description and segment formats

TECHNICAL CORRIGENDUM 1

Technologies de l'information — Diffusion en flux adaptatif dynamique sur HTTP (DASH) —

Partie 1: Description de la présentation et formats de remise des médias

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to ISO/IEC 23009-1:2013 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

Replace

4.5.1 Switching and Random Access Support

The formats defined in this part of the standard are designed for providing good user experience even in case the access bandwidth of the DASH segment delivery or the cache varies. A key functionality is the ability that the DASH client can seamlessly switch across different Representations of the same media component. DASH clients may use the common timeline across different Representation representing the same media component to present one Representation up to a certain time t and continue presentation of another Representation from time t onwards. However, in practical implementations, this operation may be complex, as switching at time t may require parallel download and decoding of two Representations. Therefore, providing suitable switching opportunities in regular time intervals simplifies client implementations. The

standard provides means for providing suitable switching opportunities and in addition provides abilities to signal the position and media time of the switching opportunities.

For this purpose this subsection defines three relevant concepts to support seamless switching, namely

- Media stream access points in 4.5.2 to signal positions where to easily switch to a Representation, and in addition where to suitable access a Representation at start-up or seek.
- Non-overlapping Segments and Subsegments in 4.5.3 to signal that at the signalled stream access points, no overlap decoding of Representations is necessary in order to provide a continuous switch
- Bitstream concatenation in 4.5.4 to signal that the concatenation of two Representations at a switch point results in a conforming bitstream

These three properties are neither sufficient nor necessary for seamless switching, but certain implementation or profiles may use these properties in order to simplify practical implementations.

with

4.5.1 Switching and Random Access Support

The formats defined in this part of the standard are designed for providing good user experience even in case the access bandwidth between the DASH segment delivery function or the cache varies. A key functionality is the ability that the DASH client can seamlessly switch across different Representations of the same media component without severely impacting the user experience.

Assume two Representations A and B. A switch from Representations A to Representation B at media time t is considered seamless, if the result of the presentation after this switch is applied is identical as if Representation A is decoded from the beginning and presented up to time t and Representation B is decoded from the beginning and presented from time t onwards.

Media Presentations may provide different Representations in one Adaptation Set representing the same media component. If such Representations are properly time-aligned (as expected by the Media Presentation), then DASH clients may apply seamless switching across different Representations provided in one Adaptation Set at any time t to obtain a perceptually continuous experience.

However, in practical implementations, the operation of seamless switching may be complex, as switching at time t may require parallel download and decoding of two Representations. Therefore, providing suitable switching opportunities in regular time intervals simplifies client implementations. This part of ISO/IEC 23009-1 provides means for providing suitable switching opportunities and in addition provides abilities to signal the position and media time of the switching opportunities.

For this purpose this subsection defines three relevant concepts to support seamless switching:

- Media stream access points in 4.5.2 to signal positions where to easily switch to a Representation, and in addition where to suitable access a Representation at start-up or seek.
- Non-overlapping Segments and Subsegments in 4.5.3 to signal that at the signalled stream access points, no overlap decoding of Representations is necessary in order to provide a continuous switch.
- Bitstream concatenation in 4.5.4 to signal that the concatenation of two Representations at a switch point results in a conforming bitstream.

These three properties are neither sufficient nor necessary for seamless switching, but certain implementation or profiles may use these properties in order to simplify practical implementations.

In 4.7 Schemes, Table 2 replace

urn:mpeg:dash:23003:3:audio_channel_configuration:2011	5.8.5.4	channel configuration.
urn:mpeg:dash:outputChannelPositionList:2012	5.8.5.4	a list of output channel position to signal individual speaker positions as defined in ISO/IEC 23001-8.

With

urn:mpeg:dash:23003:3:audio_channel_configuration:2011	5.8.5.4	channel configuration. NOTE: legacy format for backward-compatibility, it is recommended to use the signaling as defined in ISO/IEC 23001-8 instead.
urn:mpeg:dash:outputChannelPositionList:2012	5.8.5.4	a list of output channel position to signal individual speaker positions as defined in ISO/IEC 23001-8. NOTE: legacy format for backward-compatibility, it is recommended to use the signalling as defined in ISO/IEC 23001-8 instead.

In 5.2.3.2 Elements and Attributes added in this Revision replace

This revision adds the following elements and attributes to the schema defined in Annex B compared to the 2012 revision (ISO/IEC 23009-1:2012/Cor 1:2013) of this part of the standard:

- **MPD@publishTime**
- **Period.AssetIdentifier**
- **Period.EventStream**
- **RepresentationBase.InbandEventStream**
- **SegmentBase@availabilityTimeOffset**
- **SegmentBase@availabilityTimeComplete**
- **BaseURL@availabilityTimeOffset**
- **BaseURL@availabilityTimeComplete**
- **Subset@id**

with

This revision adds the following elements and attributes to the schema defined in Annex B compared to the 2012 revision (ISO/IEC 23009-1:2012/Cor 1:2013) of this part of the standard:

- **MPD@publishTime**
- **MPD.EssentialProperty**
- **MPD.SupplementalProperty**
- **Period.AssetIdentifier**
- **Period.EventStream**
- **Period.SupplementalProperty**
- **RepresentationBase.InbandEventStream**
- **SegmentBase@availabilityTimeOffset**
- **SegmentBase@availabilityTimeComplete**
- **BaseURL@availabilityTimeOffset**
- **BaseURL@availabilityTimeComplete**
- **Subset@id**
- **SegmentTimeline.S@n**

In 5.3.1.2 Semantics, Table 3 add at the end

EssentialProperty	0 ... N	specifies information about the containing element that is considered essential by the Media Presentation author for processing the containing element. For details see 5.8.4.8.
SupplementalProperty	0 ... N	specifies supplemental information about the containing element that may be used by the DASH client optimizing the processing. For details see 5.8.4.9.

In 5.3.1.3 XML syntax replace

```

<!-- MPD Type -->
<xs:complexType name="MPDtype">
  <xs:sequence>
    <xs:element name="ProgramInformation" type="ProgramInformationType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="BaseURL" type="BaseURLType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Location" type="xs:anyURI" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Period" type="PeriodType" maxOccurs="unbounded"/>
    <xs:element name="Metrics" type="MetricsType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string"/>
  <xs:attribute name="profiles" type="xs:string" use="required"/>
  <xs:attribute name="type" type="PresentationType" default="static"/>
  <xs:attribute name="availabilityStartTime" type="xs:dateTime"/>
  <xs:attribute name="availabilityEndTime" type="xs:dateTime"/>
  <xs:attribute name="publishTime" type="xs:dateTime"/>
  <xs:attribute name="mediaPresentationDuration" type="xs:duration"/>
  <xs:attribute name="minimumUpdatePeriod" type="xs:duration"/>
  <xs:attribute name="minBufferTime" type="xs:duration" use="required"/>
  <xs:attribute name="timeShiftBufferDepth" type="xs:duration"/>
  <xs:attribute name="suggestedPresentationDelay" type="xs:duration"/>
  <xs:attribute name="maxSegmentDuration" type="xs:duration"/>
  <xs:attribute name="maxSubsegmentDuration" type="xs:duration"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- Presentation Type enumeration -->
<xs:simpleType name="PresentationType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="static"/>
    <xs:enumeration value="dynamic"/>
  </xs:restriction>
</xs:simpleType>

```

with

```

<!-- MPD Type -->
<xs:complexType name="MPDtype">
  <xs:sequence>
    <xs:element name="ProgramInformation" type="ProgramInformationType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="BaseURL" type="BaseURLType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Location" type="xs:anyURI" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="Period" type="PeriodType" maxOccurs="unbounded"/>
    <xs:element name="Metrics" type="MetricsType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="EssentialProperty" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="SupplementalProperty" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="id" type="xs:string"/>
  <xs:attribute name="profiles" type="xs:string" use="required"/>
  <xs:attribute name="type" type="PresentationType" default="static"/>
  <xs:attribute name="availabilityStartTime" type="xs:dateTime"/>
  <xs:attribute name="availabilityEndTime" type="xs:dateTime"/>
  <xs:attribute name="publishTime" type="xs:dateTime"/>
  <xs:attribute name="mediaPresentationDuration" type="xs:duration"/>
  <xs:attribute name="minimumUpdatePeriod" type="xs:duration"/>
  <xs:attribute name="minBufferTime" type="xs:duration" use="required"/>
  <xs:attribute name="timeShiftBufferDepth" type="xs:duration"/>
  <xs:attribute name="suggestedPresentationDelay" type="xs:duration"/>
  <xs:attribute name="maxSegmentDuration" type="xs:duration"/>
  <xs:attribute name="maxSubsegmentDuration" type="xs:duration"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- Presentation Type enumeration -->
<xs:simpleType name="PresentationType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="static"/>
    <xs:enumeration value="dynamic"/>
  </xs:restriction>
</xs:simpleType>

```


In 5.3.2.1 Overview replace

A Media Presentation consists of one or more Periods. A Period is defined by a **Period** element in the **MPD** element.

The type of the Period, either a regular Period or an Early Available Period, as well as the *PeriodStart* time of a regular Period is determined as follows:

- If the attribute *@start* is present in the **Period**, then the Period is a regular Period and the *PeriodStart* is equal to the value of this attribute.
- If the *@start* attribute is absent, but the previous **Period** element contains a *@duration* attribute then this new Period is also a regular Period. The start time of the new Period *PeriodStart* is the sum of the start time of the previous Period *PeriodStart* and the value of the attribute *@duration* of the previous Period.
- If (i) *@start* attribute is absent, and (ii) the **Period** element is the first in the MPD, and (iii) the **MPD@type** is 'static', then the *PeriodStart* time shall be set to zero.
- If (i) *@start* attribute is absent, and (ii) the previous **Period** element does not contain a *@duration* attribute or the **Period** element is the first in the MPD, and (iii) the **MPD@type** is 'dynamic', then this Period is an Early Available Period (see below for details).

For any regular Period the following holds: *PeriodStart* reflects the actual time that should elapse after playing the media of all prior Periods in this Media Presentation relative to the *PeriodStart* time of the first Period in the Media Presentation. The Period extends until the *PeriodStart* of the next Period, or until the end of the Media Presentation in the case of the last Period. More specifically, the difference between the *PeriodStart* time of a Period and either the *PeriodStart* time of the following Period, if this is not the last Period, or the value of the **MPD@mediaPresentationDuration** if this is the last one, is the presentation duration in Media Presentation time of the media content represented by the Representations in this Period.

Early Available Periods may be used to advertise initialization of other non-media data before the media data itself is available. **Period** elements documenting early available Periods shall not occur before any **Period** element documenting a regular Period. For Early Available Periods, any resources that are announced in such a **Period** element shall be available. Such a **Period** element shall not contain URLs to Media Segments. The data contained in such a **Period** element does not represent a Period in the Media Presentation. Only when the *PeriodStart* time becomes known through an update of the MPD, such a **Period** element represents a regular Period. However, an update of the MPD may even remove a **Period** element representing an Early Available Period in later updates of the MPD as long as no *PeriodStart* time is associated with the Period.

with

A Media Presentation consists of one or more Periods. A Period is defined by a **Period** element in the **MPD** element.

Three types of Periods are defined:

- Regular Period,
- Early Available Period,
- Early Terminated Period.

The type of the Period, as well as the *PeriodStart* time of a regular Period or early terminated Period is determined as follows:

- If the attribute *@start* is present in the **Period**, then the Period is a regular Period or an early terminated Period and the *PeriodStart* is equal to the value of this attribute.
- If the *@start* attribute is absent, but the previous **Period** element contains a *@duration* attribute then this new Period is also a regular Period or an early terminated Period. The start time of the new Period *PeriodStart* is the sum of the start time of the previous Period *PeriodStart* and the value of the attribute *@duration* of the previous Period.

- If (i) @start attribute is absent, and (ii) the **Period** element is the first in the MPD, and (iii) the **MPD@type** is 'static', then the *PeriodStart* time shall be set to zero.
- If (i) @start attribute is absent, and (ii) the previous **Period** element does not contain a @duration attribute or the **Period** element is the first in the MPD, and (iii) the **MPD@type** is 'dynamic', then this Period is an Early Available Period (see below for details).
- If (i) @duration attribute is present, and (ii) the next **Period** element contains a @start attribute or the @minimumUpdatePeriod is present, then this Period is an Early Terminated Period (see below for details).

For any regular Period and early terminated Period the following holds: *PeriodStart* reflects the actual time that should elapse after playing the media of all prior Periods in this Media Presentation relative to the *PeriodStart* time of the first Period in the Media Presentation. The Period extends until the *PeriodStart* of the next Period, or until the end of the Media Presentation in the case of the last Period. For regular Periods, the difference between the *PeriodStart* time of a Period and either the *PeriodStart* time of the following Period, if this is not the last Period, or the value of the **MPD@mediaPresentationDuration** if this is the last one, is the presentation duration in Media Presentation time of the media content represented by the Representations in this Period. For Early Terminated Periods, the value of the **Period@duration** is the presentation duration in Media Presentation time of the media content represented by the Representations in this Period.

Early Available Periods may be used to advertise initialization of other non-media data before the media data itself is available. **Period** elements documenting early available Periods shall not occur before any **Period** element documenting a regular Period. For Early Available Periods, any resources that are announced in such a **Period** element shall be available. The data contained in such a **Period** element does not represent a Period in the Media Presentation. Only when the *PeriodStart* time becomes known through an update of the MPD, such a **Period** element represents a regular Period. However, an update of the MPD may even remove a **Period** element representing an Early Available Period in later updates of the MPD as long as no *PeriodStart* time is associated with the Period.

In 5.3.2.2 Semantics replace

Table 4 — Semantics of Period element

Element or Attribute Name	Use	Description
Period		specifies the information of a Period.
@xlink:href	O	specifies a reference to a remote element entity that is either empty or contains one or multiple top-level elements of type Period
@xlink:actuate	OD default: onRequest	specifies the processing instructions, which can be either "onLoad" or "onRequest". This attribute shall not be present if the @xlink:href attribute is not present.
@id	O	specifies an identifier for this Period. The identifier shall be unique within the scope of the Media Presentation. If the MPD@type is "dynamic", then this attribute shall be present and shall not change in case the MPD is updated. If not present, no identifier for the Period is provided.
@start	O	if present, specifies the <i>PeriodStart</i> time of the Period. The <i>PeriodStart</i> time is used as an anchor to determine the MPD start time of each Media Segment as well as to determine the presentation time of each access unit in the Media Presentation timeline. If not present, refer to the details in 5.3.2.1.
@duration	O	if present specifies the duration of the Period to determine the <i>PeriodStart</i> time of the next Period. If not present, refer to the details in 5.3.2.1.
@bitstreamSwitching	OD Default: false	When set to 'true', this is equivalent as if the AdaptationSet@bitstreamSwitching for each Adaptation Set contained in this Period is set to 'true'. In this case, the AdaptationSet@bitstreamSwitching attribute shall not be set to 'false' for any Adaptation Set in this Period.
BaseURL	0...N	specifies a base URL that can be used for reference resolution and alternative URL selection. For more details refer to the description in 5.6.
SegmentBase	0...1	specifies default Segment Base information. Information in this element is overridden by information in AdaptationSet.SegmentBase and Representation.SegmentBase , if present. For more details see 5.3.9.
SegmentList	0...1	specifies default Segment List information. Information in this element is overridden by information in AdaptationSet.SegmentList and Representation.SegmentList , if present. For more details see 5.3.9.
SegmentTemplate	0...1	specifies default Segment Template information. Information in this element is overridden by information in AdaptationSet.SegmentTemplate and Representation.SegmentTemplate , if present. For more details see 5.3.9.

Element or Attribute Name	Use	Description
AssetIdentifier	0...1	specifies that this Period belongs to a certain asset. For more details see 5.8.5.7.
EventStream	0...N	specifies an event stream. For more details see 5.10.2.
AdaptationSet	0...N	specifies an Adaptation Set. At least one Adaptation Set shall be present in each Period unless the value of the @duration attribute of the Period is set to zero. For more details see 5.3.3.
Subset	0...N	specifies a Subset. For more details see 5.3.8.
Legend: For attributes: M=Mandatory, O=Optional, OD=Optional with Default Value, CM=Conditionally Mandatory. For elements: <minOccurs>...<maxOccurs> (N=unbounded) Note that the conditions only holds without using xLink:href. If linking is used, then all attributes are "optional" and <minOccurs=0> Elements are bold ; attributes are non-bold and preceded with an @.		

with

Table 4 — Semantics of Period element

Element or Attribute Name	Use	Description
Period		specifies the information of a Period.
@xlink:href	O	specifies a reference to a remote element entity that is either empty or contains one or multiple top-level elements of type Period
@xlink:actuate	OD default: onRequest	specifies the processing instructions, which can be either "onLoad" or "onRequest". This attribute shall not be present if the @xlink:href attribute is not present.
@id	O	specifies an identifier for this Period. The identifier shall be unique within the scope of the Media Presentation. If the MPD@type is "dynamic", then this attribute shall be present and shall not change in case the MPD is updated. If not present, no identifier for the Period is provided.
@start	O	if present, specifies the <i>PeriodStart</i> time of the Period. The <i>PeriodStart</i> time is used as an anchor to determine the MPD start time of each Media Segment as well as to determine the presentation time of each access unit in the Media Presentation timeline. If not present, refer to the details in 5.3.2.1.
@duration	O	if present specifies the duration of the Period to determine the <i>PeriodStart</i> time of the next Period. If not present, refer to the details in 5.3.2.1.
@bitstreamSwitching	OD Default: false	When set to 'true', this is equivalent as if the AdaptationSet@bitstreamSwitching for each Adaptation Set contained in this Period is set to 'true'. In this case, the AdaptationSet@bitstreamSwitching attribute shall not be

Element or Attribute Name	Use	Description
		set to 'false' for any Adaptation Set in this Period.
BaseURL	0...N	specifies a base URL that can be used for reference resolution and alternative URL selection. For more details refer to the description in 5.6.
SegmentBase	0...1	specifies default Segment Base information. Information in this element is overridden by information in AdaptationSet.SegmentBase and Representation.SegmentBase , if present. For more details see 5.3.9.
SegmentList	0...1	specifies default Segment List information. Information in this element is overridden by information in AdaptationSet.SegmentList and Representation.SegmentList , if present. For more details see 5.3.9.
SegmentTemplate	0...1	specifies default Segment Template information. Information in this element is overridden by information in AdaptationSet.SegmentTemplate and Representation.SegmentTemplate , if present. For more details see 5.3.9.
AssetIdentifier	0...1	specifies that this Period belongs to a certain asset. For more details see 5.8.5.7.
EventStream	0...N	specifies an event stream. For more details see 5.10.2.
AdaptationSet	0...N	specifies an Adaptation Set. At least one Adaptation Set shall be present in each Period unless the value of the @duration attribute of the Period is set to zero. For more details see 5.3.3.
Subset	0...N	specifies a Subset. For more details see 5.3.8.
SupplementalProperty	0...N	specifies supplemental information about the containing element that may be used by the DASH client optimizing the processing. For details see 5.8.4.9..
Legend: For attributes: M=Mandatory, O=Optional, OD=Optional with Default Value, CM=Conditionally Mandatory. For elements: <minOccurs>...<maxOccurs> (N=unbounded) Note that the conditions only holds without using xlink:href. If linking is used, then all attributes are "optional" and <minOccurs=0> Elements are bold ; attributes are non-bold and preceded with an @.		

In 5.3.2.3 XML syntax replace

```

<!-- Period -->
<xs:complexType name="PeriodType">
  <xs:sequence>
    <xs:element name="BaseUrl" type="BaseUrlType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="SegmentBase" type="SegmentBaseType" minOccurs="0"/>
    <xs:element name="SegmentList" type="SegmentListType" minOccurs="0"/>
    <xs:element name="SegmentTemplate" type="SegmentTemplateType" minOccurs="0"/>
    <xs:element name="AssetIdentifier" type="DescriptorType" minOccurs="0"/>
    <xs:element name="EventStream" type="EventStreamType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="AdaptationSet" type="AdaptationSetType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="Subset" type="SubsetType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute ref="xlink:href"/>
  <xs:attribute ref="xlink:actuate" default="onRequest"/>
  <xs:attribute name="id" type="xs:string" />
  <xs:attribute name="start" type="xs:duration"/>
  <xs:attribute name="duration" type="xs:duration"/>
  <xs:attribute name="bitstreamSwitching" type="xs:boolean" default="false"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

```

with

```

<!-- Period -->
<xs:complexType name="PeriodType">
  <xs:sequence>
    <xs:element name="BaseUrl" type="BaseUrlType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="SegmentBase" type="SegmentBaseType" minOccurs="0"/>
    <xs:element name="SegmentList" type="SegmentListType" minOccurs="0"/>
    <xs:element name="SegmentTemplate" type="SegmentTemplateType" minOccurs="0"/>
    <xs:element name="AssetIdentifier" type="DescriptorType" minOccurs="0"/>
    <xs:element name="EventStream" type="EventStreamType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="AdaptationSet" type="AdaptationSetType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="Subset" type="SubsetType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="SupplementalProperty" type="DescriptorType" minOccurs="0"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute ref="xlink:href"/>
  <xs:attribute ref="xlink:actuate" default="onRequest"/>
  <xs:attribute name="id" type="xs:string" />
  <xs:attribute name="start" type="xs:duration"/>
  <xs:attribute name="duration" type="xs:duration"/>
  <xs:attribute name="bitstreamSwitching" type="xs:boolean" default="false"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

```

In 5.3.3.1 Overview replace

Each Period consists of one or more Adaptation Sets. An Adaptation Set is described by an **AdaptationSet** element. **AdaptationSet** elements are contained in a **Period** element.

An Adaptation Set contains alternate Representations, i.e. only one Representation within an Adaptation Set is expected to be presented at a time. All Representations contained in one Adaptation Set represent the same media content components and therefore contain media streams that are considered to be perceptually equivalent. The Adaptation Set and the contained Representations shall be prepared and contain sufficient information such that seamless switching across different Representations in one Adaptation Set is enabled. If an Adaptation Set is expected to be consumed by DASH clients with restrictions in terms of switching, then the Media Presentation author should provide sufficient means to enable seamless switching under these restrictions.

with

5.3.3.1 Overview

Each Period consists of one or more Adaptation Sets. An Adaptation Set is described by an **AdaptationSet** element. **AdaptationSet** elements are contained in a **Period** element.

An Adaptation Set contains alternate Representations, i.e. only one Representation within an Adaptation Set is expected to be presented at a time. All Representations contained in one Adaptation Set represent the same media content components and therefore contain media streams that are considered to be perceptually equivalent. The Adaptation Set and the contained Representations shall be prepared and contain sufficient information such that seamless switching (as defined in 4.5.1) across different Representations in one Adaptation Set is enabled. If an Adaptation Set is expected to be consumed by DASH clients with restrictions in terms of switching, then the Media Presentation author should provide sufficient means to enable seamless switching under these restrictions.

In 5.3.5.1 Overview add after the 1st Note

NOTE In the case a dependent Representation X depends on at least two complementary Representations Y_i , that are also dependent Representations depending, directly or indirectly, on the same complementary Representation Z, then the concatenation as defined above results in a sequence conforming to the media format if the Segments of Representation Z are concatenated only the first time encountered following the order in the @dependencyId attributes starting from X. If following recursively the dependencies indicated in the @dependencyId of Representations, certain Representations may be encountered more than once. In this case the corresponding Segments or Subsegments are expected to only be concatenated once, namely when first encountered.

In 5.3.7.3 XML syntax replace

```

<!-- Representation base (common attributes and elements) -->
<xs:complexType name="RepresentationBaseType">
  <xs:sequence>
    <xs:element name="FramePacking" type="DescriptorType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="AudioChannelConfiguration" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="ContentProtection" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="EssentialProperty" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="SupplementalProperty" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="InbandEventStream" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="profiles" type="xs:string"/>
  <xs:attribute name="width" type="xs:unsignedInt"/>
  <xs:attribute name="height" type="xs:unsignedInt"/>
  <xs:attribute name="sar" type="RatioType"/>
  <xs:attribute name="frameRate" type="FrameRateType"/>
  <xs:attribute name="audioSamplingRate" type="xs:string"/>
  <xs:attribute name="mimeType" type="xs:string"/>
  <xs:attribute name="segmentProfiles" type="xs:string"/>
  <xs:attribute name="codecs" type="xs:string"/>
  <xs:attribute name="maximumSAPPeriod" type="xs:double"/>
  <xs:attribute name="startWithSAP" type="SAPType"/>
  <xs:attribute name="maxPlayoutRate" type="xs:double"/>
  <xs:attribute name="codingDependency" type="xs:boolean"/>
  <xs:attribute name="scanType" type="VideoScanType"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- Stream Access Point type enumeration -->
<xs:simpleType name="SAPType">
  <xs:restriction base="xs:unsignedInt">
    <xs:minInclusive value="0"/>
    <xs:maxInclusive value="6"/>
  </xs:restriction>
</xs:simpleType>

<!-- Video Scan type enumeration -->
<xs:simpleType name="VideoScanType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="progressive"/>
    <xs:enumeration value="interlaced"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>

```

with

```

<!-- Representation base (common attributes and elements) -->
<xs:complexType name="RepresentationBaseType">
  <xs:sequence>
    <xs:element name="FramePacking" type="DescriptorType" minOccurs="0" maxOccurs="unbounded"/>
    <xs:element name="AudioChannelConfiguration" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="ContentProtection" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="EssentialProperty" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="SupplementalProperty" type="DescriptorType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:element name="InbandEventStream" type="EventStreamType" minOccurs="0"
maxOccurs="unbounded"/>
    <xs:any namespace="##other" processContents="lax" minOccurs="0" maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="profiles" type="xs:string"/>
  <xs:attribute name="width" type="xs:unsignedInt"/>
  <xs:attribute name="height" type="xs:unsignedInt"/>
  <xs:attribute name="sar" type="RatioType"/>
  <xs:attribute name="frameRate" type="FrameRateType"/>
  <xs:attribute name="audioSamplingRate" type="xs:string"/>
  <xs:attribute name="mimeType" type="xs:string"/>
  <xs:attribute name="segmentProfiles" type="xs:string"/>
  <xs:attribute name="codecs" type="xs:string"/>
  <xs:attribute name="maximumSAPPeriod" type="xs:double"/>
  <xs:attribute name="startWithSAP" type="SAPType"/>
  <xs:attribute name="maxPlayoutRate" type="xs:double"/>
  <xs:attribute name="codingDependency" type="xs:boolean"/>
  <xs:attribute name="scanType" type="VideoScanType"/>
  <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

<!-- Stream Access Point type enumeration -->
<xs:simpleType name="SAPType">
  <xs:restriction base="xs:unsignedInt">
    <xs:minInclusive value="0"/>
    <xs:maxInclusive value="6"/>
  </xs:restriction>
</xs:simpleType>

<!-- Video Scan type enumeration -->
<xs:simpleType name="VideoScanType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="progressive"/>
    <xs:enumeration value="interlaced"/>
    <xs:enumeration value="unknown"/>
  </xs:restriction>
</xs:simpleType>

```