INTERNATIONAL STANDARD

ISO/IEC 23009-1

Second edition
2014-05-15
AMENDMENT 2
2015-07-01

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

Part 1:
Media presentation description and segment formats

AMENDMENT 2: Spatial relationship description, generalized URL parameters and other extensions

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

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

AMENDEMENT 2: Description des relations spatiales, paramètres URL généralisés et autres extensions

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Amendment 2 to ISO/IEC 23009-1:2015 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 29, *Coding of Audio, Picture, Multimedia and Hypermedia Information*.

Introduction

This amendment to ISO/IEC 23009-1 adds the ability for MPD authors to express:

- Spatial Relationship Description between Spatial Objects in the MPD;

- Flexible parameter insertions in URLs used to query Media Segments;

- Additional Role @values;

- Association between Representations

Information Technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Spatial Relationship Description, Generalized URL parameters and other extensions

In clause 2 "Normative references", add the following reference, in alphabetical order:

HTML 4.01 Specification, W3C Recommendation, 24 December1999

In clause 3.1 "Terms and definitions", add the following definitions in alphabetical order:

3.1.X
associated Representation
Representation which provides supplemental or descriptive information for at least one other Representation

3.1.X
Spatial Object
a media content component corresponding to a region in a coordinate system associated to this media content component

In clause 3.2 "Symbols and abbreviated terms", add the following term:

SRD   Spatial Relationship Description

In clause 4.7 "Schemes", Table 2, add the following identifiers:

Scheme Identifier	Clause in this part of ISO/IEC 23009	Informative description
`urn:mpeg:dash:srd:2014`	H.1	Scheme identifier for Spatial Relationship Description
`urn:mpeg:dash:urlparam:2014`	I.1	Scheme identifier for indicating usage of the flexible insertion of URL query parameters
`urn:mpeg:dash:audio-receiver-mix:2014`	5.8.5.7	Scheme identifier for receiver mix

In clause 5.3.5.1 "Overview", add :

"Associated Representations are described by a **Representation** element that contains an `@associationId` attribute and optionally an `@associationType` attribute. Associated Representations are Representations that provide information on their relationships with other Representations. As opposed to complementary Representations, the segments of an associated Representation may be optional for decoding

and/or presentation of the Representations identified by @associationId. They can be considered as supplementary or descriptive information, the type of the association being specified by the @associationType attribute"

In clause 5.3.5.2 "Representation Semantics", Table 7 – Semantics of Representation element, add:

@associationId	O	specifies all Representations the Representation is associated with in the decoding and/or presentation process as a whitespace-separated list of values of Representation@id attributes.
@associationType	O	specifies, as a whitespace separated list of values, the kind of association for each Representation the Representation has been associated with through the @associationId attribute. Values taken by this attribute are 4 character codes for track reference types registered in MP4 registration authority. This attribute shall not be present when @associationId is not present. When present, this attribute must have as many values as the number of identifiers declared in the @associationId attribute

NOTE - @associationId and @associationType attributes can only be used between Representations that are not in the same Adaptation Sets.

In clause 5.8.5.5 "DASH Role scheme", Table 22, add the following rows:

description	Textual or audio media component containing a textual description (intended for audio synthesis) or an audio description describing a visual component
sign	Visual media component representing a sign-language interpretation of an audio component.
metadata	Media component containing information intended to be processed by application specific elements.
enhanced-audio-intelligibility	Audio component with improved intelligibility of the dialogue

In clause 5.8.5.5 "DASH Role scheme", in Table 22, add the following note:

> 4) Role descriptors with values such as "subtitle", "caption", "description", "sign" or "metadata" may be used to enable assignment of a "kind" value in HTML 5 applications for tracks exposed from a DASH MPD.

Add the following to the end of clause 5.8.5

5.8.5.7 Audio Receiver Mix

This clause defines a scheme for use in **EssentialProperty** or **SupplementaryProperty** to indicate that two audio Adaptation Sets need to be mixed by the media engine prior to playback.

The `@schemeIdUri` attribute identifying the scheme is `urn:mpeg:dash:audio-receiver-mix:2014.`

The `@value` attribute shall contain the value of the **AdaptationSet**`@id` attribute from an Adaptation Set with content type audio which the current Adaptation Set needs to be mixed with in order to provide complete audio experience.

An example of receiver mix, is the case where a single audio Adaptation Set provides music and effects – i.e., complete experience without dialogues, and one or more Adaptation Sets provide dialogues in different languages. In this case the dialogue Adaptation Sets will depend on music and effects Adaptation Set.

The mixing requirement is unidirectional – i.e., requirement of mixing Representation A with Representation B when A is selected does not imply that mixing the two is required if B is selected.

Add the following annexes:

Annex H
(normative)

Spatial Relationship Description

H.1 Spatial Relationship Description (SRD) scheme

The SRD scheme allows Media Presentation Description authors to express spatial relationships between Spatial Objects. A Spatial Object is represented by either an Adaptation Set or a Sub-Representation. As an example, a spatial relationship may express that a video represents a spatial part of another full-frame video (e.g. a region of interest, or a tile).

The **SupplementalProperty** and/or **EssentialProperty** descriptors with @schemeIdUri equal to "urn:mpeg:dash:srd:2014" are used to provide spatial relationship information associated to the containing Spatial Object. SRD shall be contained exclusively in these two MPD elements (**AdaptationSet** and **SubRepresentation**).

To preserve the compatibility with legacy clients, MPD shall use **SupplementalProperty** and **EssentialProperty** in such a way that at least one Representation can be interpreted by legacy clients after discarding the element containing **EssentialProperty.**

NOTE – Sub-Representation level SRDs may be used to represent Spatial Objects in one Representation such as HEVC tiling streams. In that case, SRD descriptors may be present at Adaptation Set as well as Sub-Representation levels.

H.2 SRD @value syntax

The @value of the **SupplementalProperty** or **EssentialProperty** elements using the SRD scheme is a comma separated list of values for SRD parameters.

The source_id parameter provides a unique identifier, within the Period, for the source of the content. It implicitly defines a coordinate system associated to this source. This coordinate system uses an arbitrary origin (0; 0); the x-axis is oriented from left to right and the y-axis from top to bottom. All SRD sharing the same source_id value have the same origin and axes orientations. Spatial relationships for Spatial Objects using SRD with different source_id values are undefined.

For a given source_id value, a reference space is defined, corresponding to the rectangular region encompassing the entire source content, whose top-left corner is at the origin of the coordinate system. The total_width and total_height values in a SRD provide the size of this reference space expressed in arbitrary units.

NOTE – There may be no Spatial Object in the MPD that covers the entire source of the content, e.g. when the entire source content is represented by two separate videos.

The object_x and object_y parameters (respectively object_width and object_height) express 2D positions (respectively 2D sizes) of the associated Spatial Object in the coordinate system associated to the source. The values of the object_x, object_y, object_width, and object_height parameters are relative to the values of the total_width and total_height parameters, as defined above. Positions (object_x, object_y) and sizes (object_width, object_height) of SRDs sharing the same source_id value may be compared after taking into account the size of the reference space, i.e. after the object_x and object_width values are divided by the total_width value and the object_y and object_height values divided by the total_height value of their respective descriptors.

NOTE – Different `total_width` and `total_height` values may be used in different descriptors to provide positions and sizes information in different units for the same reference space.

MPD authors can express, using the `spatial_set_id` parameter, that some Spatial Objects, within a given `source_id`, have a particular spatial relationship. For instance, an MPD author may group all Adaptation Sets corresponding to tiles at a same resolution level. This way, the `spatial_set_id` parameter may be used by the DASH client to quickly select spatially related Spatial Objects. When multiple full-frame videos which consist of one or more Spatial Objects with the same `total_width` and `total_height` value, different values of `spatial_set_id` can be used to distinguish the groups of full-frame video.

NOTE – ISO/IEC 23009-3 gives concrete examples showing how to use the `spatial_set_id`.

When `@value` is not present, the SRD does not express any spatial relationship information at all and can be ignored.

Table H.1 — `EssentialProperty`@value and/or `SupplementalProperty`@value attributes for the SRD scheme

`EssentialProperty`@value or `SupplementalProperty`@value parameter	Use	Description
`source_id`	M	non-negative integer in decimal representation providing the identifier for the source of the content
`object_x`	M	non-negative integer in decimal representation expressing the horizontal position of the top-left corner of the Spatial Object in arbitrary units
`object_y`	M	non-negative integer in decimal representation expressing the vertical position of the top-left corner of the Spatial Object in arbitrary units
`object_width`	M	non-negative integer in decimal representation expressing the width of the Spatial Object in arbitrary units
`object_height`	M	non-negative integer in decimal representation expressing the height of the Spatial Object in arbitrary units
`total_width`	O	optional non-negative integer in decimal representation expressing the width of the reference space in arbitrary units. At each Period and for a given `source_id` value, the following rules apply: - There shall be at least one descriptor providing a value for the `total_width` parameter. - If two or more descriptors provide different `total_width` values, all other descriptors shall explicitly provide the value of `total_width`. - If the `total_width` value is provided in only one descriptor, all other descriptors are assumed to use that `total_width` value. - The value of `total_width` shall be such, for each descriptor using this value of `total_width`, the sum of `object_x` and `object_width` is smaller or equal to `total_width`. When the value `total_width` is present, the value `total_height` shall be present.
`total_height`	O	optional non-negative integer in decimal representation expressing the height of the

		reference space in arbitrary units. At each Period and for a given `source_id` value, the following rules apply: - There shall be at least one descriptor providing a value for the `total_height` parameter. - If two or more descriptors provide different `total_height` values, all other descriptors shall explicitly provide the value of `total_height`. - If the `total_height` value is provided in only one descriptor, all other descriptors are assumed to use that `total_height` value. - The value of `total_height` shall be such that, for each descriptor using this value of `total_height`, the sum of `object_y` and `object_height` is smaller or equal to `total_height`. When the value `total_height` is present, the value `total_width` shall be present.
`spatial_set_id`	O	optional non-negative integer in decimal representation providing an identifier for a group of Spatial Objects. When not present, the Spatial Object associated to this descriptor does not belong to any spatial set and no spatial set information is given. When the value of `spatial_set_id` is present, the value of `total_width` and `total_height` shall be present.

Legend:

M=Mandatory, O=Optional

H.3 Examples

H.3.1 Zoomed video

This subclause provides a simple example of a static presentation with 2 videos, one video representing a zoomed part of the other video. The Media Presentation complies with the ISO Base media file format On Demand profile as defined in 8.3.

```xml
<?xml version="1.0" encoding="UTF-8"?>
<MPD
  xmlns="urn:mpeg:dash:schema:mpd:2011"
  type="static"
  mediaPresentationDuration="PT10S"
  minBufferTime="PT1S"
  profiles="urn:mpeg:dash:profile:isoff-on-demand:2011">

  <ProgramInformation>
    <Title>Example of a DASH Media Presentation Description using Spatial Relationship Description
to indicate that a video is a zoomed part of another</Title>
  </ProgramInformation>

  <Period>
    <!-- Panorama Video -->
    <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
      <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
      <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,0,0,3,3,3,3"/>
      <Representation mimeType="video/mp4" codecs="avc1.42c033" width="1920" height="1080"
bandwidth="1055223" startWithSAP="1">
        <BaseURL> panorama video.mp4</BaseURL>
        <SegmentBase indexRangeExact="true" indexRange="839-990"/>
      </Representation>
    </AdaptationSet>
```

```
        <!-- Zoomed Video -->
    <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="supplementary"/>
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,1,1,1,1,3,3"/>
        <Representation mimeType="video/mp4" codecs="avc1.42c028" width="1920" height="1080"
bandwidth="769458" startWithSAP="1">
            <BaseURL> zoomed_video.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="838-989"/>
        </Representation>
    </AdaptationSet>
  </Period>
</MPD>
```

H.3.2   Tiled video

This subclause provides a simple example of a static presentation of a video available in different resolutions and of tiles of that video also available in different resolutions. The Media Presentation complies with the ISO Base media file format On Demand profile as defined in 8.3.

```
<?xml version="1.0" encoding="UTF-8"?>
<MPD
  xmlns="urn:mpeg:dash:schema:mpd:2011"
  type="static"
  mediaPresentationDuration="PT10S"
  minBufferTime="PT1S"
  profiles="urn:mpeg:dash:profile:isoff-on-demand:2011">

  <ProgramInformation>
    <Title>Example of a DASH Media Presentation Description using Spatial Relationship Description
to indicate tiles of a video</Title>
  </ProgramInformation>

  <Period>
    <!-- Main Video -->
    <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="main"/>
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,0,0,2,2,2,2"/>
        <Representation mimeType="video/mp4" codecs="avc1.42c01e" width="640" height="360"
bandwidth="226597" startWithSAP="1">
            <BaseURL> full video small.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="837-988"/>
        </Representation>
        <Representation mimeType="video/mp4" codecs="avc1.42c01f" width="1280" height="720"
bandwidth="553833" startWithSAP="1">
            <BaseURL> full_video_hd.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="838-989"/>
        </Representation>
        <Representation mimeType="video/mp4" codecs="avc1.42c033" width="3840" height="2160"
bandwidth="1055223" startWithSAP="1">
            <BaseURL> full video_4k.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="839-990"/>
        </Representation>
    </AdaptationSet>
    <!-- Tile 1 -->
    <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <Role schemeIdUri="urn:mpeg:dash:role:2011" value="supplementary"/>
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,0,0,1,1,2,2"/>
        <Representation mimeType="video/mp4" codecs="avc1.42c00d" width="640" height="360"
bandwidth="218284" startWithSAP="1">
            <BaseURL> tile1 video small.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="837-988"/>
        </Representation>
        <Representation mimeType="video/mp4" codecs="avc1.42c01f" width="1280" height="720"
bandwidth="525609" startWithSAP="1">
            <BaseURL> tile1 video hd.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="838-989"/>
        </Representation>
        <Representation mimeType="video/mp4" codecs="avc1.42c028" width="1920" height="1080"
bandwidth="769514" startWithSAP="1">
            <BaseURL> tile1 video fullhd.mp4</BaseURL>
            <SegmentBase indexRangeExact="true" indexRange="839-990"/>
```

```
        </Representation>
      </AdaptationSet>
      <!-- Tile 2 -->
      <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,1,0,1,1,2,2"/>
…
      </AdaptationSet>
      <!-- Tile 3 -->
      <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,1,1,1,1,2,2"/>
…
      </AdaptationSet>
      <!-- Tile 4 -->
      <AdaptationSet segmentAlignment="true" subsegmentAlignment="true" subsegmentStartsWithSAP="1">
        <SupplementalProperty schemeIdUri="urn:mpeg:dash:srd:2014" value="0,0,1,1,1,2,2"/>
…
      </AdaptationSet>
    </Period>
</MPD>
```

Annex I
(normative)

Flexible Insertion of URL Parameters

I.1 Introduction

This Annex describes how it is possible to configure URL parameters of Media Segment URLs, in a similar fashion to the URL template mechanism.

The core specification of DASH already allows building Media Segment URLs containing static URL parameters. This Annex aims at providing more flexibility and dynamicity in the way URL parameters are inserted.

The mechanism described in this Annex allows:

- To "Inherit" from MPD URL parameters when the MPD is delivered over HTTP, i.e. to extract one or more URL parameters from the MPD URL and to use them in Media Segment URLs

- To define URL parameters that are dynamically ("just in time") computed, thanks to XLink mechanism

- To express to the DASH client that some specific feedback is expected from the client side, provided through Media Segment URLs

- To select a specific parameter out of several available parameters

I.2 Syntax

Flexible insertion of URL parameters is signaled through the use of **EssentialProperty** or **SupplementalProperty** descriptors, with @schemeIdUri equal to "urn:mpeg:dash:urlparam:2014".

A child element up:**UrlQueryInfo** shall be present in these descriptors, defined within the "urn:mpeg:dash:schema:urlparam:2014" namespace. The namespace prefix should be "up:".

As defined by this specification, each of these descriptors may be present at the MPD, Adaptation Set or at the Representation level. Only **SupplementalProperty** descriptor may be present at the Period level. When insertion of URL parameters is required for a Period, **EssentialProperty** descriptors shall be inserted in all Adaptation Sets of that Period. At most one descriptor shall be present at each level.

When the insertion of URL parameters requires scheme-dependent computation, one or several additional **EssentialProperty** or **SupplementalProperty** descriptors shall be present. These descriptors shall carry an appropriate @schemeIdUri attribute referencing the scheme to be used, and provide sufficient information to appropriately compute the required URL parameters (see I.2.3.2). Support of these schemes is not in the scope of this specification.

I.2.1   URL Query Information

I.2.1.1   Overview

The **UrlQueryInfo** element describes how to build a URL query string, which is used in the Media Segments URLs building process.

This query string can come from one of the three sources below:

- The URL of the MPD when the @useMPDUrlQuery is set

- The @queryString attribute when present

- The @queryString attribute, after any XLink resolution in case @xlink:href is present

The @queryTemplate attribute describes which URL parameters contained in the query string are used in the Media Segment URL building process, as well as the order of these parameters.

I.2.1.2   Semantics

Table I.1 — Semantics of `UrlQueryInfo` element

Element or Attribute Name	Use	Description
UrlQueryInfo		provides URL query string information
@queryTemplate	O (string)	provides URL parameters template information This string shall contain one or more $<ParamIdentifier>$ template identifiers, as listed in Table I.2. These template identifiers are to be replaced to build a query string (see I.2.3). If $<ParamIdentifier>$ is not in Table I.2, it will be replaced with an empty string. If the template has an opening $ without a matching closing $, the result is undefined, and the client will act as if it did not understand the **EssentialProperty** or **SupplementalProperty** scheme. When selection of URL parameters is enabled through the use of $query:param$ template identifiers, URL parameters shall be defined as name-value pairs separated by "&", as defined by W3C HTML 4.01 Specification (section on Forms#Form submission).
@useMPDUrlQuery	O (bool) default: false	indicates that the URL parameters of the MPD URL are used in the construction of Media Segment URLs. This flag may only be present when the MPD is delivered over HTTP, and defaults to "false" when the MPD is not delivered over HTTP. NOTE – simple parameter signaling may be used ("a=X&b=Y"), as well as scheme-dependent signaling ("a=$urn:XYZ$&b=$urn:ABC$"). When scheme-dependent signaling is used, the scheme shall be inserted between two enclosing $ characters.

		See I.2.3.2 for further details.
@queryString	O (string)	provides a query string to be used in the construction of Media Segment URLs. NOTE – simple parameter signaling may be used ("`a=X&b=Y`"), as well as scheme-dependent signaling ("`a=$urn:XYZ$&b=$urn:ABC$`"). When scheme-dependent signaling is used, the scheme shall be inserted between two enclosing $ characters.See I.2.3.2 for further details.
@xlink:href	O	specifies a reference to a remote **UrlQueryInfo** element
@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.
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 @.		

Table I.2 - Identifiers for URL parameter templates

$<ParamIdentifier>$	Substitution parameter
$$	Is an escape sequence, i.e. "$$" is replaced with a single "$"
$*querypart*$	This identifier is substituted with the query part of the computed query string (referred to as *initialQueryString* in section I.2.3) The query part starts after the "?" sign and lasts until the "#" sign
$*query:param*$	This identifier is substituted with the value of the param parameter if this parameter is present in the query part of the computed query string (referred to as *initialQueryString* in section I.2.3). If param appears more than once in the query string, the last value will be used. If param is not present, the empty string will be used. When this parameter selection is used, URL parameters shall be inserted in the query part of the URL as name-value pairs separated by the "&" symbol, according to W3C HTML 4.01 Specification (section on Forms#Form submission)

I.2.1.3 XML Syntax

```xml
<!-- Url query information -->
<xs:complexType name="UrlQueryInfoType">
  <xs:attribute name="queryTemplate" type="xs:string"/>
  <xs:attribute name="useMPDUrlQuery" type="xs:boolean"/>
  <xs:attribute name="queryString" type="xs:string"/>
  <xs:attribute name="xlink:href"/>
  <xs:attribute name="xlink:actuate" default="onRequest"/>
</xs:complexType>
```

I.2.2 Modified template-based segment URL construction, according to `UrlQueryInfo` element

When signalized through an appropriate descriptor, containing a `UrlQueryInfo` element, the following Media Segment URL building process is performed.

The process is defined in three steps:

1. Initial query string (referred to as *initialQueryString*) is derived. This process is described in I.2.3.1.
2. Final query string (referred to as *finalQueryString*) is computed, according to `@queryTemplate` and *initialQueryString*. This process is described in I.2.3.2
3. Final query string (*finalQueryString*) is processed to build Media Segment URLs. This process is described in I.2.3.3

I.2.2.1 Computation of an initial query string (*initialQueryString*)

The initial query string *initialQueryString* is constructed by concatenating the query strings, if present and available, coming from the MPD URL (if `@useMPDUrlQuery` is set to "true") and `@queryString` (after an appropriate XLink resolution, if `@xlink:href` is present).

When two or more occurrences of URL query descriptors exist within an MPD, the final URL query string used at the inner-most Representation level is a concatenation of the corresponding URL query strings of the occurrences in their orders of appearance in the MPD hierarchy. The query coming from the MPD URL is appended first.

When multiple strings are appended together, an "&" symbol shall be inserted at the start of the second and following strings to be appended.

I.2.2.2 Computation of a final query string (*finalQueryString*)

A final query string *finalQueryString* is then computed by substituting URL parameters templates present in `@queryTemplate` by their values provided in *initialQueryString*, according to Table I.2.

Simple parameter signaling may be used (`@queryString` = "a=X&b=Y"), as well as scheme-dependent signaling (`@queryString` = "a=$urn:XYZ$&b=$urn:ABC$"). In the latter case, the client needs to be aware of the provided schemes, and has to compute appropriate values for them. Further, in this case, additional **EssentialProperty** or **SupplementalProperty** descriptors, at the same level as the query descriptor, shall be present to reflect that scheme-dependent signaling is used and required to be supported by the client. These descriptors shall have the `@schemeIdUri` attribute set to the same value as used in the