**3GPP SA4#114-e *S4-210765***

**19-28 May 2021**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **Pseudo CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **26.804** | **CR** | **<CR#>** | **rev** | **-** | **Current version:** | **0.1.1** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | [FS\_5GMS-EXT] Candidate solution for Content Preparation format | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Tencent | | | | | | | | | |
| ***Source to TSG:*** | SA4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | FS\_5GMS-EXT | | | | |  | ***Date:*** | | | 2021-01-25 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **B** |  | | | | | ***Release:*** | | | Rel-17 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) Rel-12 (Release 12)* *Rel-13 (Release 13) Rel-14 (Release 14) Rel-15 (Release 15) Rel-16 (Release 16)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | The study item description identifies the key topic “Content Preparation”. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Adding some candidate solutions for simple CMAF content preparation | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Key topic not addressed | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
| ***56*** | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

**===== CHANGE 1 =====**

# References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

[31] ETSI TS 103 799: "Publicly Available Specification (PAS); DASH-IF Content Protection Information Exchange Format".

[32] ISO/IEC JTC1/SC29/WG11/N19062 23090-8 FDIS: "MPEG-I: Network-based Media Processing — Network-Based Media Processing Specification".

**===== CHANGE 2 =====**

### 5.2.8 Candidate Solutions

Editor’s Note: Provide candidate solutions (including call flows) for each of the identified issues.

#### 5.2.8.1 Content Preparation Template requirements

##### 5.2.8.1.1 Unencrypted single CMAF track to single unencrypted CMAF switching set

The Content Preparation Template must define the following parameters:

1. The address/location of the input CMAF segments.

2. Output CMAF switching set configuration:

a. *Output manifest parameters:* The characteristics that are typically described in a manifest such as an MPEG‑DASH MPD [11]. While these parameters are per track, they can be described once if one or more of them are common across tracks. Examples:

i. *Packaging parameters:* Container profile, codec/profile/level, bit rate, container profiles, maximum SAP period, start with SAP.

ii. *Video parameters:* Width, height, sample aspect ratio, frame rate.

iii. *Audio parameters:* sampling rate, audio channel configuration.

b. *Internal encoding parameters:* The parameters used for encoding each track that are not presented in the output manifest, such as bit rate control, motion search area, and algorithm, CBR/VBR/Capped VBR encoding, use of specific quality metrics.

i. Common encoding parameters (usually common in a codec/profile/level).

ii. Vendor-defined (implementation-specific) parameters.

#### 5.2.8.2 Content Preparation Template candidates

##### 5.2.8.2.1 CMAF input format candidate 1: DASH MPD manifest

In this case, the characteristics of the input CMAF track can be described by a DASH MPD manifest in order to correctly initialise the media transcoder in the Content Preparation subfunction of the 5GMS AS.The manifest may include information such as codec/profile/level, as well as general characteristics of the media, such as maximum segment duration, video width, height and frame rate, the existence of any SEI messages and other metadata, the existence of any events schemes, as well as the location of each segment in the input CMAF track. In the case of uplink media streaming, where HTTP pull is used to egest the transcoded segments via M2u, the manifest can even optionally define the locations of CMAF segments on the 5GMSu AS.

##### 5.2.8.2.2 CMAF input format candidate 2: A new document format

A new document format can be used for describing the input CMAF segments. The advantage of such a solution is when the same format is used for describing the output CMAF formats.

##### 5.2.8.2.3 CMAF output format candidate 3: Extended manifest format

In this approach, a standard manifest format is used for describing the output manifest parameters, but it is extended to also carry the internal encoding parameters (both common and vendor-defined).

An example is to use MPEG‑DASH MPD format and add descriptors to the adaptation set and/or representations for the internal encoding parameters. Two classes of descriptors can be added:

1. The common encoding descriptor per codec, carrying common parameters.

2. Vendor-specific descriptors, carrying vendor-defined parameters.

Since the MPD essential and supplemental descriptor syntax allows different scheme URIs to be defined, both of the above features can be expressed using the same descriptor data type.

##### 5.2.8.2.4 CMAF output format candidate 4: Manifest with supplementary encoding parameters document

In this approach, a standard manifest format is used to describe the output manifest parameters and a separate document is used to describe the internal encoding parameters.

An example is the following elements:

1. MPEG‑DASH MPD format for output manifest parameters.

2. A JSON document containing an array of objects that each include a DASH Representation identifier referencing a Representation in the MPD. Each object also includes encoding parameters for the Representation. The common encoding parameters may be extended with vendor-specific parameters inside a child object tagged with a URI that uniquely identifies the vendor.

For example, the MPD in Listing 5.2.8.2.4‑1 below has two representations: R1 and R2. (For simplicity the adaptation set is not shown.) The internal encoding parameters document in Listing 5.2.8.4.‑2 has a JSON array containing two elements. Each element refers to one DASH Representation in the MPD by its id attribute value. Each array element includes the encoding parameters for the corresponding Representation. The second element includes a vendor-specific encoding parameter that is signaled using the vendor urn. The other items in that object is defined by the vendor.

Listing 5.2.8.2.4‑1: MPEG-DASH MPD

|  |
| --- |
| <MPD>  <Representation @id='R1' …>  <Representation @id='R2' …>  </MPD> |

Listing 5.2.8.2.4‑2: Supplementary encoding parameters document in JSON format

|  |
| --- |
| [{ Rid = 'R1'; search-window =64; frame-rate= 30; …};  { Rid = 'R2'; search-window =64; frame-rate= 60; {vendor='urn:companyA:encoding:CAE', mode='segment-based', context='sports' …}]; |

##### 5.2.8.2.5 CMAF output format candidate 5: A document defining both the output manifest and encoding parameters

In this approach, a new document format is defined to describe both the manifest output parameters and the internal encoding parameters.

An example of such a solution would be a JSON document containing an array of objects that each include the following information:

1. Output manifest parameters.

2. Common internal encoding parameters.

3. Vendor-specific internal encoding parameters tagged with the vendor’s identifier (such as a URI).

Another alternative would be to use the DASH Industry Forum’s Content Protection Information Exchange Format (CPIX) [31] and possibly extend it to carry additional parameters that are needed.

#### 5.2.8.3 Combining the Content Preparation Template candidate solutions

Since both input, outputs and encoding information need to be provided in the Content Preparation Template, the following solutions are possible for the overall template by combining the candidate solutions described in clause 5.2.8.2 above:

1. Single MPD:

a. One adaptation set with one input representation describing the input according to 5.2.8.2.1

b. One adaptation set with multiple input representation describing the output tracks according to 5.2.8.2.3

2. A document consisting of two MPDs, with possibly a supplementary document:

a. One MPD describing the input according to 5.2.8.2.1 and

b. One of the following:

i. One MPD describing the outputs and encoding format according to 5.2.8.2.3, or

ii. One MPD describing the outputs and one document describing the encoding parameters according to 5.2.8.2.4.

3. Single JSON document:

a. One item describing the input representation according to 5.2.8.2.2, and

b. An array of objects according to 5.2.8.2.5, each of which describes:

i. One output.

ii. The encoding parameter for that output.

##### 5.2.8.1.8 Combined CMAF input and output formats candidate: NBMP Workflow Description Document

The NBMP Workflow Description Document (WDD) [32] can describe the entire workflow. In this use case, the WDD describes the input format, as well as the array of tasks/function instances, each of which defines the CMAF output track as well as the encoding parameters for that track, as is shown in the following figure:

5GMSd AS

Content Preparation

Task 1:

Transcoder 1

Task 2:

Transcoder 2

Task n-1:

Transcoder n-1

**M2d**

Content Hosting

NBMP WDD and/or function template

Figure 5.2.8-1: Using NBMP WDD to describe CMAF content preparation

The NBMP WDD in this case describes the input CMAF as the input of workflow, and the function, configurations, and output of each task. Since many features of the NBMP specification are not used in this specific workflow, the WDD features can be profiled to a suitable subset of descriptors defined by NBMP in [32].

The NBMP specification allows so-called *function templates* to be defined. One way to simplify the support for NBMP by 5GMSd AS would therefore be to define a function template for each Content Preparation Use Case. For example, the 5GMS CMAF Content Preparation function template could define (among other things):

1. Input CMAF media profile using explicit description: MPEG‑DASH MPD or HLS m3u8 playlist.

2. The push/pull protocols for ingesting CMAF content at M2d.

3. The required CMAF output formats.

4. The transocoder’s common and vendor-specific configuration parameters

5. Multiple codec output.

6. Reporting, monitoring and notification parameters for each transcoding function.

Another advantage of the NBMP WDD format is that it can be used to describe other Content Preparation use cases and therefore one single format may be able to address several applications.

In this approach:

1. A function reference template is created for the CMAF content preparation
2. A profile of NBMP spec is defined to simplify the NBMP Workflow requirements.