**3GPP TSG SA WG4 #114e *S4-210803***

**E-meeting, 19th – 28th May 2021**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **Pseudo CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **26.804** | **CR** | **<CR#>** | **rev** | **-** | **Current version:** | **1.2.8** |  |
|  | | | | | | | | |
| *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] Key Topic Scalable distribution of unicast Live Services | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Qualcomm Incorporated | | | | | | | | | |
| ***Source to TSG:*** | SA4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | FS\_5GMS-EXT | | | | |  | ***Date:*** | | | 2021-05-11 |
|  |  | | | |  | |  | | |  |
| ***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:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***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 =====**

## 5.11 TV-grade mass distribution of unicast Live Services

### 5.11.1 Description

#### 5.11.1.1 General

Live TV services of different scale (professional, user-generated, session-based, etc.) are increasingly distributed over broadband and mobile networks. Live TV services are characterized by:

- scalability (in terms of concurrent users),

- consistent quality,

- high bandwidth requirements, and

- target latency constraints.

#### 5.11.1.2 Scalability

Editor’s Note: Awaiting contributions.

Consistent support of the distribution of such services to a different scale of users and in a concurrent fashion is a prime concern. 5G Media Streaming is expected to support such service distribution and end-to-end optimizations. Improvements and optimizations on the architectural level and stage 3 are expected to be studied.

#### 5.11.1.3 Consistent quality

Editor’s Note: Awaiting contributions.

#### 5.11.1.4 High bandwidth requirements

Editor’s Note: Awaiting contributions.

#### 5.11.1.5 Target latency constraints

Based on a report developed jointly between DVB and DASH-IF on Low-Latency DASH [9], this clause defines details on how to support consistent latency in DASH for linear TV services. In [9], several definitions had been introduced, repeated here for consistency.

*- End-to-End Latency (EEL)*: The latency for an action that is captured by the camera until its visibility on the remote screen.

*- Encoder-Display Latency (EDL)*: The latency of the linear playout output (which typically serves as input to distribution encoder(s)) to the screen.

*- Packager-Display Latency*: The latency after the output of the distribution encoder to the screen.

*- CDN latency*: The delay caused by the CDN delivery from CDN input to CDN output.

*- Live Edge Start-up Delay (LSD)*: The time between a user action (service access or service join) and the time until the first media sample of the service is perceived by the user when joining at the live edge. Typically also the channel change time.

*- Seek Start-up Delay (SSD)*: The time between a user action (service access or service join) and the time until the first media sample of the service is perceived by the user when seeking to a time shift buffer.

Those two categories, latency and delay are subject to be controllable by the service provider for a consistent service offering. In the remainder, primarily the Encoder-Display Latency (EDL) and the Live Edge Start-up Delay are considered, but for some use cases also the End-to-End Latency (EEL) may be relevant. Figure 5.11.1‑1 provides a schematic overview of the different latencies.



Figure 5.11.1‑1: Different latencies and delays relevant for low-latency distribution

The Low Latency DASH scenario is a variant of the Live Services recommended approach focused on ensuring that the Encoder-Display Latency of the DASH Media Presentation is comparable to the latency when distributing over terrestrial, cable or satellite broadcast. Latency in broadcast is not a unique universal value, as it is influenced by many factors such as the duration of the broadcast encoding pipeline, the latency of the transport channel which can slightly differ per type (satellite, cable, IPTV or, DTT...), or the artificial delays introduced by local content moderation regulations. However, most of the measurements converge on a 3 - 10 seconds latency between the moment where the source signal is acquired for encoding and the moment when it's played back on the TVs, i.e the EDL. Start-up delay requirements are typically in the range of 1-2 seconds. For details refer to [9].

Low-latency mode are supported to minimize the architectural impacts on existing workflows. Figure 5.11.1‑2 provides a basic flow of information for operating a low-latency DASH service as defined in DASH-IF’s Low-latency Modes for DASH [10]. The DASH packager gets information on the general description of the service as well as the encoder configuration. The encoder produces CMAF chunks and fragments. The chunks are mapped by the MPD packager onto Segments and provided to the network in incremental fashion using HTTP/1.1 chunked transfer.



Figure 5.11.1-2 Basic operation flow Low-Latency DASH

HTTP chunked transfer coding needs to be supported up from the ingest into the packager up to the CDN edge, whereas the last mile delivery is expected happen using HTTP chunked transfer coding or HTTP in regular mode. If HTTP chunked transfer coding is supported by the DASH player, it basically means that a media segment carrying the latest moment of the program (also known as the "live edge time" as defined in clause 4 of this document) could be consumed on the player while it's still being produced by the encoder and the packager.

In case chunked segments are used, clients may want to access partially available Segments for example for fast random access, see ISO/IEC 23009-1 [11]. However, requesting available byte ranges of a partially available Segment, i.e., Segments still being produced, is not consistently supported in CDNs, but solutions are provided in RFC8673 [X6]. This functionality may also be needed to support common segment handling for low-latency DASH and low-latency HLS.

Key aspects for low-latency live distribution include:

*-* Consistent support for chunked transfer from ingest to client.

*-* Support for partially access of non-complete resources.

*-* End-to-end optimizations to support the latency requirements.

### 5.11.2 Deployment Architectures

#### 5.11.2.1 Distribution of low-latency media streams

A deployment architecture suitable for low-latency CMAF streaming is shown in Figure 5.11.2.1-1. In this case:

1. A live stream is ingested into a live encoder.

2. The encoded stream is packaged into CMAF chunks.

3. The packaged CMAF chunks are uploaded to an origin server using chunked transfer encoding input.

4. Segments are then available for retrieval by a CDN on demand and moved through the CDN all the way to the client.

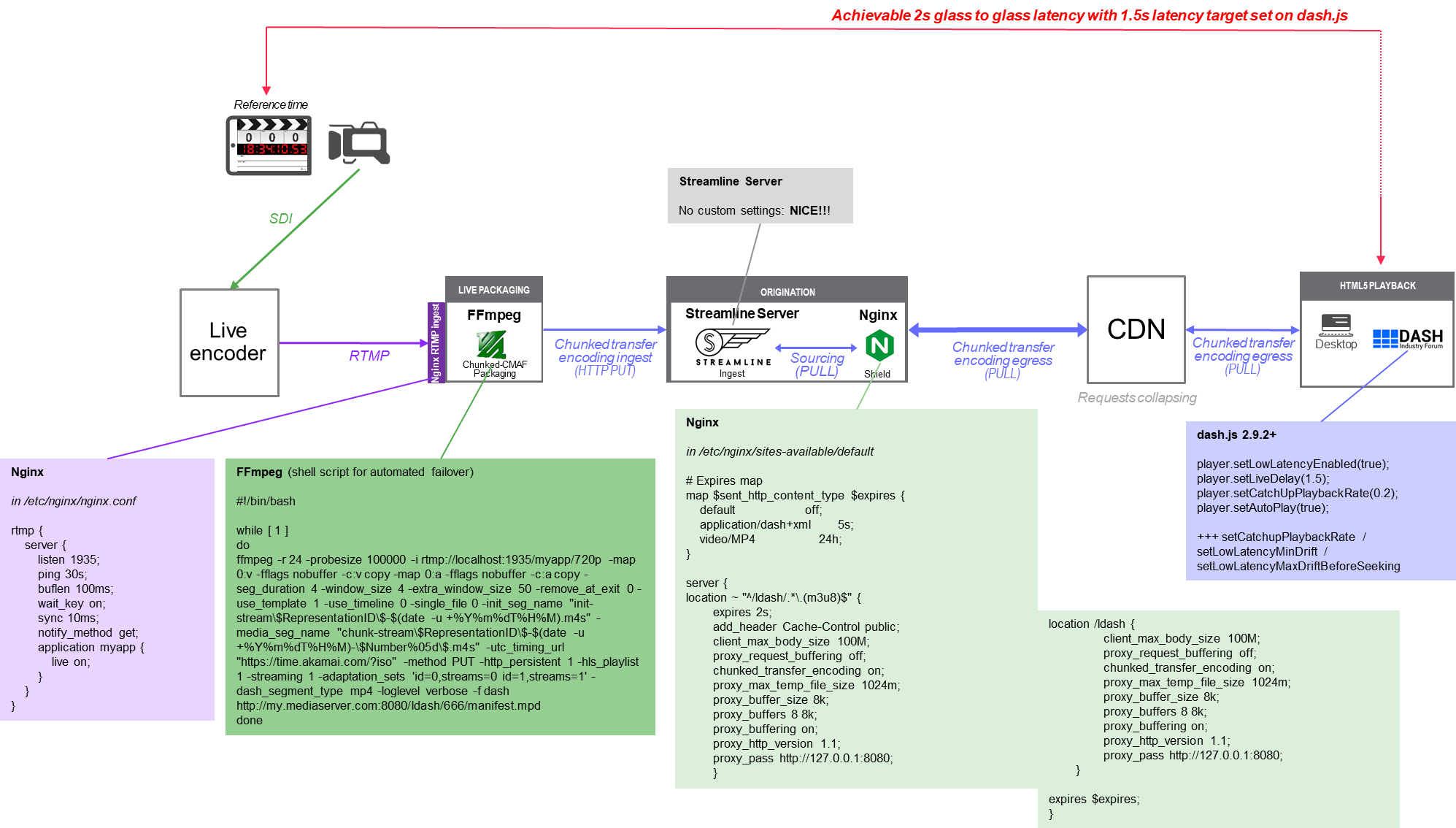


Figure 5.11.3-1 Deployment architecture for low-latency CMAF streaming

### 5.11.3 Collaboration Scenarios

#### 5.11.3.1 General

The following collaboration scenarios may be considered:

1. Live content is provided to the MNO as an (uncompressed or mezzanine-compressed) contribution feed, and the MNO does the encoding and packaging for distribution.
2. Encoded and packaged live content is uploaded to the MNO, and the content provider does the encoding and packaging. The MNO may produce an MPD for its distribution. In this case, the content is augmented with producer reference times.
3. The origin server is external to the MNO network and content is pulled through the 5GMS AS on demand by the clients.

#### 5.11.3.5 Distribution of low-latency media streams

For all of the collaboration scenarions described in clause 5.11.3.1 above, the content provider and the service provider agree on:

* The MNO may monitor if the end-to-end latency target is maintained. This may for example be done by proper reporting.
* The desired latency from glass-to-glass is met for example to be 3 seconds.
* That the content is provided in low-latency, but also for consumption in time shift mode.
* That the content can be accessed before the whole segment is uploaded.

### 5.11.4 Mapping to 5G Media Streaming and High-Level Call Flows

Editor’s Note: Map the key topics to basic functions and develop high-level call flows.

#### 5.11.4.1 Collaboration 1: MNO provides encoding and packaging

Architecture:

- Relates to content preparation

#### 5.11.4.2 Collaboration 2: MNO provides DASH distribution

Architecture:

- Relates to content preparation

#### 5.11.4.3 Collaboration 3: MNO acts as CDN

### 5.11.5 Potential open issues

Editor’s Note: Identify the issues that need to be solved.

### 5.11.6 Candidate Solutions

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