|  |  |
| --- | --- |
| 3GPP TR 26.804 V0.5.0 (2021-08) | |
| Technical Report | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Study on 5G media streaming extensions  (Release 17) | |
|  | |
|  |  |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2021, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 6

1 Scope 8

2 References 8

3 Definitions of terms, symbols and abbreviations 11

3.1 Terms 11

3.2 Symbols 11

3.3 Abbreviations 11

5 Key Topics 12

5.1 Introduction 12

5.2 Content Preparation 12

5.2.1 Overview 12

5.2.2 Gap Analysis of 26.512 12

5.2.3 Use-cases 13

5.2.3.1 Basic CMAF/DASH/HLS multi-rate live streaming of user-generated content 13

5.2.4 Collaboration Scenarios 14

5.2.4.1 Content preparation before downlink streaming 14

5.2.4.2 Content preparation after uplink ingest streaming 15

5.2.4.3 Content preparation between uplink ingest and downlink streaming 16

5.2.5 Deployment Architectures 16

5.2.6 Mapping to 5G Media Streaming and High-Level Call Flows 17

5.2.6.1 Call flow for content preparation before downlink streaming 17

5.2.6.2 Call flow for content preparation after uplink streaming 20

5.2.6.3 Baseline call flow for content processing between uplink streaming and downlink streaming 22

5.2.7 Potential open issues 23

5.2.7.1 Open issues in collaboration scenario 1: Content preparation before downlink streaming 23

5.2.7.2 Open issues in collaboration scenario 2: content preparation after uplink streaming 24

5.2.7.3 Open issues in collaboration scenario 3: content preparation between uplink and downlink 25

5.2.8 Candidate Solutions 25

5.2.8.1 Content Preparation Template requirements 25

5.2.8.1.1 Unencrypted single CMAF track to single unencrypted CMAF switching set 25

5.2.8.2 Content Preparation Template candidates 26

5.2.8.2.1 CMAF input format candidate 1: DASH MPD manifest 26

5.2.8.2.2 CMAF input format candidate 2: A new document format 26

5.2.8.2.3 CMAF output format candidate 3: Extended manifest format 26

5.2.8.2.4 CMAF output format candidate 4: Manifest with supplementary encoding parameters document 26

5.2.8.2.5 CMAF output format candidate 5: A document defining both the output manifest and encoding parameters 27

5.2.8.3 Combining the Content Preparation Template candidate solutions 27

5.2.8.4 Combined CMAF input and output formats candidate: NBMP Workflow Description Document 28

5.2.8.5 Address translation for complex pull requests 29

5.3 Traffic Identification 30

5.3.1 Description 30

5.3.2 Collaboration Scenarios 32

5.3.3 Deployment Architectures 33

5.3.4 Mapping to 5G Media Streaming and High-Level Call Flows 34

5.3.4.1 General 34

- Domain Name: The Internet domain name of an application server. This method of traffic detection is not described further in the present document.5.3.4.2 Usage of 5-tuples for Traffic Identification 34

5.3.4.3 Usage of ToS Traffic Class for Traffic Identification 36

5.3.4.4 Usage of Packet Flow Descriptions for Traffic Identification 37

5.3.5 Potential open issues 38

5.3.6 Candidate Solutions 38

5.4 Additional/new transport protocols 39

5.4.1 Description 39

5.4.1.1 General 39

5.4.1.2 Performance Considerations for HTTP/3 using IETF QUIC over 5G Networks 40

5.4.1.3 Performance Considerations for IETF QUIC over 5G networks 41

5.4.1.4 Management Considerations for HTTP/3 and QUIC in 5G networks 41

5.4.2 Collaboration Scenarios 41

5.4.3 Deployment Architectures 41

5.4.4 Mapping to 5G Media Streaming and High-Level Call Flows 41

5.4.5 Potential open issues 41

5.4.6 Candidate Solutions 41

5.5 Uplink media streaming 42

5.5.1 Description 42

5.5.1.1 Overview 42

5.5.1.2 Gap analysis of TS 26.501 42

5.5.1.3 Gap analysis of TS 26.512 42

5.5.1.4 Gap analysis between TS 26.238 (FLUS) and TS 26.512 (5G Media Streaming) 43

5.5.2 Collaboration Scenarios 43

5.5.2.1 Overview 43

5.5.2.2 Collaboration Scenario 1 44

5.5.2.3 Collaboration Scenario 2 44

5.5.2.4 Collaboration Scenario 3 45

5.5.2.5 Collaboration Scenario 4 45

5.5.2.6 Collaboration Scenario 5 46

5.5.2.7 Collaboration Scenario 6 46

5.5.3 Deployment Architectures 47

5.5.4 Mapping to 5G Media Streaming and High-Level Call Flows 47

5.5.4.1 Collaboration scenario 1 call flow 47

5.5.4.2 Collaboration scenario 2 call flow 48

5.5.4.3 Collaboration scenario 3 call flow 50

5.5.4.4 Collaboration scenario 4 call flow 52

5.5.4.5 Collaboration scenario 5 call flow 54

5.5.4.6 Collaboration scenario 6 call flow 55

5.5.5 Potential open issues 55

5.5.5.1 Potential open issues in 5G Media Streaming stage 3 55

5.5.5.2 Potential open issues compared with FLUS 56

5.5.5.2.1 General 56

5.5.5.2.2 Discussion 1 56

5.5.5.2.3 Discussion 2 57

5.5.5.2.4 Discussion 3 57

5.5.6 Candidate Solutions 57

5.5.6.1 Content egest protocols 57

5.5.6.2 Content Publishing Configuration API 58

5.5.6.3 Content Publishing Configuration Template 58

5.5.6.4 Uplink entry point 59

5.6 Background traffic 60

5.6.1 Description 60

5.6.2 Collaboration Scenarios 61

5.6.3 Deployment Architectures 61

5.6.4 Mapping to 5G Media Streaming and High-Level Call Flows 61

5.6.5 Potential open issues 62

5.6.6 Candidate Solutions 62

5.7 Content-Aware Streaming 62

5.7.1 Description 62

5.7.2 Collaboration Scenarios 64

5.7.3 Deployment Architectures 65

5.7.4 Mapping to 5G Media Streaming and High-Level Call Flows 65

5.7.5 Potential open issues 65

5.7.6 Candidate Solutions 66

5.8 Network Event usage 66

5.8.1 Description 66

5.8.1.1 Events exposed by 5GMS AF 66

5.8.1.2 Events consumed by 5GMS AF 66

5.8.2 Collaboration Scenarios 67

5.8.3 Deployment Architectures 67

5.8.4 Mapping to 5G Media Streaming and High-Level Call Flows 68

5.8.5 Potential open issues 69

5.8.6 Candidate Solution 69

5.8.6.1 UE Data Collection via Direct and Indirect Methods 69

5.8.6.2 AF Collection of CDN Access Logs 71

5.8.6.3 Candidate media-related information for Event Exposure 71

5.9 Per-application-authorization 72

5.9.1 Description 72

5.9.2 Collaboration Scenarios 73

5.9.2.1 Collaboration A: UE hosting multiple Applications 73

5.9.2.2 Collaboration B: Applications with multiple subscription levels 74

5.9.3 Deployment Architectures 74

5.9.4 Mapping to 5G Media Streaming and High-Level Call Flows 74

5.9.5 Potential open issues 74

5.9.6 Candidate Solutions 74

5.10 Support for encrypted and high-value content 75

5.10.1 Description 75

5.10.2 Collaboration Scenarios 75

5.10.3 Deployment Architectures 76

5.10.4 Mapping to 5G Media Streaming and High-Level Call Flows 76

5.10.5 Potential open issues 76

5.10.6 Candidate Solutions 76

5.11 TV-grade mass distribution of unicast Live Services 76

5.11.1 Description 76

5.11.1.1 General 76

5.11.1.2 Scalability 76

5.11.1.3 Consistent quality 76

5.11.1.4 High bandwidth requirements 76

5.11.1.5 Target latency constraints 77

5.11.2 Deployment Architectures 79

5.11.2.1 Distribution of low-latency media streams 79

5.11.3 Collaboration Scenarios 79

5.11.3.1 General 79

5.11.3.2 Distribution of low-latency media streams 80

5.11.4 Mapping to 5G Media Streaming and High-Level Call Flows 80

5.11.4.1 Collaboration 1: MNO provides encoding and packaging 80

5.11.4.2 Collaboration 2: MNO provides DASH distribution 80

5.11.4.3 Collaboration 3: MNO acts as CDN 80

5.11.5 Potential open issues 80

5.11.6 Candidate Solutions 80

Annex A – Media Streaming Protocols 81

A.1 Status and usage of Web Protocols 81

A.1.1 M4d protocol usage 81

A.1.2 Results of HTTP protocol version usage study 81

Annex <X> (informative): Change history 82

# Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# 1 Scope

The present document …

This Technical Report identifies and evaluates a set of potential improvements and extensions, referred to as key topics. The key topics are

- Content Preparation

- Traffic Identification

- Additional / New transport protocols

- Uplink media streaming

- Background traffic

- Content Aware Streaming

- Network Event usage

- Per-application-authorization

- Support for encrypted and high-value content

- Scalable distribution of unicast Live Services

For each of the above key topics, the following objectives are identified:

1. Document the above key topics in more detail, in particular how they relate to the 5GMS Architecture and protocols.

2. Study collaboration scenarios between the 5G System and Application Provider for each of the key topics.

3. Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

4. Map the key topics to basic functions and develop high-level call flows.

5. Identify the issues that need to be solved.

6. Provide candidate solutions (including call flows) for each of the identified issues.

7. Coordinate work with other 3GPP groups e.g. SA2, SA3, SA5, and others as needed.

8. Coordinate work with external organizations such as DASH-IF, CTA WAVE, ISO/IEC JTC29 WG3 (MPEG Systems), or IETF, as needed.

9. Identify gaps and recommend potential normative work for stage-2 call flows and possibly stage-3.

# 2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] Akamai Blog, "A QUICk Introduction to HTTP/3", April 2020, <https://developer.akamai.com/blog/2020/04/14/quick-introduction-http3>

[3] Fielding, R., Nottingham, M., and J. Reschke, "HTTP/1.1", Work in Progress, Internet-Draft, draft-ietf-httpbis-messaging-13, 14 December 2020, http://www.ietf.org/internet-drafts/draft-ietf-httpbis-messaging-13.txt

[4] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, May 2015, https://www.rfc-editor.org/info/rfc7540

[5] draft-ietf-quic-http-34, "Hypertext Transfer Protocol Version 3 (HTTP/3)", February 2021

[6] D. Bhat, A. Rizk, and M. Zink, "Not so QUIC: A Performance Study of DASH over QUIC," NOSSDAV'17: Proceedings of the 27th Workshop on Network and Operating Systems Support for Digital Audio and VideoJune 2017 Pages 13–18 https://doi.org/10.1145/3083165.3083175

[7] AWS, "Achieving Great Video Quality Without Breaking the Bank", Streaming Media June 2019, [[https://pages.awscloud.com/rs/112-TZM-766/images/GEN elemental-wp-achieving-great-video-quality-without-breaking-the-bank.pdf](https://pages.awscloud.com/rs/112-TZM-766/images/GEN%20elemental-wp-achieving-great-video-quality-without-breaking-the-bank.pdf)](https://pages.awscloud.com/rs/112-TZM-766/images/GEN%20elemental-wp-achieving-great-video-quality-without-breaking-the-bank.pdf)

[8] Netflix, "Optimized shot-based encodes: Now Streaming!", Netflix Blog, May 2018, https://netflixtechblog.com/optimized-shot-based-encodes-now-streaming-4b9464204830

[9] DASH-IF/DVB Report on Low-Latency Live Service with DASH, July 2017, available here: <https://dash-industry-forum.github.io/docs/Report%20on%20Low%20Latency%20DASH.pdf>

[10] DASH-IF IOP Guidelines v5, Low-latency Modes for DASH, available here: <https://dash-industry-forum.github.io/docs/CR-Low-Latency-Live-r8.pdf>

[11] ISO/IEC 23009-1, "Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats".

[12] IETF RFC 8673, "HTTP Random Access and Live Content".

[13] 3GPP TR 26.939: "Guidelines on the Framework for Live Uplink Streaming (FLUS)".

[14] 3GPP TS 26.238: "Uplink Streaming".

[15] 3GPP TS 26.501: "5G Media Streaming (5GMS); General description and architecture".

[16] 3GPP TS 26.512: "5G Media Streaming (5GMS); Protocols".

[17] ISO/IEC 13818-1:2019: "Information technology — Generic coding of moving pictures and associated audio information — Part 1: Systems".

[18] SCTE 35 2020: "Digital Program Insertion Cueing Message", <https://www.scte.org/pdf-redirect/?url=https://scte-cms-resource-storage.s3.amazonaws.com/SCTE-35-2020_notice-1609861286512.pdf>

[19] ISO/IEC 23000-19:2020 Information technology — Multimedia application format (MPEG-A) —Part 19: Common media application format (CMAF) for segmented media

[20] ISO/IEC 23009-1:2019/DAMD1 Information technology — Dynamic adaptive streaming over HTTP (DASH) — Part 1: Media presentation description and segment formats — Amendment 1: CMAF support, events processing model and other extensions [21] VSF TR-06-01, RIST Simple Profile, <https://www.videoservicesforum.org/download/technical_recommendations/VSF_TR-06-1_2018_10_17.pdf>

[22] VSF TR-06-02: "RIST Main Profile", <https://www.videoservicesforum.org/download/technical_recommendations/VSF_TR-06-2_2020_03_24.pdf>

[23] 3GPP TS 23.501: "System architecture for the 5G System (5GS)".

[24] 3GPP TS 23.502: "Procedures for the 5G System (5GS)".

[25] 3GPP TS 29.517: "5G System; Application Function Event Exposure Service; Stage 3".

[26] 3GPP TS 29.244: "Interface between the Control Plane and the User Plane nodes; Stage 3".

[27] IETF RFC 6733: "Diameter Base Protocol".

[28] 3GPP TS 29.514: "5G System; Policy and Charging Control over Rx reference point; Stage 3".

[29] IETF RFC 7657: "Differentiated Services (Diffserv) and Real-Time Communication", November 1995.

[30] IETF RFC 3168: "The Addition of Explicit Congestion Notification (ECN) to IP", September 2001.

[31] C. Krasic, M. Bishop, and A. Frindell, Ed., draft-ietf-quic-qpack-21, "QPACK: Header Compression for HTTP/3", Work in Progress, Internet-Draft, 2 February 2021

[32] J. Iyengar, Ed. and M. Thomson, Ed., draft-ietf-quic-transport-34, "QUIC: A UDP-Based Multiplexed and Secure Transport", Work in Progress, Internet-Draft, 15 January 2021

[33] M. Thomson, Ed. and S. Turner, Ed., draft-ietf-quic-tls-34, "Using TLS to Secure QUIC", Work in Progress, Internet-Draft, 15 January 2021

[34] J. Iyengar, Ed. and I. Swett, Ed., draft-ietf-quic-recovery-34, "QUIC Loss Detection and Congestion Control", Work in Progress, Internet-Draft, 15 January 2021

[35] IETF RFC 5681, "TCP Congestion Control".

[36] M. Kuehlewind and B. Trammell, draft-ietf-quic-manageability-11, “Manageability of the QUIC Transport Protocol”, Work in Progress, Internet-Draft, 21 April 2021

[37] N. Cardwell et. al. “BBR Updates: Internal Deployment, Code, Draft Plans”, 9 March 2021, https://datatracker.ietf.org/meeting/110/materials/slides-110-iccrg-bbr-updates-00.pdf

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

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

[40] 3GPP TS 26.247: "Transparent end-to-end Packet-switched Streaming Service (PSS); Progressive Download and Dynamic Adaptive Streaming over HTTP (3GP-DASH)".[41] 3GPP TS 23.503: "Policy and charging control framework for the 5G System (5GS); Stage 2".

[42] 3GPP TS 29.514: "5G System; Policy Authorization Service; Stage 3".

[43] 3GPP TS 29.522: "5G System; Network Exposure Function Northbound APIs; Stage 3".

[44] 3GPP TS 29.122: "T8 reference point for Northbound APIs".

[45] 3GPP TS 29.512: "5G System; Session Management Policy Control Service; Stage 3".

[46] 3GPP TS 26.803: "5G Media Streaming (5GMS); Architecture extensions".

[47] 3GPP TS 23.558: "Architecture for enabling Edge Applications (EA)".

[48] 3GPP TS 23.288: "Architecture enhancements for 5G System (5GS) to support network data analytics services".

[49] Tdoc S4-210723: "Generic architecture for data collection and reporting", submission from BBC, Dolby Laboratories Inc., LM Ericsson and Qualcomm Incorporated to SA4#114-e, May 19-28, 2021.

[50] Tdoc S2-2103267: "Extension of Naf\_EventExposure for observed service experience data collection from UEs", CR from InterDigital to SA2#144e, Apr 12-16, 2021.

[51] 3GPP TS 26.114: "IP Multimedia Subsystem (IMS); Multimedia telephony; Media handling and interaction".

[52] Tdoc S2-2104496: "Extension of Naf\_EventExposure for observed service experience data collection from UEs", CR from Qualcomm Incorporated to SA2#145e, May 17-28, 2021.

[53] 3GPP TS 26.118: "Virtual Reality (VR) profiles for streaming applications".

[54] 3GPP TS 26.346: "Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs".

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

For the purposes of the present document, the terms given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

Definition format (Normal)

**<defined term>:** <definition>.

**example:** text used to clarify abstract rules by applying them literally.

## 3.2 Symbols

For the purposes of the present document, the following symbols apply:

Symbol format (EW)

<symbol> <Explanation>

## 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

CDN Content Delivery Network

FAR Forward Action Rule

MAR Multi-Access Rule

PDR Packet Detection Rule

PFCP Packet Forwarding Control Protocol

QER QoS Enforcement Rule

PFD Packet Flow Description

SDF Service Data Flow

URL Uniform Resource Locator

URR Usage Reporting Rule

# 5 Key Topics

## 5.1 Introduction

## 5.2 Content Preparation

### 5.2.1 Overview

[TS 26.501 [15] includes the high-level concept of content preparation and/or modification within the 5G Media Streaming System. However, Content Preparation has not yet been defined in detail in TS 26.512 [16]. The extent of content preparation support (including. use- cases, functionalities and features, and resulting formats) needed in the existing 5G Media Streaming Architecture, and how these functions can be realized in with current 5GMS architecture are subject of this study.]

Content preparation may be applied in the following scenarios:

1. In uplink streaming: on an uplinked stream, providing the resulting content to Application Provider.
2. In downlink streaming: on an input stream from an Application Provider, providing the resulting content for distribution.
3. Between uplink and downlink streaming: on an uplink stream, providing the resulting content for distribution.

Two aspects are of relevance for the discussion on Content Preparation:

* Uplink ingest content formats and protocols
* Content preparation instructions

On uplink ingest formats, a survey of existing formats is needed to understand the available options. Examples include

1. MPEG-2 Transport Stream [17] with SCTE metadata [18].
2. CMAF content [19] with timed metadata tracks.
3. CMAF content [19] with a manifest such as the MPD [20].
4. Reliable Internet Stream Transport (RIST) is an open source, open specification transport protocol designed for reliable transmission of video over lossy networks (including the internet) with low latency and high quality. It is currently under development under the Video Services Forum's "RIST Activity Group." To date, two open specifications have been produced:
   1. [TR-06-1 (RIST Simple Profile)](https://www.videoservicesforum.org/download/technical_recommendations/VSF_TR-06-1_2018_10_17.pdf) [21], first released in 2018, defines the basic ARQ (Automatic Repeat Query) technology used to recover lost packets. It has been updated in 2020 to include an optional "RTT Echo" message that streamlines the process of configuring packet buffers.
   2. [TR-06-2 (RIST Main Profile)](https://www.videoservicesforum.org/download/technical_recommendations/VSF_TR-06-2_2020_03_24.pdf) [22], released in 2020, defines additional functions required for commercial applications, including stream encryption, sender/receiver authentication, in-band data tunneling, and bandwidth optimization.
5. Secure Reliable Transport (SRT) is an open-source video transport protocol maintained by the SRT Alliance.

### 5.2.2 Gap Analysis of 26.512

TS 26.512 [16] defines a very limited set of features for content preparation:

1. The location for content preparation template provisioning.

2. CRUD Operation for content preparation template.

The current specification neither defines the uplink ingest formats nor the content preparation instructions.

### 5.2.3 Use-cases

#### 5.2.3.1 Basic CMAF/DASH/HLS multi-rate live streaming of user-generated content

|  |
| --- |
| **Description**  Kim is subscribed to an Application for live streaming of captured videos from her everyday life. Based on the previous number and diversity of Kim’s usual audience (e.g. close friends), the Application/Application Provider has an “audience codecs-rates” profile which represents the typical number of streams based on Kim’s previous streaming sessions and provides the corresponding Content Preparation Template to the MNO as Kim starts the session live stream session.  While Kim is streaming a single stream through her device’s uplink, the MNO processes the received content based on the Content Preparation Template and provides multi-rate tracks for distribution.  If new users join Kim’s streaming session which cannot be supported with the current codecs-rates, the Application/Application Provider may update the Content Preparation Template in the session, and the MNO updates the content preparation process accordingly. |
| **Categorization**  **Delivery:** Live Streaming  **Device:** Any device connected to the cell network |
| **Preconditions**   1. On the device:    1. A 3GPP supported encoder is installed.    2. UE is 5GMS capable.    3. UE’s Application is installed which supports 5GMS. 2. On the network:    1. The AS may or may not have resources for running a requested content preparation with the update to the Content Preparation Template and responds accordingly. |
| **Nominal Cost Analysis**  The cost of service increases linearly with the number of uplink ingest streams.  The cost of service increases less than linearly with the number of download streaming clients because the encoding and caching requirement are common to a large number of viewers. |
| **Potential Standardization Status and Needs**  The Content Preparation Template is expected to define uplink ingest format(s) as well as the following instructions for content preparation:  1. Input characteristics.  2, Outputs characteristics.  3. The media processes and/or functionalities applied.  4. Mechanism for updating the content preparation template.  Content preparation for CMAF streaming when the uplink ingest format in CMAF and output is CMAF content with DASH and HLS manifests for live and on-demand services. In this case, the Content Preparation Template should include the following:   1. Uplink ingest characteristics such as:    1. Video characteristics    2. Audio characteristics    3. Format of the subtitle track(s)    4. Metadata tracks and presence of media track events    5. Ingest protocols 2. Instructions such as:    1. CMAF/DASH publishing profile    2. Configuration of Switching sets:       1. Ladder of bit rate/quality       2. The encoder configuration for each track       3. Nominal and maximum segment duration       4. Frequency and characteristics of random access and switching points       5. Initialization segment characteristics       6. Content protection scheme and encryption mode       7. Metadata and annotation such as roles, languages, ratings, and accessibility       8. Metadata tracks associated with the switching set or individual tracks       9. Media track events to be included in each track       10. Chunk encoding for low latency streaming    3. Period structure, splicing opportunities, and conditions    4. 5GMSd AS URL exposed at M4    5. Segments Addressing modes    6. Trick mode and thumbnail navigation tracks    7. Service description    8. Content splicing/ad opportunity signaling    9. Subtitle generation from the uplink ingest audio track(s)    10. Multi-language support (when audio tracks for two or more languages are ingested).    11. Manifest format, annotations, and metadata |

### 5.2.4 Collaboration Scenarios

### 5.2.4.1 Content preparation before downlink streaming

In this collaboration, the 5GMSd Application Provider requests content preparation for its stream before distribution. Figure 5.2.4.1-1 shows such a scenario.

UE

5GMSd

Client

5GMSd

Aware Application

5GMSd AS

External DN

5GMSd

Application Provider

M1d

M2d

Trusted DN

M4d

5GMSd AF

M5d

PCF

M8d

N5

Figure 5.2.4.1-1: Content preparation before downlink streaming

In this case:

1. The Application Provider defines the required content preparation and requests the 5GMSd AF to create the process using the Content Preparation Template (CPT) through M1d.

2. The 5GMSd AF configures the 5GMSd AS according to the media transformation(s) specified in the Content Preparation Template (M3d) and responds to the Application Provider (M1d).

3. The media is streamed from Application Provider to 5GMSd AS (M2d).

4. The content is transformed by 5GMSd AS according to the Content Preparation Template and provided for distributions (M4d).

5. The Application Provider may update, retrieve, or destroy the Content Preparation Template using M1d.

Editor’s Note: How does DistributionConfiguration relate to CPT? Does the CPT define the initial/generic formats (such as HLS and DASH) and DistributionConfiguration defines a subset of it for distribution?

### 5.2.4.2 Content preparation after uplink ingest streaming

In this collaboration, the 5GMSu Application Provider or 5GMSu Aware Application requests content preparation for the uplink ingest stream and delivery to 5GMSu Application Provider. Figure 5.2.4.2-1 shows such a scenario.

UE

5GMSu

Client

5GMSu

Aware Application

5GMSu AS

External DN

5GMSu

Application Provider

M1u

M2u

Trusted DN

M4u

5GMSu AF

M5u

PCF

M8u

N5

Figure 5.2.4.2-1: Content preparation after uplink ingest streaming

In this case:

1. The Application Provider defines the required content preparation and requests the 5GMSu AF to create the process using the Content Preparation Template (CPT) through M1u.

2. The 5GMSu AF configures the 5GMSu AS according to the media transformation(s) specified in the Content Preparation Template (M3u) and responds to the Application Provider (M1u).

3. The media is streamed from 5GMSu Client to 5GMSu AS (M4u).

4. The content is transformed by 5GMSu AS according to the Content Preparation Template and egested to the Application Provider (M2u).

5. The Application Provider/Application may update, retrieve, or destroy the Content Preparation Template using M1u/M5u.

### 5.2.4.3 Content preparation between uplink ingest and downlink streaming

In this collaboration, the 5GMSu Application Provider requests content preparation for the uplink ingest stream from one UE and before downlink streaming to other UEs. Figure 5.2.4.3-1 shows such a scenario.

UE

5GMSu Client

5GMSu Aware Application

5GMSu AS

External DN

5GMS Application Provider

M1u

Trusted DN

M4u

5GMSu AF

M5u

PCF

M8u

N5

5GMSd AS

5GMSd AF

M1d

UE

5GMSd Client

5GMSd Aware Application

M4d

M5d

M3u

M3d

I2

I1

M8d

Figure 5.2.4.3-1: Content preparation after uplink ingest streaming

In this case:

1. The Application Provider defines the required content preparation and requests the 5GMSd AF to create the process using the Content Preparation Template (CPT) through M1d.
2. The 5GMSd AF configures the 5GMSd AS according to the media transformation(s) specified in the Content Preparation Template (M3d) and responds to the Application Provider (M1d).
3. The 5GMSu AF configures the 5GMSu AS according to the media transformation(s) specified in the Content Preparation Template (M3u) and responds to the Application Provider (M1u).
4. The content is transferred to 5GMSd AS and transformed according to the Content Preparation Template (I2). I2 is an implementation specific interface and is considered out of scope of this document.
5. The content is streamed to the 5GSd Client from 5GMSd AS (M4d).
6. The Application Provider/Application may update, retrieve, or destroy the Content Preparation Template (M1d).

NOTE 1: In the above steps, the I1 interface is not used for the simplicity. In section 5.6, two alternative call flows are presented that use I1 interface. I1 is an implementation specific interface and is considered out of scope of this document.

NOTE 2: In the deployment of this collaboration scenario, 5GMSu AS and 5GMSd AS may be co-located and therefore the I2 interface is not exposed. Similarly, 5GMSu AF and 5GMSd AF may be co-located.

### 5.2.5 Deployment Architectures

Editor’s Note: Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

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

#### 5.2.6.1 Call flow for content preparation before downlink streaming

The call flow is shown in Figure 5.2.6.1-1.



Figure 5.2.6.1-1: High-level call flow for content preparation before downlink streaming

Steps:

1. The 5GMSd Application Provider creates a Provisioning Session with the 5GMSd AF (M1d).

2. The 5GMSd Application Provider provisions the 5GMSd AF with one or more Content Preparation Templates defining instructions for content preparation, as well as the required output format(s) (M1d).

3. The 5GMSd AF, based on the received Content Preparation Templates, requests the 5GMSd AS to verify availability of resources for content preparation (M3d, procedures not specified):

a. Syntax checking of Content Preparation Template(s).

b. Semantic validation of Content Preparation Template(s).

c. Basic assessment of 5GMSd AS resource availability implied by the Content Preparation Template(s).

4. The 5GMSd AF acknowledges to the 5GMSd Application Provider the successful creation of Content Preparation Templates (M1d).

5. The 5GMSd Application Provider provisions the 5GMSD AF with a Content Hosting Configuration that references one or more Content Preparation Templates created in step 2 above (M1d).

6. The 5GMSd AF, based on the received Content Hosting Configuration, requests the 5GMSd AS to confirm the availability of distribution resources (M3d, procedures not specified).

7. The 5GMSd AF acknowledges to the 5GMSd Application Provider the successful creation of the Content Hosting Configuration (M1d).

8. The 5GMSd Application Provider feeds the content to the 5GMSd AS (M2d).

9. The 5GMSd Application Provider optionally provides the service access information to the 5GMSd-Aware Application (M8d, out of scope)

If needed, steps 10–14 may optionally be executed:

10. The 5GMSd-Aware Application requests the 5GMSd Application Provider to use the service (M8d).

11. The 5GMSd Application Provider provides the 5GMSd AF with updated Content Preparation Template(s) (M1d).

12. The 5GMSd AF, based on the modified Content Preparation Template(s), requests an updated confirmation of downlink streaming resource availability (M3d).

13. The 5GMSd AF acknowledges to the 5GMSd Application Provider that the Content Preparation Template(s) have been successfully updated (M1d).

14. The 5GMSd Application Provider acknowledges to the 5GMSd AF the use of the service (M8d).

The rest of the call flow concerns the 5GMS downlink streaming process:

15. The 5GMS-Aware Application request the 5GMSd Client to start an uplink streaming session (M6d/M7d).

16. If Service Access Information was not provided in step 9, the 5GMSd Client requests this information from the 5GSMd AF (M5d).

17. The 5GMSd Client requests start of the downlink streaming session from the 5GSMd AF (M5d).

18. The 5GMSd AF requests instantiation of the content preparation process (M3d).

19. The 5GMSd AS instantiates the media preparation process if it (or any of its parts) are not already running.

20. The 5GMSd AF acknowledges the instantiation of the content preparation process (M3d).

Steps 18–20 are not needed if another user has already requested the content and therefore the content preparation process is already running on the 5GMS AS, or if this process was instantiated in earlier steps of the workflow (e.g. in step 3 or step 12).

One use-case for steps 18–20 is when a user requests a content format that is not included in the previously instantiated content preparation process. In this case, 5GMSd AS may add new processes to the content preparation processing to provide the requested content format.

21. The downlink media streaming starts (M4d).

Finally:

22. The 5GMSd AS releases its resources after observing a period of interactivity.

NOTE: Step 22 is implementation-dependent.

#### 5.2.6.2 Call flow for content preparation after uplink streaming

The call flow is shown in Figure 5.2.6.2-1.



Figure 5.2.6.2-1: Call flow for content preparation after uplink streaming

Steps:

1. The 5GMSu Application Provider creates a Provisioning Session with the 5GMSu AF (M1u).

2. The 5GMSu Application Provider creates one or more Content Preparation Templates defining instructions for content preparation, as well as the required output format(s) (M1u).

3. The 5GMSu AF, based on the received Content Preparation Templates requests the 5GMSu AS to verify availability of content preparation and contribution resources, e.g. to allocate 5GMSu content ingest and contribution egest resources. (M3u, procedures not specified):

a. Syntax checking of Content Preparation Template.

b. Semantic validation of Content Preparation Template.

c. Basic assessment of 5GMSu AS resource availability implied by the Content Preparation Template.

4. The 5GMSu AF acknowledges to the 5GMSu Application Provider the successful creation of Content Preparation Templates, as well as successful provisioning of content preparation (M1u).

5. The 5GMSu Application Provider optionally provides the service access information to the 5GMSu-Aware Application (M8u, out of scope)

If needed, steps 6–10 may optionally be executed.

6. The 5GMSu-Aware Application requests the 5GMSu Application Provider to support impending reception of uplink streaming content (M8u)

7. The 5GMSu Application Provider provides the 5GMSu AF with updated Content Preparation Template(s) (M1u).

8. The 5GMSu AF, based on the modified Content Preparation Templates, requests an updated confirmation of uplink streaming resource availability (M3u).

9. The 5GMSu AF acknowledges to the 5GMSu Application Provider that the Content Preparation Templates have been successfully updated (M1u).

10. The 5GMSu Application Provider acknowledges to the 5GMSu AF the use of the service (M8u).

The rest of the call flow concerns the 5GMS uplink streaming process:

11. The 5GMSu-Aware Application request the 5GMSu Client to start an uplink streaming session (M6/7u).

12. If Service Access Information was not provided in Step 5, the 5GMSu Client requests this information from the 5GSMu AF (M5u).

13. The 5GMSu Client requests start of the uplink streaming session from the 5GSMu AF (M5u).

NOTE 1: Although the above step is defined by the stage 2 design in TS 26.501 [15], it is not realised in Release 16 by stage 3 procedures defined in TS 26.512 [16].

14. The 5GMSd AF requests instantiation of the content preparation process (M3u).

15. The 5GMSd AS instantiate the media preparation process if it has not started before (M3u).

16. The 5GMSd AF acknowledges the instantiation of the content preparation process (M3u).

Steps 14–16 may not be needed if the content preparation process has been instantiated during earlier steps in the call flow (such as between 3 and 8).

17. Uplink media streaming from the 5GMSu Client to the 5GMSu AS commences (M4u).

18. Media streaming egest from the 5GMSu AS to the 5GMSu Application Provider (M2u) commences.

Finally:

19. The 5GMSu AS releases its resources after observing a period of interactivity. Note that this is implementation dependent.

NOTE 2: The 5GMSu Application Provider also sets up the M2u configuration during the provisioning section. The details of such set-up are not shown in the above call flow.

Editor’s Note: The M2u configuration process is the subject of the uplink streaming topic of this study.

#### 5.2.6.3 Baseline call flow for content processing between uplink streaming and downlink streaming

Figure 5.2.6.3-1 shows the call flow for this scenario.



Figure 5.2.6.3-1: High-level call flow for content preparation  
between uplink streaming and downlink streaming

Steps:

1. Identical to steps 1–7 in 5.2.6.1-1.

2. The 5GMS Application Provider creates a Provisioning Session with the 5GMSu AF (M1u).

3. The 5GMSu AF requests the 5GMSu AS to confirm the uplink resources availability. (M3u, procedures not specified).

4. The 5GMSu AF acknowledges to the 5GMSu Application Provider of the successful provisioning (M1u).

5. The 5GMS Application optionally provides the service access information to the 5GMSu-Aware Application (M8u, out of scope).

6. The 5GMS-Aware Application request the 5GMSu Client to start an uplink streaming session (M6/7u).

7. If Service Access Information was not provided in step 5, the 5GMSu Client requests this information from the 5GSMu AF (M5u).

8. The 5GMSu Client requests start of the uplink streaming session from the 5GSMu AF (M5u).

9. The uplink media streaming starts (M4u).

1. The 5GMSu AS streams the content to the 5GMSd AS (I2, not specified).

11. The 5GMS Application Provider optionally provides the service access information to the 5GMSd-Aware Application (M8d, out of scope).

12. Identical to steps 10–14 in 5.2.6.1-1.

13. Identical to steps 15–21 in 5.2.6.1-1.

14. The 5GMSu AS releases its resources after observing a period of interactivity.

NOTE 1: Step 14 is implementation-dependent.

15. The 5GMSd AS releases its resources after observing a period of interactivity.

NOTE 2: Step 15 is implementation-dependent.

### 5.2.7 Potential open issues

#### 5.2.7.1 Open issues in collaboration scenario 1: Content preparation before downlink streaming

a) Open issue 1: Content Preparation Template information

TS 26.512 [16] only mentions that the format of the Content Preparation Template is identified by its MIME type. At least one format needs to be defined for the instruction in Content Preparation Template for common services such as multi-rate streaming.

b) Open issue 2: Support of other protocols

The current specification only supports HTTP pull and DASH-IF ingest. Other protocols may need to be added.

c) Open issue 3: Address translation for complex pull requests

Figure 5.2.7.2-1 shows an example in which the content preparation takes segments *n* and *n+1* from the upstream 5GMSd Application Provider and combines them into one segment with double duration for delivery to the 5GMSd Client. The goal could be to achieve higher compression efficiency for distribution.

5GMSd AS

Content Hosting

**M2d**

Content Preparation

Cache

**M4d**

5GMSd Client

5GMSd Application Provider

Request for Segment 246

Segment 246

Segment 247

Request for Segment 247

Segment 123

Request for Segment 123

Figure 5.2.7.2-1: An example of a media distribution by pull

Steps:

1. The 5GMSd Client requests a media segment http://cdn.com/segment123.mp4 through M4d.

2. The requested URL is mapped in the Content Hosting distribution configuration (“cdn.com” => “originprovider.com/video/” and transformation of request URL using the rule {n} => {n, n+1}). Two separate requests are made via M2d to the 5GMSd Application Provider for the following media segments:

- http://originprovider.com/video/segment246.mp4

- http://originprovider.com/video/segment247.mp4

3. The 5GMSd Application Provider provides the two requested media segments to the Content Preparation subfunction, which in turn merges the two segments into one, delivers the output segment to the Cache.

4. The Cache subfunction delivers the merged media segment to the UE via an M4d response.

NOTE: It is assumed that the initialization segment is available for initialization of the decoder in the content preparation module.

Table 7.6.3.1 1 in TS 26.512 [16] defines the content hosting configuration resource. Each Content Hosting Configuration may include one or more Distribution Configurations, each one with an optional Content Preparation Template. However, the Content Hosting Configuration is currently allowed to include only one Ingest Configuration.

The Ingest Configuration defines the input format for content preparation. Currently, two ingest protocols are specified in clause 8.2 of [16]: HTTP pull-based ingest and DASH-IF push-based. In the case of HTTP pull, if a request is received in M4d that can’t be satisfied, an HTTP pull request is made through M2. The PathRewriteRules currently defined in clause 7.6.3 of [16] do not address step 2 of the above example (i.e. transforming one request into two different requests).

#### 5.2.7.2 Open issues in collaboration scenario 2: content preparation after uplink streaming

In addition of the above issues of collaboration scenario 1, the following issues may be considered.

In this case, the content is delivered to the 5GMSu Application Provider through M2u.

c) Open issue 3: Egest protocols

TS 26.512 [16] Table 8.1-1 defines the ingest protocols at M2d. But the egest protocols for M2u are not defined. One possibility is use already defined ingest protocols also as possible egest protocols. However, it must be investigated whether the current definitions of these protocols are adequate for egest.

Editor’s Note: We recommend addressing this issue in uplink streaming topic of this study.

d) Open issue 4: Egest Configuration

TS 26.512 [16] Content Hosting Configuration resource defines an Ingest Configuration. It is not clear whether the same resource can be used for Egest configuration and whether the parameters are adequate.

Editor’s Note: We recommend addressing this issue in uplink streaming topic of this study.

#### 5.2.7.3 Open issues in collaboration scenario 3: content preparation between uplink and downlink

Since both uplink and downlink is used, the following issues should be considered.

e) Open issue 5: Signalling the connection between the uplink and downlink

Assuming the previous issues are addressed, then the Content Hosting Configuration resource for uplink and downlink would be separately used in M1u and M1d. The uplink Content Hosting Configuration’s egest configuration should be aligned with the downlink Content Hosting Configuration’s ingest configuration as is shown in the following figure.

The following issues must be addressed:

1. Is the protocol left to the network operator and only direct connection between two Application Servers signalled?

2. Is any (optional) standard protocol needed for the connection? If so, which protocols?

3. Are the current Ingest Configuration’s (and its counterpart for Egest) parameters adequate for setting up the connection between uplink and downlink?

4. Do the connection configuration and 5GMS Application Server’s addresses need to be hidden from the 5GMS Application Provider?

### 5.2.8 Candidate Solutions

#### 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 are 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) [38] 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.

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

The NBMP Workflow Description Document (WDD) [39] 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 [39].

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 transcoder’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.

#### 5.2.8.5 Address translation for complex pull requests

Open issue 3 of 5.2.7.1 provide a gap in address translation. There are three possibilities for addressing this gap:

1. Keep PathRewriteRules as is, and therefore exclude content preparation cases with more than one input and/or with more complex translation to be used with push protocol(s) only.

2. Extend the current request URL transformation capabilities of the 5GMSd Content Hosting feature.

3. Keep PathRewriteRules as is and include an additional (and more sophisticated) address translation function as a part of the Content Preparation subfunction of the 5GMSd AS.

Figure 5.2.8.5-1 shows the architecture for the third option above.

5GMSd AS

Content Hosting

**M2d**

Content Preparation

PathRewriteRules

Cache

**M4d**

5GMSd Client

5GMSd Application Provider

Request for Segment 246

Segment 246

Segment 247

Request for Segment 247

Segment 123

Request for Segment 123

Address Translator

Figure 5.2.8.5-1: A candidate solution for the example of media distribution by pull

Steps:

1. The 5GMSd Client requests a media segment http://cdn.com/segment123.mp4 through M4d.

2. The requested URL is mapped according to the PathRewriteRules in the Content Hosting distribution configuration and provided to the address translator of the Content Preparation module.

3. The address translator module transforms the received request into two separate requests via M2d to the 5GMSd Application Provider for the following media segments:

- http://originprovider.com/video/segment246.mp4

- http://originprovider.com/video/segment247.mp4

4. The 5GMSd Application Provider provides the two requested media segments to the Content Preparation subfunction, which in turn merges the two segments into one, delivers the output segment to the Cache.

5. The Cache subfunction delivers the merged media segment to the UE via an M4d response.

NOTE: It is assumed that the initialization segment is available for initialization of the decoder in the content preparation module.

## 5.3 Traffic Identification

### 5.3.1 Description

For different features within the 5G Media Streaming Architecture, it is necessary for the 5G System to identify the traffic flows. The increased usage of transport encryption (e.g. HTTPS) increases the difficulty of detecting the packets for certain application flows. Existing detection methods, such as using “significant parts of the URL to be matched” (contained in a Packet Flow Description, see below), are impractical for HTTPS traffic, since the URL is carried in the encrypted payload.

Multimedia streaming applications might not be able to uniquely identify the 5-tuple of the streaming session, since the 5-tuples are often changing. This is due to factors such as load balancing, CDN distribution, multiple concurrent requests for different types of resources, etc. This study will address how to properly configure the 5G System to enable efficient detection of application flows (service data flows) e.g. for event reporting, and QoS profile usage, etc.

Note that the TS 23.50x specifications use different terminology from the TS 29.xxx specifications. Furthermore, TS 23.503 [41] uses slightly different terms than TS 23.501 [23] and TS 23.502 [24]. The two common terms are defined in TS 23.503:

**- Packet flow:** A specific user data flow from and/or to the UE.

**- Service data flow**: An aggregate set of packet flows carried through the UPF that matches a service data flow template.

The terms *traffic detection* [23] and *application detection* [23] refer to the process of finding matching service data flows among all packet flows. This logic is defined in TS 23.503 as an *application detection filter*.

The procedures in TS 23.502 use the term *flow description*, which is only a subset of an *IP Packet Filter Set* (as defined in TS 23.501).

Figure 5.3.1-1 depicts the chain of functions (taken from TS 29.244 [26], Figure 5.2.1-1) within an UPF.



Figure 5.3.1-1: Packet processing flow in the UP function (Figure 5.2.1-1 from TS 29.244 [26])

The steps are as follows:

1. The UPF always first looks up the Packet Forwarding Control Protocol (PFCP) session context to which a packet belongs. The PFCP session context is an individual PDU session or a standalone PFCP session not tied to any PDU session.

2. Then there are so-called Packet Detection Rules (PDR), which implement traffic detection of the service data flows with respect to different conditions.

NOTE: A PDR is direction specific. Thus, an Uplink (UL) PDR and a Downlink (DL) PDR are needed to detect a bidirectional Service Data Flow.

3. Based on the PDR result, the next rules are executed, namely Multi-Access Rule (MAR), Forward Action Rule (FAR), QoS Enforcement Rule (QER), and Usage Reporting Rule (URR).

NOTE: Only the Forward Action Rule (FAR) is mandatory. The QoS Enforcement Rule (QER) is only present for QoS Flows. The Usage Reporting Rule (URR) is only available when traffic volume measurements (e.g. for charging) are needed.

The Packet Detection Rule (PDR) is based on Service Data Flow Templates, which contain one or more Service Data Flow (SDF) Filters or an Application Identifier. An Application Identifier refers to one or more Packet Flow Descriptions (PFDs).

A Service Data Flow (SDF) Filter contains for IP PDU Sessions a single IP Packet filter, i.e. any combination of

- Source/destination IP address or IPv6 prefix.

- Source / destination port number.

- Protocol ID of the protocol above IP/Next header type.

- Type of Service (TOS) (IPv4) or Traffic class (IPv6) and Mask.

- Flow Label (IPv6).

- Security parameter index.

- Packet Filter direction.

A PFD includes a PFD ID; and one or more of the following:

- 3-tuple(s) comprising protocol, server-side IP address and port number.

- the significant parts of the URL to be matched, e.g. host name.

- a domain name matching criterion and information about applicable protocol(s).

The application detection filter can be configured in the SMF and the SMF then provides it in the Service Data Flow Template to the UPF. Alternatively, the Service Data Flow Template for traffic handling in the UPF is received from the dynamic PCC Rule.

Besides, the Management of Packet Flow Descriptions enables the UPF to perform accurate application detection when PFD(s) are provided by an Application Service Provider (ASP) and then to apply enforcement actions as instructed in a PCC Rule.

The operator is able to configure pre-defined PCC Rules in the SMF or dynamic PCC Rules in the PCF. A PCC rule includes either a list of Service Data Flow filters or an application identifier for Service Data Flow detection. The PCC rule further includes charging control information, i.e. charging key and optionally a Sponsor identifier or an ASP identifier or both.

The application identifier references one or more PFDs, which are managed using the PFD Management API. Depending on the service level agreements between the operator and the Application Server Provider, it may be possible for the ASP to provide to the SMF individual PFDs or the full set of PFDs for each application identifier maintained by the ASP via the PFD Management service in the NEF (PFDF). The PFDs become part of the application detection filters in the SMF/UPF and are thereafter used as part of the logic to detect traffic generated by an application. The ASP may remove or modify some or all of the PFDs which have been provided previously for one or more application identifiers. The SMF may report the application stop to the PCF for an application instance identifier as defined in clause 5.8.2.8.4 of TS 23.501 [5] if the removed/modified PFD in SMF/UPF would result in an inability to detect traffic for that application instance.

The ASP manages (i.e. provisions, updates, deletes) the PFDs through the NEF (PFDF). The PFD(s) are transferred to the SMF through the NEF (PFDF). The PFDF is a logical functionality in the NEF which receives PFD(s) from the ASP through the NEF, stores the PFD(s) in the UDR and provides the PFD(s) to the SMF(s) either on the request from ASP PFD management through NEF (PFDF) (push mode) or on the request from SMF (pull mode). Finally, the PFDF functionality is a service provided by the NEF.

The ASP may provide/update/remove PFDs with an allowed delay to the NEF (PFDF). Upon reception of the request from the ASP, the NEF (PFDF) checks if the ASP is authorized to provide/update/remove those PFD(s) and request the allowed delay. The NEF (PFDF) may be configured with a minimum allowed delay based on SLA to authorize the allowed delay provided by the ASP. When both the requesting ASP and the requested allowed delay are successfully authorized, the NEF (PFDF) translates each external Application Identifier to the corresponding Application Identifier known in the core network. The NEF (PFDF) stores the PDF(s) into the UDR.

The Application Identifier is simply an index to a set of application detection rules configured in the UPF. It is an identifier that can be mapped to a specific application traffic detection rule.

The procedure is depicted Figure 5.3.1‑2 below:



Figure 5.3.1‑2:

The PFD (Packet Flow Description) is a set of information enabling the detection of application traffic.

Each PFD may be identified by a PFD ID. A PFD ID is unique in the scope of a particular Application Identifier. Conditions for when a PFD ID is included in the PFD are described in TS 29.551 [6]. There may be different PFD types associated with an Application Identifier.

### 5.3.2 Collaboration Scenarios

The 5GMSd Application Provider negotiates with the MNO an SLA to provide differentiated treatment, including network QoS and charging for its 5GMSd-Aware Application. The Application Provider provides the necessary information to the MNO to detect the traffic, to ensure its correct and exclusive identification. The MNO detects the traffic correctly and applies the agreed traffic treatment.

Due to privacy concerns, the content hosting is provided by the Application Provider in an external Data Network. However, the 5GMSd Application Provider leverages the network features either via a 5GMSd AF in the trusted Data Network (Figure 5.9.2‑1) or via a 5GMSd AF in the external Data Network (Figure 5.9.2‑2).



Figure 5.9.2-1: Collaboration 1 (Collaboration 3 of TS 26.501)



Figure 5.9.2-2: Collaboration 2 (Collaboration 4 of TS 26.501)

In order to use flow-based network features (such as different QoS classes or different charging policies), the 5G System needs to detect the relevant traffic

### 5.3.3 Deployment Architectures

The following figure depicts a potential architecture design for the realization of traffic detection. The architecture shows the involved network functions in the traffic detection.

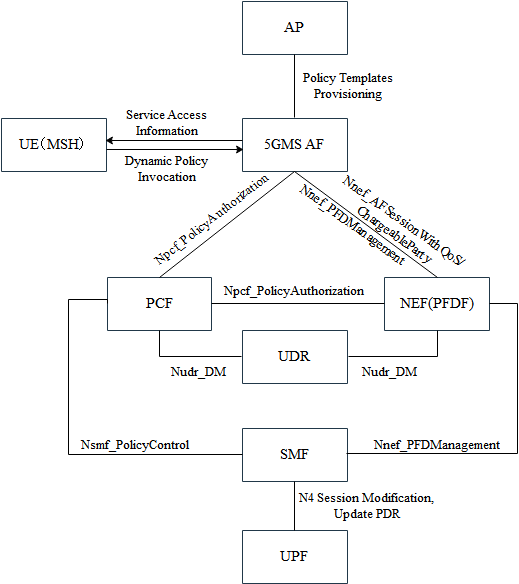


Figure 5.3.3-1: Relevant architecture components

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

#### 5.3.4.1 General

The Service Data Flow Templates support multiple different combinations to define parameters for traffic detection. This clause describes the common parameter combinations to detect specifically media streaming application traffic.

The Service Data Flow Template can take the form of either Service Data Flow filters (i.e. IP Packet Filter Sets) or an Application Id referencing Packet Flow Descriptions (PFDs).

An IP Packet Filter Set can contain different combinations of parameter values. Unspecified parameter values in the IP Packet Filter Sets are used to match any value of the corresponding information in the header of an IP packet. Common IP Packet Filter Set combinations are:

- 5-Tuple: The source and destination IP addresses, source and destination port numbers (potentially expressed as a small range of values) and the Protocol ID. This method of traffic detection is further described in clause 5.3.4.2 below.

- ToS: The source IP address and the Type of Service (ToS). This method of traffic detection is further described in clause 5.3.4.3 below.

Editor’s Note: Additional parameter value combinations such as 3-Tuple or usage of Flow Label (IPV6 only) can be beneficial.

A Packet Flow Description (PFD) can contain different parameters. Common parameters are:

- Domain Name: The Internet domain name of an application server. This method of traffic detection is not described further in the present document.

#### 5.3.4.2 Usage of 5-tuples for Traffic Identification

The application detection filters in the UPF can be configured based on a pre-configured PCC rule (i.e. in the SMF and provided to the UPF) or a dynamic PCC rule (i.e. provided by the PCF) By interacting with the PCF (possibly via the NEF) the 5GMS AF is able to provision, update and remove a dynamic PCC rule which contains Service Data Flow description parameters for traffic handling and traffic detection in the UPF.

When using 5-tuples for traffic detection, the following fields of the IP Packet Filter Set are used:

- Source/destination IP address or IPv6 prefix.

- Source/destination port number.

- Protocol identifier of the protocol above IP/Next header type.

- Packet Filter direction (uplink or downlink).

NOTE: These fields are encoded in the Flow Description field, defined in clause 5.3.8 of TS 29.514 [28].

As shown in figure 5.3.4.2-1 (below), the 5GMSd AF in the external DN can send a request using Nnef\_AFsessionWithQos API to provision, update or remove a request to reserve resources for a specific application/flow with specific flow descriptions. After the AF request authorization, the NEF interacts with the PCF, providing the flow description together with the QoS reference, the optional other parameters like Alternative Service Requirements, period of time or traffic volume, etc.



Figure 5.3.4.2-1:Flow description usage for traffic flow identification

If the request is authorised, the PCF determines the required QoS parameters based on the information provided by NEF/AF. After the Nnef\_AFsessionWithQoS\_Create Procedure, a transaction identifier is allocated by the NEF to identify this AF Session. Then the 5GMSd AF can subsequently invoke the Nnef\_AFsessionWithQoS\_Update API with this transaction identifier to update the flow description.

Alternatively, the 5GMSd AF in the trusted DN can directly send a request using Npcf\_PolicyAuthorization API to provision, update and remove a request to reserve resources for a specific application/flow with specific flow descriptions.

Then the PCF initiates the PDU Session modification procedure to provide the updated PCC rule to the SMF and the SMF updates the PDRs in the UPF for the application/traffic identification and policy handling.

However, when a new TCP connection is opened and the old one is closed, the 5-tuple in the Flow Description needs to be changed. (This may be from the consequence of factors such as load balancing, multiple concurrent requests for different types of resources, use of a shared TCP connection pool, etc.) In such cases, the 5GMSd AF can invoke the NEF/PCF-related APIs with new flow description to update the PDRs installed in UPF to follow the changed transport layer 5-tuples for application/flow identification.

Editor’s Note: Whether a single or multiple modification procedures are needed depends on further check and study.

#### 5.3.4.3 Usage of ToS Traffic Class for Traffic Identification

The following is a simplified call flow when using the ToS Traffic Class for Traffic Identification, meaning, only the Type of Service field is used within a SDF Filter. The Type of Service (ToS) is an 8-bit field within the IP header (both IPv4 and IPv6) that can be used as DiffServ Code Point (DSCP) value [29] and for ECN marking [30]. It is assumed here that the QoS flow should be used (e.g. for Premium QoS) as described in TS 26.512, Annex A.



Figure 5.3.4.3-1: ToS usage within an application traffic detection rule (simplified)

Figure 5.3.4.3-1 depicts a call flow for ToS-based traffic detection. It is assumed here that the 5GMSd AF provides the ToS value for traffic identification in the Policy Activation response message (step 5). Another solution might be that the Media Session Handler allocates a ToS value and then provides the value to the 5GMSd AF.

The call flow works as the following steps:

1: The Media Session Handler activates a Dynamic Policy and provides the Policy Template Id with the activation request (among other parameters).

The 5GMSd AF triggers the activation of a Dynamic PCC rule:

2: The 5GMSd AF uses the Policy Authorization Service API and triggers a PCC rule activation. The 5GMSd AF provides the UE IP address, an IP Packet Filter Set with the ToS value and the UE IP address of the requesting UE and QoS parameters.

3: As result, the PCF uses the Npcf\_SMPolicyControl APIs to provide a new PCC rule to the SMF.

4: The SMF uses the N4 interface to provide a new Packet Detection Rule (PDR) together with other rules for the UE to the UPF. Once the new rule is installed in the UPF, the UPF starts taking actions on the detected traffic.

5: If the Dynamic Policy can be activated, the 5GMSd AF provides a value for the ToS field in return.

NOTE 1: The ToS Value is not immediately provided to the Media Session Handler to prevent race conditions.

6: The Media Player prepares a new TCP connection and sets the ToS value nominated by the 5GMSd AF on the TCP socket using the setsockopt() API or equivalent. As a result, all TCP packets for the flow will be marked by the UE with the ToS value.

7: The TCP Connection is established, and the traffic is marked with the ToS field. The UPF detects the traffic (by inspecting the IP header) and handles it according to the policy in the PCC Rule.

NOTE 2: The PCC Rule is scoped by the PDU Session, so the treatment of the ToS field value by the UPF is limited to the requesting UE. The UPF first looks up the relevant PDRs for a PDU session based on the incoming GTP Tunnel Id.

The UPF also needs to detect the downlink traffic matching the uplink traffic. There are different solutions to achieve this:

A: The 5GMSd AS uses the same ToS field for downlink traffic as used for uplink traffic.

NOTE 3: This solution may not work for cases where traffic crosses operational domain boundaries, since the ToS header field is often reset by border IP routers.

B: The UPF captures the 5-tuple carrying a specific ToS field from the TCP SYN Packet that establishes the connection in the uplink direction. As result, the UPF automatically creates a new PDR in the opposite direction derived by inverting the address fields found in the SYN packet.

NOTE 4: The connection handshake of other transport protocols may be more difficult to detect.

C: Often, the UEs in a PLMN are shielded from public Internet traffic by means of firewalls that employ Network Address Translation (NAT). In order to set the ToS field within the Trusted DN to an appropriate value, the N6‑NAT may set the downlink ToS to the same value as the uplink ToS.

NOTE 5: This is similar to solution A above.

#### 5.3.4.4 Usage of Packet Flow Descriptions for Traffic Identification

The following are potential and simplified call flows for the realization of the traffic identification.

In the first call flow (Figure 5.3.4.4‑1) the provisioning step is described, in which one or more PFDs for a single application are provisioned. The provisioned PFDs for a single application are identified by the Application Identifier.



Figure 5.3.4.4-1: PFD Provisioning using the PFD Management API (simplified)

In the second call flow (Figure 5.3.4.4‑2) the update procedure for the PFD to adjust to an actual session is described.



Figure 5.3.4.4-2: PFD usage within an application detection filter (simplified)

### 5.3.5 Potential open issues

The exact behaviour and information that needs to be provided to and by the 5GMSd AF as well as the MSH need to be specified.

The following open issues have been identified:

1. The Npcf\_PolicyAuthorization API as defined in TS 23.502 [24] only supports usage of a flow description or an application identifier. The flow description is not further defined in TS 23.501 or TS 23.502. In Stage 3 specifications, a flow description represents only a 5-tuple. Other information elements of the Service Data Flow Filter are not supported.

2. The Nnef\_ChargeableParty and Nnef\_AFsessionWithQOS APIs only support usage of a flow description. The flow description is not further defined in TS 23.501 or TS 23.502. Other information elements of the Service Data Flow Filter are not supported.

3. The Npcf\_PolicyAuthorization API Stage 3 as defined in TS 29.514 [42], only supports a flow description and a ToS value. However, it is not possible to define whether the ToS value should be used in uplink traffic detection or downlink traffic detection.

4. The Nnef\_AFsessionWithQOS and Nnef\_ChargeableParty stage 3 APIs, as defined in TS 29.522 [43], only supports a Packet Flow Description (through the FlowInfo Type). Other information elements of the Service Data Flow Filter are not supported. Note, the FlowInfo Type from TS 29.122 [44] is different from the FlowInformation Type in TS 29.512 [45].

### 5.3.6 Candidate Solutions

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

## 5.4 Additional/new transport protocols

### 5.4.1 Description

#### 5.4.1.1 General

Media streaming applications are continued to use HTTP-based distribution protocols, but newer versions of HTTP such as HTTP/2 or HTTP/3 are introduced, see for example also TR 26.925 [5], clause 6.1.4. The architectural and performance impacts of such protocols for 5G-based media distribution is unclear and requires study. The study also considers how Media Players may use functionalities existing in new transport protocols, and also investigate the impact of new transport protocols on 5GMS usage and traffic identification (e.g. Service Data Flow Descriptions).

Based on [X], HTTP protocol (also known as web protocol), powers most websites, mobile apps, and videos. It was created by Tim Berners-Lee at CERN in 1989 and has been enhanced over the years to keep up with the ever-changing World Wide Web. Currently, the web is a mixture of HTTP/1.1 [3] and HTTP/2 [4] adoption. Most well-known websites are running HTTP/2, while smaller websites and late adopters plan to migrate to HTTP/2 in the near future as it is relatively easy to implement. HTTP/2 is used by about 45% of websites and supported by all major web browsers. HTTP/3 is only used by about 5% of websites now and not well-supported by web browsers yet. However. significant HTTP/3 deployments are emerging. For example, YouTube™ has for a long time been offering a pre-RFC draft version to any client that wants to use it, especially the Chrome™ browser. Other browsers are expected to follow soon after waiting for the QUIC and HTTP/3 RFCs to be published before mainlining that feature.

HTTP/2 introduces the "Streams" concept at HTTP level and each stream can have different priorities. All objects can from a web-page can be multiplexed in single long-lived TCP connection. Also, HTTP/uses header compression (HPACK) to avoid verbose/clear text. Also, HTTP/2 pseudo-mandates TLS to prevent “middle boxes” from messing up with the content. However, HTTP/2 does not remove the drawbacks of TCP’s head-of-line blocking - packet loss on one stream will block all other streams until recovery even if packets for all other streams are correctly received.

HTTP/2 testing shows [2] that the delivery of large objects over HTTP/2 can be slower than over HTTP/1.1 when there is packet loss. This is because HTTP/2 uses a single TCP connection, versus about six connections which most web browsers open over HTTP/1.1. In addition, the TCP congestion control algorithms reduce the TCP congestion window size, resulting in fewer bytes sent over the wire when using just one TCP connection.

HTTP/2 provides on average a 5% to 15% performance improvement on page load times over HTTP/1.1 [2]. HTTP/1.1 allows persistent TCP connections, but requests still had to be serialized, resulting in the well-known "HTTP head of queue blocking". In order to improve downloads, many TCP flows still needed to be parallelized to speed up delivery.

The solution to this problem is to use HTTP/2 over a different transport protocol that provides more efficient congestion control. One option would be to upgrade and modify TCP, but modifying TCP implementations is viewed as an impossible task. For example, middle boxes such as NAT, Firewalls, and Load balancers are problematic, because they get rarely upgraded which prevents any updates to TCP. TCP is also hard to evolve as it is almost always implemented as part of operating system kernels, requiring an updated operating system as part of TCP updates. Hence, it was considered easier to introduce transport functions on top of UDP, outside the operating system kernel, in the user space – referred to as QUIC.

That, in essence, is what HTTP/3 [5] is: HTTP/2 over User Datagram Protocol (UDP) based on IETF QUIC. HTTP/3 is a thin layer on top of QUIC [32] including QPACK header compression [31]. The main QUIC functions are connection and stream multiplexing [32], fast startup[32], loss recovery, in-order delivery (within stream) [32], flow control [32]. TLS1.3 (handshake) [33], loss recovery and congestion control [34].

By multiplexing multiple concurrent logical streams over a single UDP-based transport association, and by giving each stream its own independent byte offset numbering space, packet loss in one stream does not block progress on other logical streams in the same QUIC connection. (However, the affected stream will still block when packets are lost, so as to guarantee in-order delivery of payloads to the application.).

A screenshot of a cell phone

Description automatically generated

Figure 5.4-1: HTTP/2 and HTTP/3 protocol stacks

For an entertaining introduction to QUIC and HTTP/3, please check <https://www.youtube.com/watch?v=B1SQFjIXJtc>.

However, using QUIC for adaptive streaming still requires study as under certain circumstances, the quality using QUIC may even degrade for DASH-based streaming than it would increase [6]. The evaluation results show that using the unmodified DASH algorithms on top of QUIC may not provide the anticipated performance boost when compared to the standard DASH over TCP.

The main expected benefit of QUIC is being able to multiplex requests for all Adaptation Sets onto the same transport association, and then to manage the network QoS on that aggregate connection. This has a valuable operational benefit to a CDN operator (including the 5GMS AS) in reducing the number of UDP ports that a server needs to keep open. Another benefit is being able to migrate connections from one IP address to another with minimal interruption to either client or server. This is useful when the client moves, but it is also useful when the server changes (e.g. in edge computing relocation Use Cases).

Because HTTP/3 and IETF QUIC are new protocols, there are several questions about performance and management that need to be investigated during this study.

#### 5.4.1.2 Performance Considerations for HTTP/3 using IETF QUIC over 5G Networks

The IETF specifications for HTTP/3 [5][31] and the core QUIC functions [32, 33, 34] are now approved in the IETF, and have been broadly deployed by a number of browser vendors and content providers, since the IETF QUIC working group has focused on specification, implementation, and, after the specifications were sufficiently stable enough, deployment, all in parallel. The performance of HTTP/3 over IETF QUIC in environments that have not been encountered during deployments to date is still an open question. Of greatest interest for this study, is the performance of HTTP/3 over IETF QUIC in 5G networks. Although deployment of 5G networks has begun, most deployment experience with HTTP/3 over IETF QUIC in mobile networks over the past few years has been in non-5G networks.

When end users have used HTTP/3 over QUIC to access servers outside the 5G core network, 3GPP terminal mobility was handled transparently by the 3GPP network, and the UE’s IP address(es) didn’t change. If a server’s IP addresses changed, this was often not visible to the user, due to the widespread deployment of CDNs and load balancers in data centers. If edge computing resource IP addresses change in relocation use cases, QUIC connection migration could be used to reduce the impact on user experience, but this needs to be analyzed carefully.

MPEG-DASH has provided years of good user experience running over HTTP/1.1. As use cases arise which require very low latency, it is reasonable to better understand how, and when, MPEG-DASH might take advantage of HTTP/3 and analyze whether this has any implications for this study.

[6] raised the issue that MPEG-DASH performance might be lower over HTTP/3 than over HTTP/2. This reference was comparing Google’s pre-standardization QUIC implementation to highly optimized HTTP/2 over TCP implementations, and is about five years old, as of this writing, but the point remains – we need to know more about unmodified MPEG-DASH over standardized HTTP/3 implementations.

In particular, in the downlink media streaming Use Case, the server chooses the algorithm but the client media player decides whether to use a long-lived connection or to drop and reconnect when changing representation/rendition or adaptation/switching set, and the decision on connection (re)use is often delegated by the client application to a library, and this library might not offer an API to influence connection reuse. QUIC optimisations for reducing connection setup time (0-RTT or 1-RTT handshake (as described in [32]) mitigate this suboptimal use of connections to some extent, but the if the client, directly or indirectly by delegating the decision to the library, chooses to close and open connections, any QUIC congestion control algorithm begins probing for available bandwidth with no information about path characteristics. The connection handling characteristics of common APIs need to be analyzed as part of this study.

#### 5.4.1.3 Performance Considerations for IETF QUIC over 5G networks

The standardized QUIC congestion control and recovery procedures in [34] are chosen to emulate TCP’s standardized behaviors ([35], plus extensions). These are quite conservative, and not match current work on delay-based congestion control and recovery mechanisms, which have also seen wide deployment in QUIC implementations. In principle, delay-based congestion control and recovery mechanisms should improve user experience for streaming media applications, but this isn’t known yet, and this needs to be carefully analyzed. One of the core functions of QUIC is the capability to migrate connections without application involvement when endpoint IP addresses change, rather than requiring the detection of a connection failure, teardown of that connection, and setup of a new connection. Connection migration is one of the key QUIC functions that we do not have a great deal of experience with – implementers in the IETF said they were concentrating on performance for a connection, and many had not completely implemented or tested connection migrations at scale (data point is from October 2020).

#### 5.4.1.4 Management Considerations for HTTP/3 and QUIC in 5G networks

One of the biggest distinctions between HTTP/2 over TCP and HTTP/3 over QUIC has been the encryption of almost all transport-level information in QUIC. This information, which was not encrypted in TCP even when it was carrying encrypted payloads, was often used in network management to identify and troubleshoot performance problems on the Internet. In most of experience with HTTP/3 over QUIC deployments to date, content providers (e.g. Google/YouTube, Facebook, etc.) have terminated at least one end of the encrypted end-to-end connections, allowing them to identify problems at the QUIC transport level. That might be true in 5G deployments, or it might not be the case. If that is not the case, it would be very useful to consider the guidance in [36] as part of this study. It is also likely that operator deployments which relied on split-TCP connections to improve performance over radio links will require reconsideration for QUIC-based t, since QUIC transport-level information is not available unless a device has a security context for the encrypted QUIC connection. It is likely that we need to enhance the 3GPP QoS framework, and that if an application is using QUIC, 5-tuples are not sufficient for per-flow QoS.

Although QUIC can be implemented as part of operating system kernels, it is commonly implemented in user space, allowing frequent updates to congestion control and recovery procedures, including introduction of entirely new procedures (e.g. BBR, "Bottleneck Bandwidth and Round-trip propagation time" congestion control [37]). It is likely that the performance characteristics of HTTP/3 applications will vary more dynamically than HTTP/1.1 implementations that have been used by MPEG-DASH in the past. This may also have implications for existing applications if they migrate from MPEG-DASH over HTTP/1.1 or HTTP/2, to MPEG-DASH over HTTP/3.

### 5.4.2 Collaboration Scenarios

A service provider/content provider runs an adaptive media streaming service between HTTP/3 and QUIC enabled 5G Media Streaming AS and an HTTP/3 and QUIC enabled UE using 5G Media Streaming over M2d and M4d.

Editor’s Note: Study collaboration scenarios between the 5G System and Application Provider for each of the key topics.

### 5.4.3 Deployment Architectures

Editor’s Note: Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

### 5.4.4 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.

### 5.4.5 Potential open issues

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

### 5.4.6 Candidate Solutions

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

## 5.5 Uplink media streaming

### 5.5.1 Description

#### 5.5.1.1 Overview

Uplink media streaming functionality is currently under-specified in TS 26.501 [15] and TS 26.512 [16]. This part of the study investigates the gaps and potential solutions for completing the uplink streaming procedures, and associated protocols and APIs.

#### 5.5.1.2 Gap analysis of TS 26.501

The original focus of Rel-16 3GPP TS 26.501 [15] is on the overall system architecture, description of high-level procedures and call flows pertaining to downlink media streaming services.

In TS 26.501, there is significant imbalance in the scope and details of the described procedures for downlink vs. uplink media streaming. Key issues include the following:

1. The procedures for downlink media streaming include session establishment, provisioning of various types of configuration information, metrics reporting, consumption reporting, dynamic policy and network assistance. In comparison, the described procedures for uplink media streaming are limited to session management, remote control, and network assistance. It is unclear for uplink streaming whether and how the configurations for content preparation, content protocols discovery, dynamic policy, metrics reporting, etc., should be defined and how the associated functions will operate.

- Absent in clause 6 are procedural description and call flow regarding QoE metrics collection and reporting in uplink media streaming as compared to the presence of such text in clause 5.5 on metrics collection and reporting in downlink media streaming.

2. For downlink streaming, it is described that access to Service Access Information by the 5GMSd Client may be provided either over M8d by the Application Provider, or else fetched by the Client via M5u. For uplink streaming, the solely described method for the 5GMSu Client to obtain Service Access Information is via application metadata delivery over M8u. The only exception to this rule is the alternative method for provisioning Service Access Information to the 5GMSu Client by the 5GMSu AF, associated with remote control sessions in reference to remote control use cases and operational mechanisms in the context of FLUS (Framework for Live Uplink Streaming) as described in TR 26.939 [13] and TS 26.238 [14]. However, it should be noted that due to the limited description of the relationship between the uplink streaming framework and use cases, defined procedures and APIs in TS 26.512, it is unclear whether or how remote control sessions associated with uplink streaming delivery can make use of those interface procedures and APIs.

- Clause 4.3 is missing the description of Service Access Information parameters for a metrics configuration set pertaining to uplink media streaming, as compared to the presence of such information for downlink media streaming in clause 4.2.3. In particular, the “Metrics” parameter of the downlink streaming metrics configuration set is explicitly bound to the 3GPP “metrics” scheme and corresponds to one or more of the QoE metrics for either a progressive download or 3GP-DASH streaming service as defined in TS 26.247 [26]. On the other hand, given the lack of definition by SA4 of uplink streaming QoE metrics collection and reporting related functionality for existing application service specifications, there is no associated list of metrics on collection and reporting by the 5GMSu Client that can be referenced.

#### 5.5.1.3 Gap analysis of TS 26.512

Similar to TS 26.512, the originally-defined features, protocols and APIs for 5GMS services mainly pertain to downlink media streaming services. More recently, as part of the specification maintenance process, additional descriptive, clarification and corrective text towards supporting uplink streaming services has been incorporated. However, remaining shortcoming include:

1. The latter informationis centered on identifying relevance of existing and downlink streaming centric interface functionality for uplink streaming. For instance, provisioning and subsequent execution of content protocol discovery, dynamic policy invocation, and metrics reporting are regarded as applicable to support for uplink media streaming services. Also, content preparation via M1-based configuration of Content Preparation Templates may be considered a means for defining manipulations by a 5GMSu AS of media content uploaded from the 5GMSu Client. Such content manipulation may be associated with network-based media processing (NBMP) of uplink-delivered streaming media content (e.g., user-generated content of social media services, professionally-produced uplink streaming of sports events or concerts) as defined in FLUS [13] and further evaluated in the FS\_FLUS\_NBMP study item. However, further assessment of the potential linkage between content preparation and NBMP should be conducted.

2. Although the Metrics Reporting Configuration resource as part of M1 provisioning procedures and described in clause 4.3.9 is intended to apply to either downlink or uplink media streaming, the corresponding data model of that resource (see clause 7.8.3) contains no explicitly-referenceable metrics for collection and reporting associated with uplink streaming services. This is due to the absence of a 3GPP-defined metrics scheme for uplink media streaming as it exists for downlink media streaming (e.g., the 3GPP scheme urn:‌3GPP:‌ns:‌PSS:‌DASH:‌QM10).

- It would be desirable in this study to identify a candidate set of QoE metrics associated with uplink media streaming services, and as specified by a 3GPP-defined metrics scheme, to be collected by the 5GMSu Client and reported to the 5GMSu AF.

- Up to now, in the development of the 5GMS architecture and associated protocols and APIs, very little attention has been given to defining control plane functionality that can offer unique/beneficial features to the end-user, network operator or Application Provider in the context of uplink media streaming service operation. Such value-added functionality could be enabled by leveraging the information available in the 5GMS network (e.g., subscription class, service characteristics, user/device mobility, network conditions) and which may be static or dynamic in nature. This is also an area that should be further studied.

#### 5.5.1.4 Gap analysis between TS 26.238 (FLUS) and TS 26.512 (5G Media Streaming)

TS 26.238 provides the following features:

1. The FLUS Control Source may discover multiple FLUS sinks.

2. The FLUS Control Source may discover the capabilities of each discovered FLUS Sink, including its network-based media processing capabilities.

3. The FLUS Control Source may also request a FLUS Sink to perform media processing.

4. The UE capabilities (formats, connectivity protocol, remote control) may be discovered by a FLUS Control Sink.

TS 26.512 uplink streaming currently lacks the above features.

### 5.5.2 Collaboration Scenarios

Editor’s Note: Study collaboration scenarios between the 5G System and Application Provider for each of the key topics.

#### 5.5.2.1 Overview

A set of key collaboration scenarios between an 5GMSu Application Provider and the 5G System Provider is described below.

Scenarios associated with Media Plane only collaboration and which may also involve downlink media distribution are presented first. Subsequently, scenarios pertain to both Control Plane and Media Plane collaboration and which may or may not involve downlink media distribution are presented.

NOTE: In the collaboration scenario descriptions and diagrams below, an interface or API marked with a prime (′), e.g., M1′ or M2u′, denotes that while that interface/API functionally maps to its 3GPP-defined counterpart (e.g., M1 or M2u), its protocol and format are defined by the 5GMSu Application Provider. The implementation of these interfaces is up to the 5GMSu Application Provider discretion.

#### 5.5.2.2 Collaboration Scenario 1

This scenario pertains to a media plane only collaboration for which the 5GMSu AS is deployed in the trusted domain. Here, the 5GMS System Provider is assumed to offer uplink streaming capabilities as a service to an external 5GMSu Application Provider.

NOTE: Although a Provisioning Session is shown in Figure 5.5.2.2-1 between the (external) 5GMSu Application Provider and the 5GMSu AF, due to the absence of the M5u interface in this diagram, there is no control plane collaboration between the 5GMSu Application Provider and the 5GMS System Provider.



Figure 5.5.2.2-1: Collaboration 1

#### 5.5.2.3 Collaboration Scenario 2

This scenario pertains to a media plane only collaboration for which the 5GMSu AS is deployed in the external domain and the 5GMSu AF is not involved. Specifically, the M1′ and/or M2u′ protocols do not follow 3GPP specifications.



Figure 5.5.2.2-1: Collaboration 2

#### 5.5.2.4 Collaboration Scenario 3

In this collaboration scenario, both the 5GMSu AS and 5GMSu AF are present. The 5GMSu AS resides in the external domain and does not employ 5GMS protocols and formats for uplink media reception from the 5GMSu Client, nor for content egest. The 5GMSu AF is used to interact with the 5G System, e.g., for dynamic policy invocation and/or other uplink streaming related network features such as metrics reporting and network assistance).



Figure 5.5.2.3-1: Collaboration 3

#### 5.5.2.5 Collaboration Scenario 4

In this collaboration scenario, both the 5GMSu AS and 5GMSu AF are present and follow 3GPP specifications. Both the 5GMSu AS and 5GMSu AF reside in the external DN/domain.



Figure 5.5.2.4-1: Collaboration 4

#### 5.5.2.6 Collaboration Scenario 5

This scenario is similar to collaboration scenario 4, with the exception that the 5GMSu AS and 5GMSu AF reside in the trusted DN/domain. An additional difference from collaboration scenario 4 is that the M2u API is used by the external 5GMSu Application Provider for content egest.



Figure 5.5.2.5-1: Collaboration 5

#### 5.5.2.7 Collaboration Scenario 6

This scenario represents a hybrid, i.e. end-to-end form of collaboration across uplink media streaming and downlink media streaming services. An external 5GMS Application Provider relies on the 5GMS System Provider to support both the uplink streaming media transmission by 5GMSu Clients and subsequent distribution of that content via downlink media streaming for reception by 5GMSd Clients.

Editor’s Note: The study on the key topic “Content Preparation” includes a use case whereby content preparation is used between uplink and downlink media streaming, and that use case can be considered as a specific example of this collaboration scenario.



### 5.5.3 Deployment Architectures

Editor’s Note: Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

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

#### 5.5.4.1 Collaboration scenario 1 call flow

Figure 5.5.4.1‑1 provides a high-level call flow for the scenario depicted in clause 5.5.2.2.



Figure 5.5.4.1-1: Collaboration scenario 1 Call flow

Steps:

1. The 5GMSu Application Provider creates a Provisioning Session for uplink streaming with the 5GMSu AF.

2. The 5GMSu Application Provider creates a Content Publishing Configuration as part of the Provisioning Session that defines the instructions for content egest (M1u).

3. The 5GMSu AF, based on the received Content Publishing Configuration, requests the 5GMSu AS to instantiate the content preparation process (M3u).

4. The 5GMSu AS initialises the content preparation process.

5. The 5GMSu AS acknowledges the initialisation of the required process (M3u).

NOTE: M3u procedures between the 5GMS AF and the 5GMS AS are outside the scope of TS 26.512 [16].

6. The 5GMSu AF acknowledges the successful creation of the Content Publishing Configuration to the 5GMSu Application Provider (M1u).

At some later point in time:

7. The 5GMSu Application Provider optionally provides Service Access Information to the 5GMS-Aware Application (M8, out of scope).

8. Uplink media streaming starts from the 5GMSu Client to the 5GMSu AS (M4u).

9. Media streaming egest starts from the 5GMSu AS to the 5GMSu Application Provider (M2u).

Finally:

10. The 5GMSu AS releases its resources after observing a period of inactivity.

NOTE: Step 10 is implementation-dependent.

#### 5.5.4.2 Collaboration scenario 2 call flow

Figure 5.5.4.2‑1 provides a high-level call flow for the scenario depicted in clause 5.5.2.3.



Figure 5.5.4.2-1: Collaboration scenario 2 Call flow

Steps:

1. The 5GMSu Application Provider creates a Provisioning Session for uplink streaming with the 5GMSu AF (M1u′).

2. The Provisioning function requests the 5GMSu AS to initialise the required content prepatation process (M3u′).

3. The 5GMSu AS initialises the content preparation process.

4. The 5GMSu AS acknowledges the initialisation of the required process (M3u′).

5. The Provisioning function acknowledges the successful creation of the Provisioning Session to the 5GMSu Application Provider (M1u′).

At some later point in time:

6. The 5GMSu Application Provider provides Service Access Information to the 5GMS-Aware Application (M8, out of scope).

7. Uplink media streaming starts from the 5GMSu Client to the 5GMSu AS (M4u).

8. Media streaming egest starts from the 5GMSu AS to the 5GMSu Application Provider (M2u).

Finally:

9. The 5GMSu AS releases its resources after observing a period of inactivity.

NOTE: Step 9 is implementation-dependent.

#### 5.5.4.3 Collaboration scenario 3 call flow



Figure 5.5.4.3-1: Collaboration scenario 3 Call flow

Steps:

1. The 5GMSu Application Provider creates a Provisioning Session with its internal Provisioning function (M1u′).

2. The Provisioning function requests the 5GMSu-like AS to initialise the required content preparation process instantiation (M3u′).

3. The 5GMSu-like AS instantiates the content preparation process.

4. The 5GMSu-like AS acknowledges the Provisioning the instantiation of required process (M3u′).

5. The Provisioning function acknowledges successful provisioning to the 5GMSu Application Provider (M1u′).

6. The 5GMSu Application Provider creates a Provisioning Session for uplink streaming with the 5GMSu AF.

X. The 5GMSu Application Provider creates a Content Publishing Configuration as part of the Provisioning Session that defines the instructions for content egest (M1u).

At some later point in time:

7. The 5GMSu Application Provider optionally provides Service Access Information to the 5GMS-Aware Application (M8, out of scope).

8. The 5GMS-Aware Application requests the 5GMSu Client to start an uplink streaming session (M6u/M7u).

9. The 5GMSu Client optionally (and in the case step 5 was not performed) requests Service Access Information from the 5GSMu AF (M5u).10. Uplink media streaming starts from the 5GMSu Client to the 5GMSu-like AS (M4u′).

11. Media streaming egest starts from the 5GMSu-like AS to the 5GMSu Application Provider (M2u′).

Finally:

12. The 5GMSu AS releases its resources after observing a period of inactivity.

NOTE: This step is implementation dependent.

#### 5.5.4.4 Collaboration scenario 4 call flow



Figure 5.5.4.4-1: Collaboration scenario 5 Call flow

Steps:

1. The 5GMSu Application Provider creates a Provisioning Session for uplink streaming with the 5GMSu AF (M1u′).

2. The 5GMSu Application Provider creates a Content Publishing Configuration as part of the Provisioning Session that defines the instructions for content egest (M1u′).

3. The 5GMSu AF, based on the received publishing configuration, requests the 5GMSu AS to confirm the availability of content resources for egest (M3u).

NOTE: M3u procedures between the 5GMS AF and the 5GMS AS are outside the scope of TS 26.512 [16].

4. The 5GMSu AF acknowledges the successful creation of the Content Publishing Configuration to the 5GMSu Application Provider (M1u′).

At some later point in time:

5. The 5GMSu Application Provider requests that the 5GMSu AF initialises the content preparation process (M1u′).

6. The 5GMSd AF requests initialisation of the content preparation process (M3u).

7. The 5GMSd AS initialises the content preparation process, if is not already running (M3u).

8. The 5GMSd AS acknowledges the initialisation of the content preparation process (M3u).

9. The 5GMSu AF acknowledges the initialisation of the cotent preparation process (M1u′).

10. The 5GMSu Application Provider provides Service Access Information to the 5GMS-Aware Application (M8, out of scope).

11. The 5GMS-Aware Application requests the 5GMSu Client to start an uplink streaming session (M6u/M7u).

Alternatively:

12. The 5GMS-Aware Application requests the 5GMSu Client to start an uplink streaming session (M6u/M7u).

13. The 5GMSu Client requests Service Access Information from the 5GSMu AF (M5u).

14. The 5GMSd AF requests initialisation of the content preparation process (M3u).

15. The 5GMSd AS initialises the content preparation process, if is not already running (M3u).

16. The 5GMSd AF acknowledges the initialisation of the content preparation process (M3u).

17. The 5GMSMu AF provides Service Access Information to the 5GMSu Client (M5u).

Then:

18. Uplink media streaming starts from the 5GMSu Client to the 5GMSu AS (M4u).

19. Media streaming egest starts from the 5GMSu AS to the 5GMSu Application Provider (M2u′).

Finally:

20. The 5GMSu AS releases its resources after observing a period of inactivity.

NOTE: This step is implementation-dependent.

#### 5.5.4.5 Collaboration scenario 5 call flow



Figure 5.5.4.5-1: Collaboration scenario 5 Call flow

Steps:

1. The 5GMSu Application Provider creates a Provisioning Session with the 5GMSu AF.

2. The 5GMSu Application Provider requests the 5GMSu AF to create one Content Publishing Configuration that defines the instructions for content egest (M1u).

3. The 5GMSu AF, based on the received Content Publishing Configuration, requests the 5GMSu AS to confirm the availability of content resources for egest.

NOTE 1: M3u procedures between the 5GMS AF and the 5GMS AS are outside the scope of TS 26.512 [?].

4. The 5GMSu AF acknowledges to the 5GMSu Application Provider the successful creation of the Content Publishing Configuration (M1u).

At some later point in time:

5. The 5GMSu Application Provider requests that the 5GMSu AF initialises the content preparation process (M1u).

6. The 5GMSd AF requests initialisation of the content preparation process (M3u).

7. The 5GMSd AS initialises the content preparation process, if is not already running (M3u).

8. The 5GMSd AS acknowledges the initialisation of the content preparation process (M3u).

9. The 5GMSu AF acknowledges the initialisation of the cotent preparation process (M1u).

10. The 5GMSu Application Provider provides Service Access Information to the 5GMS-Aware Application (M8, out of scope).

11. The 5GMS-Aware Application requests the 5GMSu Client to start an uplink streaming session (M6u/M7u).

Alternatively:

12. The 5GMS-Aware Application requests the 5GMSu Client to start an uplink streaming session (M6u/M7u).

13. The 5GMSu Client requests Service Access Information from the 5GSMu AF (M5u).

14. The 5GMSd AF requests initialisation of the content preparation process (M3u).

15. The 5GMSd AS initialises the content preparation process, if is not already running (M3u).

16. The 5GMSd AF acknowledges the initialisation of the content preparation process (M3u).

17. The 5GMSMu AF provides Service Access Information to the 5GMSu Client (M5u).

Then:

18. Uplink media streaming starts from the 5GMSu Client to the 5GMSu AS (M4u).

19. Media streaming egest starts from the 5GMSu AS to the 5GMSu Application Provider (M2u).

Finally:

20. The 5GMSu AS releases its resources after observing a period of interactivity.

NOTE: This step is implementation dependent.

As is shown, a new resource type, the Content Publishing Configuration is added. This describes the configuration of the egest (M2u) used in step 19.

#### 5.5.4.6 Collaboration scenario 6 call flow

The call flow for this collaboration scenario is described in 5.2.6.3.

### 5.5.5 Potential open issues

#### 5.5.5.1 Potential open issues in 5G Media Streaming stage 3

The following open issues seem to exist in TS 26.512 [16]:

1. Lack of a standard template (or clear reference on how to use an existing standard template) for inclusion in a Content Publishing Configuration, i.e. to be able to provide content preparation instructions in a defined, interoperable format that the 5GMS AF supports through M1.

2. Lack of definition of protocols for media egest from the 5GMSu AS to the 5GMSu Application Provider in uplink throughvia M2u.

NOTE: The Content Protocols Discovery APIs allows the 5GMSu Application Provider to discover the supported egest protocols by 5GMSu AS. However, clause 8.1 of TS 26.512 does not currently list any specific egest protocols alongside those for downlink ingest streaming.

3. Lack of content publishing API, i.e. a similar functionality to Content Hosting Configuration in downlink streaming, for provisioning the uplink streaming through M1u..

4. Lack of Service Access Information for uplink streaming.

For downlink streaming, TS 26.512 [16] defines a StreamingAccess object as part of the Service‌Access‌Infromation resource. The StreamingAccess object includes a URL string that points to a media download resource or a manifest that describes a media presentation. In the case of uplink streaming, TS 26.512 does not yet specify which uplink streaming protocols are supported in M5u. Furthermore, it is not clear how the Media Session Handler would retrieve the entry point for uplink streaming to the 5GMSu AS.

#### 5.5.5.2 Potential open issues compared with FLUS

##### 5.5.5.2.1 General

Clause 5.5.1.3 describes the uplink streaming features from TS 26.238 [14] that are missing from TS 26.512 [16]. This section translates these missing FLUS features into potential new 5G Media Streaming features.

Table 5.5.5.2 shows a list of FLUS features and the equivalent features missing from TS 26.512. Note that in this table, the missing features of TS 26.512 are only listed for further discussion below, i.e. this is not a list of proposed features to be added.

Table 5.5.5.2‑1: Mapping existing additional features of FLUS to 5GMS architecture

|  |  |  |  |
| --- | --- | --- | --- |
| Feature # | Existing support in FLUS | Equivalent in 5GMS | Needed or not? |
| 1 | The FLUS Control Source can discover multiple FLUS sinks. | The 5GMSu Client can discover multiple 5GMSu AS instances. | Supported by Edge Application Server (EAS) profile discovery as defined in TR 26.803 [Y] (see Discussion 1 below). |
| 2 | The FLUS Control Source can discover the capabilities of each discovered FLUS Sink, including its network-based media processing capabilities. | The UE5GMSu Client can discover the capabilities of each discovered 5GMSu AS. | Supported by EAS profile discovery (see Discussion 1 below). |
| 3 | The FLUS Control Source can also request a FLUS Sink to perform media processing. | The UE can also request the 5GMSu AS to perform media processing. | Not needed if the Content Preparation Template supports a generic media processing description such as NBMP (see Discussion 2 below). |
| 4 | The UE capabilities (formats, connectivity protocol, remote control) may be discovered by a FLUS Control Sink. | The 5GMSu Client capabilities may be discovered by 5GMSu AF. | Not needed in this form, since this information can be provided by 5GMS Application Provider (see Discussion 3 below). |

See the discussions below for further explanation.

##### 5.5.5.2.2 Discussion 1

The FLUS Discovery Server provides the means for a FLUS Control Source to discover multiple FLUS sinks and their capabilities. In the 5GMS architecture, various 5GMSd AS instances might have different capabilities. However, TS 26.512 does not provide a framework for describing 5GMS AS capabilities or any capability-based discovery mechanism.

TR 26.803 [46] proposes an edge-enabled 5GMS architecture for discovering EAS-enhanced 5GMSd AS instances and their capabilities by an Edge-Enabled Client (EEC) using EAS discovery filters. One possible way to discover 5GMSu AS capabilities and/or to instantiate a new 5GMSu AS with the desired capabilities is to use the procedure described in TS 26.803 [46] for the 5GMSu AS. This approach requires that the 5GMSu Client’s Media Session Handler supports the EEC logical function, the 5GMSu AF supports the EES logical function, and the 5GMSu AS supports the EAS logical function, as defined by TS 23.558 [47].

##### 5.5.5.2.3 Discussion 2

In some deployment scenarios, the request for media processing is performed by a FLUS Control Source by including a media processing document in its request to the FLUS Control Sink. Since in the present document content preparation is investigated for uplink streaming collaboration scenarios (clause 5.2.4.2 in the content preparation key topic), such functionality can also be used for media processing. If the content preparation template supports a generic media processing description framework such as NBMP, then content preparation can be used to provide equivalent functionality to media processing in the FLUS specification.

The 5GMS Content Preparation Template is provisioned through the M1 interface whereas in FLUS it is possible that the media processing is provisioned using the equivalent of the M5u interface. To provide the UE with the ability to provision Content Preparation Templates, the following are possible options:

A. The 5GMSu-Aware Application (if needed) provides the desired Content Preparation Template to the 5GMSu Application Provider via M8u and then the Application Provider requests provisioning of the Content Preparation Template through M1u, or

B. The Media Session Handler in the 5GMSu Client requests the setting up of a Content Preparation Template by direct interaction with the 5GMSu AF via M5u. In this case, M5u needs to be extended to support Content Preparation Template provisioning requests from the Media Session Handler.

The current design supports option A. Option B seems unnecessary for the following reasons:

- It wouldn’t be scalable to maintain a separate uplink streaming Provisioning Session at the 5GMSu AF for each and every UE, especially as the number of UEs becomes large.

- In the current design, it is possible to create a separate Provisioning Session for each class of UE. In this approach, the UE signals its capabilities to the 5GMSu AF when requested Service Access Information at M5u, and then it is the task of 5GMSu AF to match the declared UE capabilities against the right uplink Provisioning Session metadata when responding to the request.

##### 5.5.5.2.4 Discussion 3

In the 5GMS architecture, the session is generally provisioned by the 5GMSu Application Provider. The Application Provider may already know the 5GMSu Client’s capabilities, for example through information in a user profile, or provided by the 5GMSu-Aware Application via M8u. Therefore, the need for the 5GMSu AF to discover the 5GMSu Client capabilities through M5u seems unnecessary.

### 5.5.6 Candidate Solutions

Since TS 26.512 [16] already defines solutions for the content ingest concerning the open issues of 5.5.5, one possible approach is to allow similar data structures, APIs, and protocols for content egest.

NOTE: The candidate solution provided in the present document merely shows the desired features (by mirroring the distribution features) and is not intended as a proposed solution.

#### 5.5.6.1 Content egest protocols

The existing ingest protocols can be used for egest.

Table 5.5.6.1-1: Adding egest content protocols

| Description | Term identifier | Clause |
| --- | --- | --- |
| Content ingest protocols at interface M2d | | |
| HTTP pull-based content ingest protocol | urn:3gpp:5gms:content-protocol:http-pull-ingest | 8.2 |
| DASH-IF push-based content ingest protocol | urn:3gpp:5gms:content-protocol:dash-if-ingest | 8.3 |
| **Content egest protocols at interface M2u** | | |
| HTTP pull-based content ingest protocol | urn:3gpp:5gms:content-protocol:http-pull-ingest | 8.2 |
| DASH-IF push-based content ingest protocol | urn:3gpp:5gms:content-protocol:dash-if-ingest | 8.3 |

The highlighted rows indicate the added protocols.

#### 5.5.6.2 Content Publishing Configuration API

An M1u API, similar to the M1d Content Hosting Configuration API as used for content ingest, can be defined for content egest.

Table 2: Operations supported by the Content Egest Configuration API

|  |  |  |  |
| --- | --- | --- | --- |
| Operation | Sub‑resource path | Allowed HTTP method(s) | Description |
| Create Content Publishing Configuration | content-publishing-configuration | POST | Used to create a Content Publishing Configuration resource. |
| Retrieve Content Publishing Configuration | GET | Used to retrieve an existing Content Publishing Configuration. |
| Update Content Egest Configuration | PUT,  PATCH | Used to modify an existing Content Egest Configuration. |
| Delete Content Publishing Configuration | DELETE | Used to delete an existing Content Publishing Configuration. |
| Purge Content Publishing Configuration cache | content-publishing-configuration/purge | POST | This operation is used to invalidate some or all cached media resources associated with this Content Publishing Configuration. |

As is shown in the table, the sub-resource path in particular is changed for this resource, and other aspects remain identical to Content Hosting Configuration API.

#### 5.5.6.3 Content Publishing Configuration Template

The Content Publishing Configuration resource, modelled after the Content Hosting Configuration resource, is shown in Table 5.5.6.3-1.

Table 5.5.6.3-1: Definition of Content Publishing Configuration resource

| Property name | Data Type | Cardinality | Description |
| --- | --- | --- | --- |
| name | String | 1..1 | A name for this Content Publishing Configuration. |
| PublishingConfiguration | Object | 1..1 | Describes the 5GMSu Application Provider's origin server to which media resources will be egested via interface M2u. |
| path | String | 1..1 | The relative path which will be used to address the media resources at interface M2u.  This path is provided by the 5GMSu AF in the case of pull-based egest. |
| pull | Boolean | 1..1 | Indicates whether to the 5GMSu AS shall use Pull or Push for egesting the content. |
| protocol | URI String | 1..1 | A fully-qualified term identifier allocated in the name space urn:3gpp:5gms:content-protocol that identifies the content egest protocol.  The set of supported protocols is defined in Table XXX. |
| entryPoint | String | 1..1 | An entry point to egest the content. The semantics of the entry point are dependent on the selected egest protocol.  In the case of Pull ingest (pull flag is set to True), this parameter is returned by the 5GMSu AF to the 5GMSu Application Provider and indicates the entry point for pulling the content. In this case, the *entryPoint* shall be used as the base URL. In this case, the relative URL content address is provided out of band (e.g. with a manifest through M8u) to the 5GMSu Application Provider.  In case of Push (pull flag is set to false), the entryPoint shall be provided to the 5GMSu AF to indicate the location to which content is to be pushed. In this case, the *entryPoint* shall be used as the base URL. |
| ContributionConfigurations | Object | 1..N | Specifies content preparation for the egested content. |
| contentPreparationTemplateId | String | 0..1 | Indicates that content preparation prior to egest is requested by the 5GMSu Application Provider. |
| canonicalDomainName | String | 1..1 | All resources of the upload shall be accessible through this default FQDN assigned by the 5GMSu AF. |
| certificateId | String | 0..1 | When content is distributed using TLS [16], the X.509 certificate for the origin domain is shared with the 5GMSd AF so that it can be presented by the 5GMSd AS in the TLS handshake at M2d. This attribute indicates the identifier of the certificate to use. |

Note that in the above table:

1. The Pull mode is defined when the Application Service Provider pulls the content from the 5GMSu AS, and conversely, the Push mode is defined when the 5GMSu AS pushed the content to Application Service Provider.

2. Each parameter description is updated based on item 1.

3. Since in the Pull mode, the Application Service Provider needs to have the content URL addresses, it is assumed that that information is provided by other means (e.g. through M8u by the 5GMSu Aware Application).

4. The DistributionConfigurations property in the Content Hosting Configuration resource for M1d is here replaced by the ContributionConfigurations property.

5. The ContributionConfigurations.contentPreparationTemplateId property identifies the Content Preparation Template to be used. Like the Content Preparation Template resource as defined for downlink streaming in TS 26.512 [16], the data model of this resource is determined by its MIME content type.

#### 5.5.6.4 Uplink entry point

The StreamingAccess object in the Service Access Information resource can be extended to support uplink streaming entry points. This object may include the following information:

* A URN, indicating an uplink streaming protocol provisioned for use over M4u (e.g. MPEG DASH, HLS, DASH-IF Ingest profile 1 or profile 2)
* The entry URL for the above service (i.e. address of the 5GMSu AS for uplink streaming delivery by Media Streamer over M4u).

Additionally, the StreamingAccess property may include alternative media uplink streaming protocols for the same Provisioning Session by making it an array. For instance, two different entry points may be described in the Service Access Information for uplink streaming using DASH-IF Ingest profile 1 and profile 2.

An example of such extension is shown in the following table:

Table 5.5.6.4‑1: Definition of ServiceAccessInformation resource  
(based on an extract from clause 11.2.3.1 of TS 26.512 [16])

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Property name | Type | Cardinality | Usage | Description |
| provisioningSessionId | String | 1..1 | RO | Unique identification of the M1d Provisioning Session. |
| … | | | | |
| StreamingAccess | Array(Object) | 1..1 | RO |  |
| mediaEntryType | Urn | 1..1 | RO | A fully-qualified term identifier from the controlled vocabulary urn:3gpp:5gms:content-protocol, as specified in clause 8, indicating the type of media service available at mediaEntry. |
| media~~Player~~Entry | Url | 1..1 | RO | For downlink streaming: Depending on the type of media entry indicated in mediaEntryType, either a pointer to a document that defines a media presentation (e.g. MPD for DASH content) that can be consumed via M4d, or else the URL of a media resource that can be streamed at M4d.  For uplink streaming: Depending on the type of media entry indicated in mediaEntryType, either a URL endpoint on the 5GMSu AS to which media can be streamed directly at M4u, or else the URL of a document that can be downloaded from the 5GMSu AS which contains the parameters for uplink media streaming at M4u. |
| … | | | | |

## 5.6 Background traffic

### 5.6.1 Description

Mobile Network Operators (MNO) are faced with the challenge of overload of their networks during peak hours. The following diagram shows a typical distribution of traffic over the day hours in a residential area:

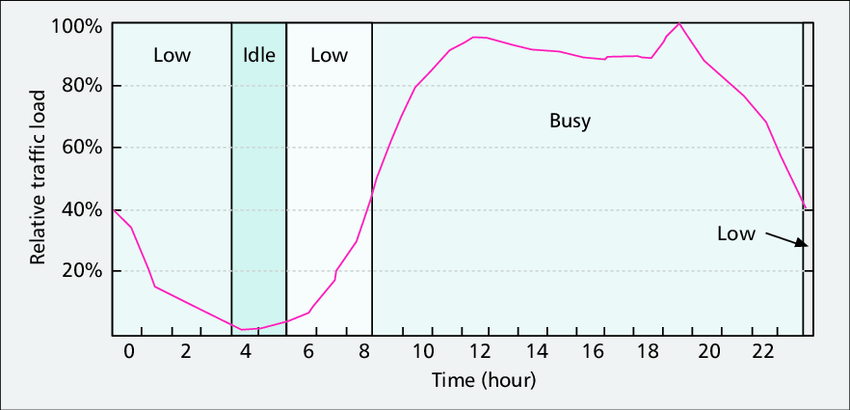


Figure 5.6.1-1: Example of traffic distribution over time

Note that the traffic distribution is also highly dependent on the geographic area. For instance, traffic distribution in a residential area maybe quite different from traffic distribution in a business or commercial area. Another example is traffic along roads during commute hours, which maybe higher by orders of magnitude than traffic during other hours.

As can be seen from the diagram, the traffic distribution is non-uniform/uneven throughout the day, which leads to congestion during the peak hours and very low utilization during off-peak hours. To alleviate this situation, the MNO may incentivize offloading traffic to off-peak hours to balance the network resource usage throught the day. The incentives may be provided in terms of preferential charging and guaranteed QoS.

### 5.6.2 Collaboration Scenarios

MNO and content provider enter an SLA that allows the content provider to distribute its content during off-peak hours to a set of receivers in a pre-determined geographic location.

Collaboration scenarios 2,4,5,7, and 8 from [TS256.501] are potential collaboration scenarios for traffic offloading using Background Data Transfer.

### 5.6.3 Deployment Architectures

There is no anticipated change to the 5G media streaming architecture to enable the usage of Background Data Transfer (BDT) for media distribution.

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

The following is a potential high-level call flow for the configuration and usage of a BDT session in 5G media streaming:



Figure 5.6.4-1: Potential call flow for BDT session configuration and establishment

### 5.6.5 Potential open issues

The extensions to the client APIs, Provisioning API, and 5GMSd AF services to enable signaling and management of background data traffic sessions need to be specified.

### 5.6.6 Candidate Solutions

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

## 5.7 Content-Aware Streaming

### 5.7.1 Description

Content-Aware Encoding and statistical multiplexing of services are important and relevant technologies in the media industry. The impacts and opportunities of such technologies for 5GMS is not fully understood and requires study. For example, the currently-defined 5GMSd Application Function (AF) based network assistance solution is exclusively triggered by the Media Player, which instructs the Media Session Handler to interact with the Network. It might be more efficient for such network assistance functionality to be obtained directly from the content provider based on dynamic content complexity. Greater interaction with the 5GMS Application Provider during the lifetime of a session should be studied.

According to [2], if one analyses, almost any movie or television show scene by scene, you’ll notice the content has varying needs in terms of its fundamental complexity. Scenes with a lot of action and detail need a lot more bits in order to hit a quality target, whereas other scenes—say, a newsreader delivering a monologue—can achieve the same quality target with a reduced number of bits.

As an example, a game sequences provided for XR Traffic was encoded with x265 over 1 minute in Figure 5.7.1‑1. One can see that at the same quality, the number of bits required to represent the content can be quite different.

Figure 5.7.1-1: Bit rate and quality over time for an example sequence.  
(Blue bits, red PSNR in dB × 100)

Ideally, to maintain quality, one wants the bit rate to vary over time to maintain consistent quality regardless of the complexity of the scene. Four different scene types may be considered, and they differ in complexity- easy, moderate, hard, and very hard to compress. The “very hard” content might be a panning shot over a crowd, a shot of confetti falling, or simply a scene with a lot of high motion. Scenes such as these require more bits to convert all the motion and detail into a high-quality output that can be decoded and recreated accurately. A moderate scene, perhaps a close-up of a car, or an easy scene, like a single person speaking with no camera movement, will require fewer bits to deliver the same quality target as the harder scenes. In order to most efficiently encode the entire video, ideally a rate control mode that allocates more bits to the complex scenes, and fewer bits to the easier ones.

Different rate control algorithms exist:

* **CBR:** Constant-Bit Rate encoding keeps the bit rate at a constant level, but the quality fluctuates. In ancient systems such as MPEG-2 TS, this is even addressed by sending lots of filler data just to keep the pipe constant
* **VBR:** Variable Bit Rate encoding following the principle from above to keep the quality constant. This is often also referred to as Content-Aware Encoding nowadays (CAE).
* **Capped VBR:** in this case the basic idea is to ensure that you have a mix of the above, i.e. a certain bit rate is never exceeded, but in case the content does not need the data rate, less data is sent.

The below diagram attempts to address and show these issues, but is more confusing then helpful.



Figure 5.7-2: Capped VBR rate control encoding compared with CBR and VBR

Variance in bit rates for different users may also result on the device and the consumption model. For a smaller screen, quite likely quality and bit rate requirements can be lower than for example going to a large screen such as a 4K TV.

The 3GPP QoS model contradicts this, as typically resources and QoS parameters are assigned for a session and only GBR is addressed.

From a adaptive bitaret streaming perspective, this content model needs to also be viewed as part on the streaming model, as the complexity of the content may be addressed based on the buffer availability, and also the situation of the network needs to be studied.

1) On-demand Streaming:

a. Stationary streaming of C-VBR/CBR content: Typically one operates with receiver buffer levels of 5-30 seconds [check details in TS 26.512]. One tries to keep the buffer filled. As soon as your buffer drains below some threshold, the client triggers a down-switch to a more sustainable bit rate. Switching typically can happen at segment boundaries, for example every 2 seconds.

b. Start-up and seek. In this case, one starts basically starts from an empty buffer. In order to have quick and stable start up and good quality right away, there may be a benefit to get a higher short-term bit rate from the network to fill the buffer quicker to at least the switching threshold as you would not start playback until the threshold is reached. This is a very instantaneous action and needs to be fulfilled instantanteously, at most after 1 second.

c. Stationary streaming of VBR/CAE content: In this case you basically operate on buffers of 5-30 seconds as above. The client typically has a map of the bit rate over time profile. In this case the client knows how much bit rate it needs for the next 5-10 seconds in order to keep the buffer stable and it can provide this information in a continuous manner to the network. The network will then grant a certain bit rate. This aspect may be fulfilled with using existing 5GMS functionalities.

2) Live Streaming and especially low-latency live streaming:

a. *General:* In this case the buffer is something of duration 1-5 seconds, it can be kept really low for low-latency streaming. Typically, one operates e2e latency of 3-5 seconds, so the buffer in the client is low. In addition, the client does not know the exact bit rate of the content as it is produced on the fly. Switching can typically be done every 1–2 seconds.

b. *Stationary streaming CBR:* the buffer is much more susceptible, and you may have a threshold of maybe 500 ms when the client needs a fast arriving Segment is not arriving fast enough. This aspect may be fulfilled with using existing 5GMS functionalities.

c. Start-up is similar to on-demand streaming as your buffer is anyways low. So no difference.

d. Yet another and probably the most interesting case is the live case, for which the content and each of Representations are VBR encoded, but more following the content complexity and VBR/CAE is done as shown below. The content complexity is not known in advance, but it needs to be provided on an content ingest interface to the network. In this case, the network should provision very fast and dynamically the bit rate if needed, but can relax.

There are many other cases, where content complexity and device characteristics need to be taken into account when addressing quality of service.

### 5.7.2 Collaboration Scenarios

In the following, difference collaboration scenarios are provided. In Figure 5.7-3, content is generated by a third-party content provider in different formats and configurations, taking into account for example:

1. Different device types (resolution, frame rates, codecs).
2. Different streams (target qualities and bit rates).
3. Encoding parameters (CBR, VBR, etc.).



Figure 5.7-3: Content-Aware Streaming based on static parameters

The network and client can make use of this information in order to optimize the streaming.

In a second variant, not only static information is provided, but also dynamic information with the media stream. This data is provided from the content provider to the 5G Media Streaming system and the client.



Figure 5.7-4: Content-Aware Streaming based on dynamic parameters

The client can use the information for:

* Optimizing its own quality.
* Acting fairly in a way that it only requests higher bit rates when the content is more complex, but leaves remaining capacity to the community.

In an On-Demand service, the use of the information is client controlled, i.e. the network is unaware of content complexity.

* Client downloads a description of the content variations and the associated quality initially.
* Client now brokers with the network for an average bit rate, but ability to request higher bit rate when content is complex
* DASH client includes logic to use the VBR options smartly to ask for “boosts” ahead of time when content is complex

Live Streaming and Ingest (network more actively included in streaming):

* Encoder provides as early as possible indication that content for same quality is getting a complex.
* Network uses this and identifies, if and how to fulfill this for the clients that request it.
* To not confuse client throughput estimation, communication between network and client is necessary.

### 5.7.3 Deployment Architectures

Editor’s Note: Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

### 5.7.4 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.

### 5.7.5 Potential open issues

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

### 5.7.6 Candidate Solutions

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

## 5.8 Network Event usage

### 5.8.1 Description

#### 5.8.1.1 Events exposed by 5GMS AF

The 5GMS AF performs several critical support operations for media streaming sessions. It also is responsible for collecting information about the progress and status of media streaming sessions. This information may be of interest to the AP or to other NFs in the network.

Thee 5G architecture defines event exposure mechanisms by the AF to other NFs in the network. TS 23.501 [23] and TS 23.502 [24] define the stage 2 Exposure service that can be offered by the AF. In TS 29.517 [25], the stage 3 realization of the Event Exposure service is specified as a RESTful API.

The resource structure is replicated in the following figure for convenience:



An Event Consumer subscribes to an application event and provides a URL on which it desires to receive the related event notifications. Both periodic reporting and immediate event reporting options are available. Event Filter Information and Event Reporting Information definitions as specified in TS 23.502 [24] and TS 29.517 [25] are used by the Event Consumer to indicate the desired event parameters and method of reporting for the selected event set.

So far, the following AfEvents are defined:

|  |  |  |
| --- | --- | --- |
| Enumeration value | Description | Applicability |
| SVC\_EXPERIENCE | Indicates that the event subscribed is service experience data for an application. | ServiceExperience |
| UE\_MOBILITY | Indicates that the event subscribed is UE mobility information. | UeMobility |
| UE\_COMM | Indicates that the event subscribed is UE communication information. | UeCommunication |
| EXCEPTIONS | Indicates that the event subscribed is exceptions information. | Exceptions |

Additional AFEvents may be defined.

#### 5.8.1.2 Events consumed by 5GMS AF

The 5GMSd AF may subscribe for event notifications related to application sessions, and receive event notifications for these sessions. Currently, TS 26.512 [16] requires the 5GMSd AF to subscribe with the PCF for the following types of event notifications:

- Service Data Flow QoS notification control;

- Service Data Flow deactivation;

- Resources allocation outcome.

Additional events of interest to the 5GMSd AF are for further study.

### 5.8.2 Collaboration Scenarios

The Application Provider (AP) is outsourcing part of its content hosting to the MNO. The AP makes use of the Provisioning APIs to configure its content distribution. The AP would like to track the usage of the network resources for the distribution of its content as well as the QoE for its mobile subscribers. At the same time, it may choose to limit access to this information by the MNO to protect the AP’s service interests and/or user’s privacy. The AP configures data collection from UEs and the 5GMSd AS(s) to identity the data to be collected as well as the permitted entities and access levels to that data. The 5GMSd AF triggers the data collection accordingly and uses the AF Event Exposure framework to notify Event consumers about collected data and events.

### 5.8.3 Deployment Architectures

The deployment architecture for the data collection and exposure by the 5GMSd AF is depicted by the following figure:

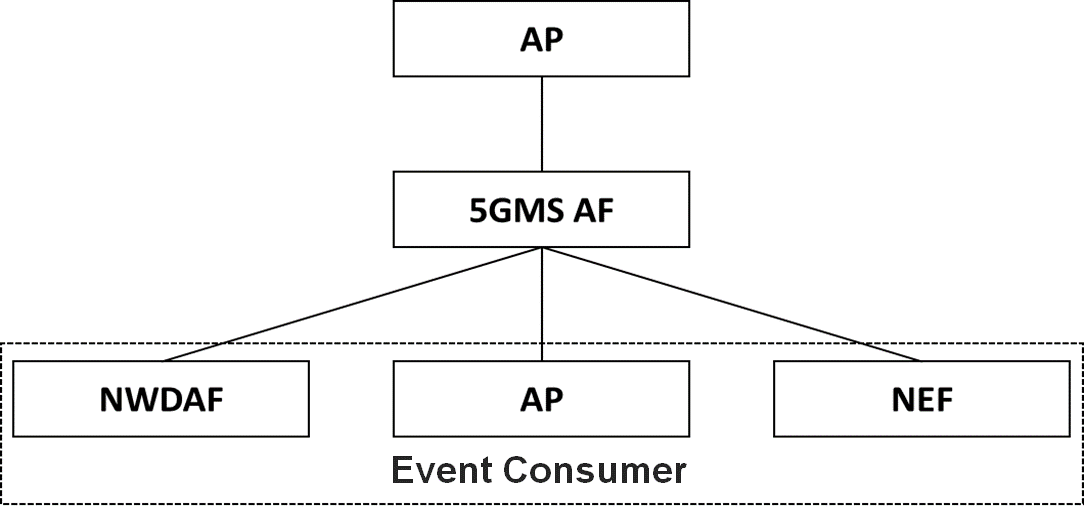


Figure 5.8.3‑1: Provisioning and Exposure Architecture

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

The following is a sample call flow of the operation of the data collection and exposure by the 5GMSd AF:



Figure 5.8.4‑1: Call flow for event exposure

A description of the steps is as follows:

1. *Provision data collection:* The Application Provider provisions the data collection by configuring the data that is to be collected for all streaming sessions of this Provisioning Session.

2. *Trigger data collection:* The 5GMS AF determines the data that is to be collected from the UE and provides this configuration to the Media Session Handler.

3. *Trigger data collection:* The 5GMS AF determines the data that is to be collected by the 5GMS AS and uses the M3 interface to configure the 5GMS AS.

4. *Subscribe-notify pattern – Subscribe:* In this phase, Event Consumers subscribe for the reception of notifications from the 5GMS AF. This phase may happen at any time. Subscriptions may be updated and canceled.

5. *Consumption and QoE reports:* The Media Session Handler collects logs and information and sends them to the 5GMS AF.

6. *Access logs:* The 5GMS AS collects logs and sends the information to the 5GMS AF

7. *Data aggregation and filtering:* In this phase, the 5GMS AF aggregates the information received in the data collection phase, determines for each Event consumer the allowed level of access, and provides the information to the Event Consumers.

8. If an Event Consumer opted for the subscribe-notify pattern, the 5GMS AF sends a notify message to that Event Consumer.

9. In case the Event Consumer opted for the Query pattern, by setting the immRep flag to true, the 5GMS AF sends a one-time response with the event information to that Event Consumer.

### 5.8.5 Potential open issues

The following events are expected to be defined:

* Consumption reporting: this event contains reports sent by the UE about the consumption of the service, in terms of streaming session establishment.
* Quality of Experience Metrics reporting: this event contains reports sent by the UE about the QoE of the streaming sessions.
* Network Assistance: this event contains reports collected by the AF on the usage of network assistance such as bitrate recommendation and bitrate boost requests that have been offered for a streaming session.
* QoS and Charging usage: this event contains reports about QoS and charging policy requests made by the 5GMS AF for a streaming session.
* Access logs: this event contains reports about the access to the streaming content that is hosted by the network.

The granularity of the access to this information depends on the authorized access level. For each of these events, the triggers for the data collection and the levels of access to the collected data need to be defined as well.

### 5.8.6 Candidate Solution

#### 5.8.6.1 UE Data Collection via Direct and Indirect Methods

The 5GMS AF is designed to be the recipient of various types of application event information specific to media streaming, such as consumption reports, metrics reports and AF-based network assistance interactions, from the 5GMS Client (its Media Session Handler) over the M5 interface. Such *collection* of UE data is referred to in SA2’s eNA\_Ph2 work item (see TS 23.288 [48]) as “direct data collection”.

It is possible that data collection from the UE is accomplished by the M8 application layer interface between the 5GMS-Aware Application and the 5GMS Application Provider (or, more generically, the Application Service Provider), and in turn forwarded by the ASP to the 5GMS AF, for further exposure s event services to NF consumers. Such mechanism, also referred by SA2 in [48] as “indirect data collection”, is not specified in TS 26.501 [6] and TS 26.512 [7], mainly because M8 functionality is outside the scope of 5GMS, and additionally, there was no use case/requirement in the 5GMSA/5GMS3 work items to drive the related stage 2 and stage 3 definition. On the other hand, due to the outreach from SA2 for AF support of indirect data collection, as shown by Figure 5.8.6.1-1, at least from the standpoint of such UE data being specific to media streaming services, it would make sense to define the requisite interface (i.e., between the ASP and the 5GMS AF) to support indirect data collection.



Figure 5.8.6.1-1: Indirect UE data collection  
(copied from TR 23.700-91 [8])

While in principle the M1 interface could be extended for the purpose of indirect data collection (as even suggested by SA2 in the above diagram), logically it would make more sense to define a separate (SBI-based) interface for such purpose. M1 is mainly intended to support provisioning of session-based media streaming services. Indirect UE data collection by the 5GMS AF from the ASP corresponds to a more peripheral or auxiliary feature and might be better served by defining a separate interface or API exposed by the 5GMS AF, for example R1, which is the interface name identified in the BBC *et al.* discussion paper [49], as shown in Figure 5.8.6.1-2.



Figure 5.8.6.1-2: Generic reference architecture for UE data collection and reporting  
(copied from S4-210723 [4])

It can be seen from the above diagram that R1 represents more than simply the indirect data collection procedure (referred to in Figure 5.8.6.1-2 as “indirect reporting from ASP”), by including reporting provisioning and event subscription and event publication (event exposure, e.g., via notifications) to the ASP, as further described in [49].

Note that reporting-related provisioning may pertain to the configuration of rules to be applied by the AF in the processing (e.g., anonymization, normalization, filtering, aggregation) of the data it has collected via direct or indirect methods. Such processing rules may differ for data to be reported by the AF to the ASP versus the data to be reported by the AF (via event exposure) to other NF consumers such as the NWDAF. From that perspective, when the Data Collection AF is instantiated inside a 5GMS AF, the reporting provisioning/configuration function logically belongs to M1 as part of overall provisioning functionality for 5G Media Streaming. Such extension of M1 along with additional specification of R1 is also reflected in [49] as shown in Figure 5.8.6.1-3:



Figure 5.8.6.1-3: 5GMS instantiation of generic architecture for data collection and reporting  
(copied from S4-210723 [49])

The details of R1 coupling to and decoupling from M1 should be further studied for each of the R1 functions, with particular regard for the associated procedures, resources and data structures, for example, via emulation of M1 mechanisms and/or reuse of those defined for the Naf Event Exposure service defined in TS 29.517 [25].

#### 5.8.6.2 AF Collection of CDN Access Logs

As indicated in TS 26.501, the 5GMS AS acts as a CDN server (e.g., edge server) in the hosting and delivery of streaming media content (i.e., of ingested/egested content in downlink/uplink streaming). The corresponding content hosting related information, i.e., CDN access logs, available at the 5GMS AS can be forwarded to the 5GMS AF for subsequent event exposure to consumer entities such as the NWDAF or the Application (Service) Provider. Doing so requires the specification of an interface between the 5GMS AS and 5GMS AF. Such interface is logically represented by M3, although this internal interface, intended for the exchange of content hosting related information, is not further described/specified in Rel-16 TS 26.501 and TS 26.512. As part of the present study, it would seem possible to define M3 for the transfer of CDN log information between these entities. However, as defined in TS 26.501, M3 is intended to represent an undefined interface between the 5GMS AF and 5GMS AS for the exchange of content hosting related information. Therefore, it would be more logical to define an SBI-based reporting interface across R3, as shown in Figure 5.8.6.1-3, for the transfer of CDN access logs.

#### 5.8.6.3 Candidate media-related information for Event Exposure

As indicated previously, the stage 2 and stage 3 specifications for the Naf\_EventExposure service are provided by TS 23.502 [24] and TS 29.517 [25], respectively. Up through Release 16, the categories of UE data at the AF available for event information subscription by NF consumers are the following:

- Service Experience information for an application;

- UE mobility information;

- UE communication information; and

- Exceptions information.

Comparison of the above categories with QoE metrics (as defined in TS 26.247 [40] for progressive download and DASH streaming services) and service experience and UE communication information types eligible for Naf event exposure (as defined in TS 23.288 [48]) shows the only common information attribute to be ***throughput***. “Average throughput” is a defined quality metric in TS 26.247 for both progressive download and 3GP-DASH services, and “Throughput” is a defined type of performance data from the AF in TS 23.288. However, it is not unreasonable to expect that other QoE metrics collected by the 5GMS AF (based on TS 26.247 definitions for progressive download and DASH streaming) will also be eligible for event exposure to the NWDAF. In particular, at SA2#144-e, a CR to Rel-17 TS 23.288 in S2-2103267 [50] was agreed which adds “QoE metrics” as an additional type of service data related to Service Experience information for subscription by the NWDAF from the AF. Although the identified QoE metrics in that CR references those defined by MTSI in TS 26.114 [51], a CR from Qualcomm in S2-2104496 [52] proposes the inclusion in TS 23.288 of QoE metrics defined for 3GP-DASH and progressive download (per TS 26.247 [40]), VR (per TS 26.118 [53]), MBMS (per TS 26.346 [54]) and 5GMS (per TS 26.512 [45]). Therefore, it is possible that the QoE metrics defined by the 5GMS architecture will be adopted as valid service data for NWDAF subscription to Naf\_EventExposure services in Rel-17 TS 23.288 [48] and TS 29.517 [25].

The definition of application event types, data components and formats relating to media streaming should be coordinated by SA4 with SA2 and CT3 in the production of the associated stage 2 and stage 3 specifications of Naf\_EventExposure services.

## 5.9 Per-application-authorization

### 5.9.1 Description

Operation of certain 5GMSA and 5G System enabled services include an SLA between the Application Provider and the 5GMS System provider. Different solutions to enable per-application authorization should be studied. . “Per-application authorization” refers to scenarios where one or more 5GMS-Aware Applications are hosted on the same UE (e.g. a SmartPhone) and may access services only from the associated 5GMS Application Provider.

The 5G System provider may offer one common 5GMSd AF or dedicated 5GMSd AFs. In the later case, the one 5GMSd AF instance services only a single 5GMSd Application Provider.

An example collaboration scenario is depicted below.

### 5.9.2 Collaboration Scenarios

#### 5.9.2.1 Collaboration A: UE hosting multiple Applications

This collaboration scenario focuses on cases where one or more 5GMSd-Aware Applications are hosted on the same UE and are using the same 5GMSd Client. This may be the case when the 5GMSd Client is provided as an Operating System level service. The 5GMSd Client supports isolation of the different 5GMSd-Aware Applications.



Figure 5.9.2-1: Per-Application Authorization Collaboration Scenario

Each 5GMSd-Aware Application uses an M8d reference point instance to connect to its 5GMSd Application Provider.

The 5G System provider offers a common 5GMSd AF within the trusted DN. The 5GMSd AF supports request and provider isolation so that 5GMSd Application Provider #1 and #2 are not interfering with each other. For example, 5GMSd Application Provider#1 has agreed different charging conditions than Provider #2 and the 5G System should ensure that only 5GMSd-Aware Application #1 can benefit from the conditions. Another example is different QoS levels, e.g. 5GMSd-Aware Application #1 is entitled to receive higher QoS than Application #2.

#### 5.9.2.2 Collaboration B: Applications with multiple subscription levels

This collaboration scenario focuses on cases where an Application Provider is offering multiple subscription levels to its consumers, for example 4K Premium or SD Standard QoS. This example is inspired by the use case from TS 26.512 [16], Annex A.2.

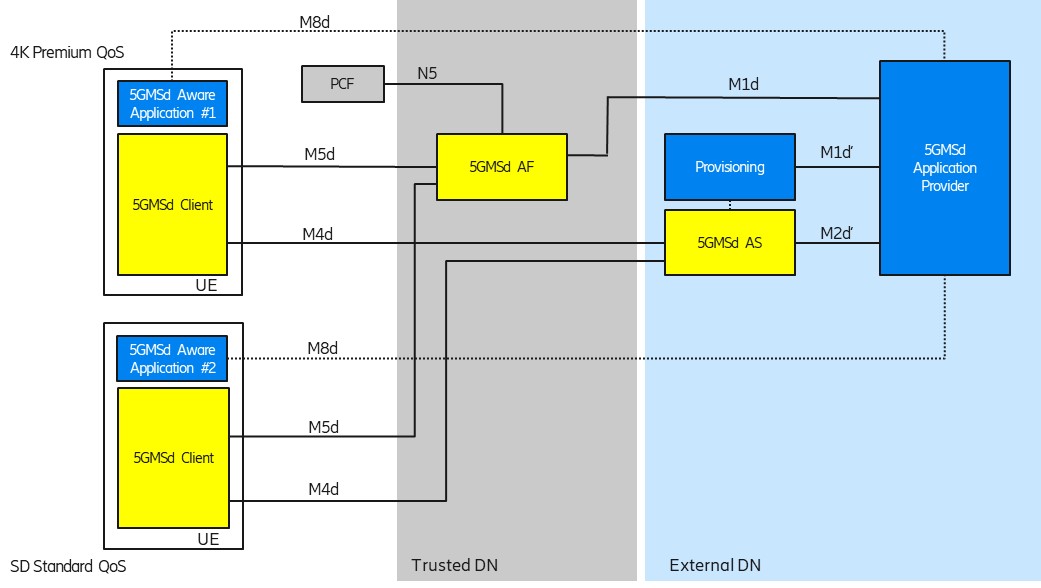


Figure 5.9.2-2: Per-Application Authorization Collaboration Scenario

Each 5GMSd-Aware Application uses an M8d reference point instance to connect to its 5GMSd Application Provider. The 5GMSd Application Provider is aware about the different subscription levels of the user.

The 5G System provider offers a common 5GMSd AF within the trusted DN. The 5GMSd AF needs to determine that 5GMSd Aware Application #1 is entitled to higher bit rates than 5GMSd-Aware Application #2.

### 5.9.3 Deployment Architectures

Editor’s Note: Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

### 5.9.4 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.

### 5.9.5 Potential open issues

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

### 5.9.6 Candidate Solutions

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

## 5.10 Support for encrypted and high-value content

### 5.10.1 Description

Content is increasingly encrypted for distribution for different reasons, e.g. Content Protection, Conditional Access, or integrity of playback. The management of keys for different use cases is a prime concern. Examples include scalable access to keys, secure storage of keys, key availabilities. It is envisioned that an MNO can provide key management and/or key distribution services for content providers. In particular, providing scalable and secure key management within 5GMS for multiple different devices needs further study.

Examples for secure media specification are for example provided by the MovieLabs ECP requirements and other content providers requirements.

In a specific example, a live sports service provider wants to offer a live stream. Examples include where the content needs to be delivered with low latency (typically encoder to glass in 3–10 seconds) in order to be on par with regular TV distribution means. Other services may also be considered.

The service may require different tools and functionalities levels of security:

1. *Conditional access supported by DRM management.* As an example, users need to get a master key for decrypting the secondary level keys.

2. *Key rotation in order to support live streaming.* As an example, these keys are changed periodically but protected by the master key.

3. *DRM and key management* to ensure playback rules, for example to avoid that clients attempting early playback of the content too early and have advantages in betting/wagering, skipping content, etc.

4. *Watermarking.* The content is distributed and a unique signature is added at the latest possible time (in the device, at the Edge). An example of such approach can be found here <https://learn.akamai.com/en-us/webhelp/adaptive-media-delivery/adaptive-media-delivery-implementation-guide/GUID-3F89E64C-415D-452D-9541-BB650CD783B9.html>.

5. *Content encryption.*

6. *A secure implementation* (use of TEE, Secure Media Path).

### 5.10.2 Collaboration Scenarios

It is assumed that the content provider provides DRM protections for the content. However, beyond this different collaboration models between the content provider and 5G System operator/MNO exist.

As examples, the MNO provides infrastructure to the content service provider in order to support security related functions.

- The service provider may want to provide scalable access to the content and in particular the key distribution. Hence it uses 5G Media streaming servers to support secure key distribution.

- The streaming service provider wants to rule playback, for example to avoid that the situation whereby users can see the streamed content too early while at the same time, the streaming service provider does not want to delay the distribution artificially either and want to give the clients the ability to download the main content (without buffer underruns).

- The service provider asks for fairness in the client, but the client cannot be trusted to act fairly. Hacked clients are possible. Clients may have DRM systems that the service providers will use.

- The service provider asks for a watermarking solution from the MNO.

Encryption (as already defined in TS 26.511 [3]) and secure keys may be used for other purposes, for example for conditional access or DRM systems. In some cases, keys are also provided in hierarchically, depending on business rules, security levels and deployment scenarios.

In an extension of the above use case, the content is distributed via multiple operators network. In this case, the encryption may be done by the service provider and the service provider provides the keys to the MNO. In another case, the service is offered by the MNO and the MNO does encryption and key management.

Editor’s Note: Study collaboration scenarios between the 5G System and Application Provider for each of the key topics.

### 5.10.3 Deployment Architectures

Editor’s Note: Based on the 5GMS Architecture, develop one or more deployment architectures that address the key topics and the collaboration models.

### 5.10.4 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.

### 5.10.5 Potential open issues

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

### 5.10.6 Candidate Solutions

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

## 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

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.

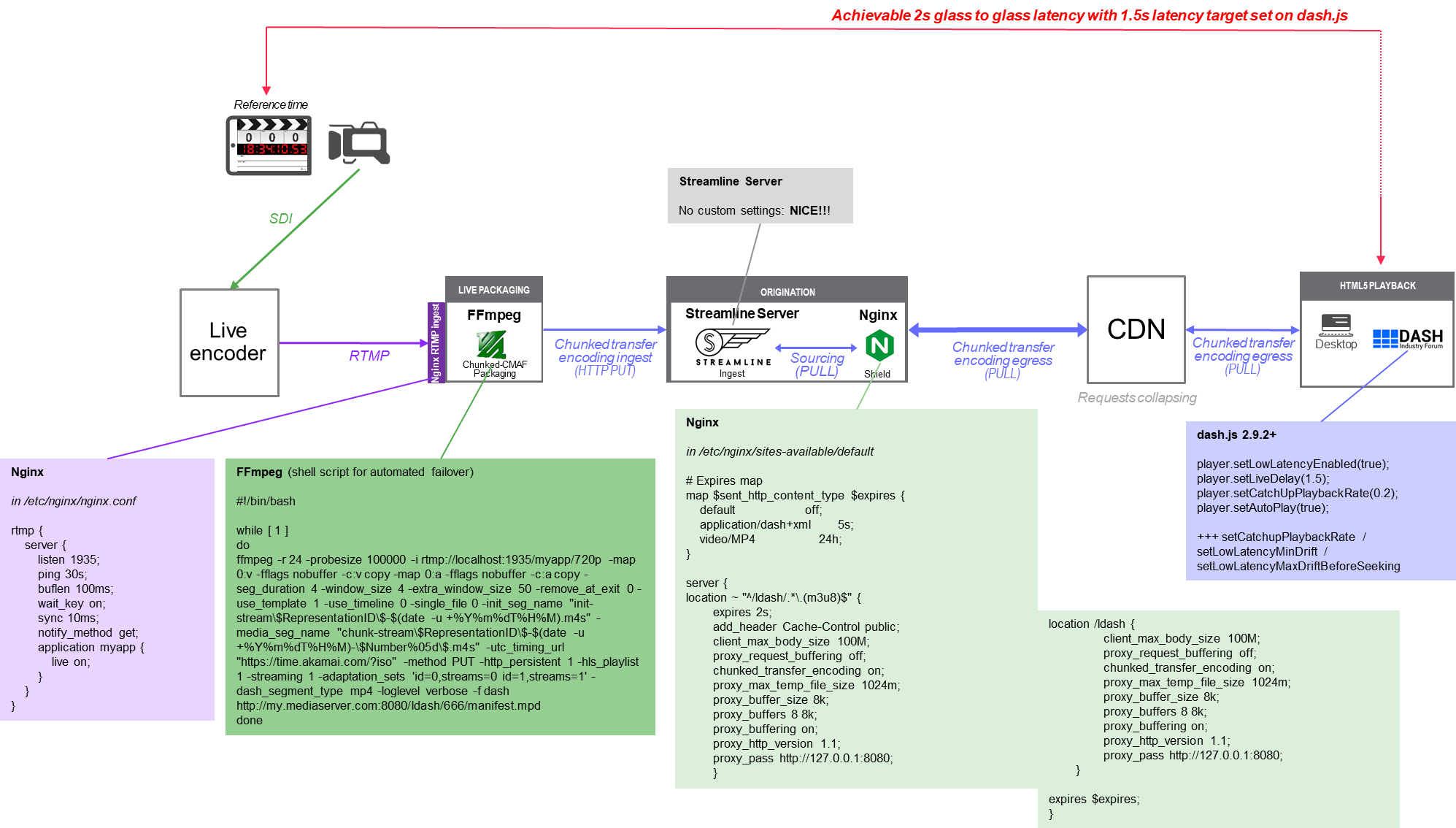


Figure 5.11.3-1 Deployment architecture for low-latency CMAF streaming

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.

### 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. Live content is encoded and packaged by the content provider (for example as low-latency CMAF) and uploaded to the MNO. The MNO may produce an MPD for its distribution. However, the content provider augments the media with production and/or encoding timestamps (e.g. producer reference times) in order to permit latency measurements.
3. The origin server is external to the MNO network and content is pulled through the 5GMS AS on demand by the clients.

#### 5.11.3.2 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.

# Annex A – Media Streaming Protocols

### A.1 Status and usage of Web Protocols

The site HTTPArchive.org [x1] offers some insights into the uptake of different HTTP protocol versions by publicly accessible websites. The Report “State of the Web” contains statistics about the number of TCP connections per page and the number of HTTP/2 requests over a time period. The site crawls millions of URLs every month. The URLs are taken from the Chrome User Experience Report.

Currently, around 70% of websites support HTTP/2. Unfortunately, the site does not show statistics for video usage.

The site quic.netray.io [x2] offers some insights into the HTTP/3 (QUIC) take-up.

### A.1.1 M4d protocol usage

It is anticipated that MPEG‑DASH would be used by many Application Providers on the M4d Interface if 5GMS services become widely deployed. MPEG‑DASH defines the manifest format and also the media segment format. MPEG‑DASH allows several different ways to use the underlying HTTP transport, depending on the DASH Profile.

For traffic identification, the identification of the transport protocol (TCP or UDP) used on interface M4d is essential, since the transport protocol needs to be described in the Service Data Flow Template. HTTP/1.1 and HTTP/2 both use TCP transport. HTTP/3 uses a UDP-based QUIC transport. Furthermore, HTTP/1.1. often leverages multiple TCP connections simultaneously, while HTTP/2 and HTTP/3 allow more efficient reuse of the transport through the technique of non-blocking request multiplexing on a single transport connection.

### A.1.2 Results of HTTP protocol version usage study

Editor’s Note: It is currently unclear how to document the results of the transport connection usage study. It is clear, this this represents only a small snapshot on how the different HTTP versions are used and currently only focused on browser based clients.

Within a small study, the transport protocol usage of three major video-on-demand providers were studied, namely YouTube, Netflix and Amazon. The study leveraged browser-based DASH players, using the popular web browsers Google Chrome (version 87.0.4280.141, 64-bit running on Win 10 Pro Version 2004 b 19041.746) and Mozilla Firefox (version 84.0.2, 64-bit running on Win 10 Pro Version 2004 b 19041.746). The intention was to get more insights into HTTP usage.

a) Accessing YouTube with Chrome, we found that YouTube in a Chrome Browser uses MPEG‑DASH with HTTP/3 transport. Several YouTube clips were selected, and HTTP/3 was consistently used for retrieving both media segments and other content. Detailed investigations showed that only a single HTTP/3 connection was established to the server.

b) Accessing Amazon Prime with Chrome, we found that Amazon Prime uses MPEG‑DASH. For some movies, HTTP/2 is used for all content (including media segments). Some other movies used HTTP/1.1 for media segments and HTTP/2 for non-media segments. It is not clear on which basis the application protocol is selected.

c) Accessing Netflix with Firefox, we found that Netflix uses MPEG‑DASH with HTTP/1.1. Some objects, such as images, are fetched using HTTP/2.

d) Accessing YouTube with Firefox, we found that YouTube uses MPEG‑DASH with HTTP/1.1. Non-video transactions use HTTP/2.

Annex <X> (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2021-02 | SA4#112-e | S4-210136 |  |  |  | Initial version | 0.0.1 |
| 2021-02 | SA4#112-e | S4-210305 |  |  |  | S4-210054: Key Topic Content Aware Streaming  S4-210056: Key Topic Per-application-authorization  S4-210298: Key Topic Additional / New transport protocols  S4-210302: Key Topic Support for encrypted and high-value content  S4-210303: Key Topic Scalable distribution of unicast Live Services | 0.1.0 |
| 2021-03 | Post SA4#112-e CCs | S4-210518 |  |  |  | S4aI201129: Key Topic Uplink media streaming  S4aI201126: Key Topic Background traffic  S4aI201134: Key Topic Content Preparation (online edited)  S4aI201154: [FS\_5GMS-EXT] Updated text for Content Preparation (only agreed parts from Clause 5.2.4)  S4aI201158: Updates on Key Topic: Network Event usage | 0.1.1 |
| 2021-04 | SA4#113-e | S4-210677 |  |  |  | S4-210637: [FS\_5GMS-EXT] Collaboration scenario proposal for KI “per-application-authorization”  S4-210636: [FS\_5GMS-EXT] Key Topic Additional / New transport protocols  S4-210640: [FS\_5GMS-EXT] Updated text for Content Preparation | 0.2.0 |
| 2021-04 | Post SA4#113-e CCs | S4-210725 |  |  |  | S4aI201160: [FS\_5GMS-EXT] Updated text for uplink streaming  S4aI201161: Editorial update to Content Preparation topic  S4aI201162: Key Topic Traffic Identification  S4aI201166: [FS\_5GMS-EXT] Updated text for Content Preparation  S4aI211193: Clarification of Traffic Identification description | 0.2.1 |
| 2021-05 | SA4#114-e | S4-210942 |  |  |  | S4-210771: Update on relevant architecture components for traffic identification  S4-210912: [FS\_5GMS-EXT] New transport protocols/Corrections and Improvements  S4-210911: [FS\_5GMS-EXT] Key Topic Scalable distribution of unicast Live Services  S4-210940: [FS\_5GMS-EXT] Candidate solution for Content Preparation format  S4-210775: Updated text on uplink streaming  S4-210943: Clarification of Traffic Identification description and addition of identified open issues  S4-210917: [FS\_5GMS-EXT] Updated text for uplink streaming  S4-210918: [FS\_5GMS-EXT] Updated text for uplink streaming: additional gap analysis | 0.3.0 |
| 2021-06 | SA4#114-e | S4-210960 |  |  |  | S4-210964: Interfaces and Formats for AF Data Collection and Event Exposure | 0.4.0 |
| 2021-06 | Post SA4#114-e |  |  |  |  | Editorial Corrections  Correction of S4-210918 implementation  Change of 5G logo. | 0.4.1 |
| 2021-08 | SA4#115-e | S4-21xxxx |  |  |  | S4aI211196: [FS\_5GMS\_EXT] Content preparation gap analysis: address translation | 0.5.0 |