## **3GPP SA4 MBS SWG Post SA4#129-eS4aI240160**

**Online, 10th October 2024** *revision of S4aI240145*

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.3* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** |  | **rev** | **18** | **Current version:** |  |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

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

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | [FS\_AMD ] WT 03a: Multi-CDN and Multi-Access Media Delivery | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** |  | | | | | | | | | |
| ***Source to TSG:*** | S4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** |  | | | | |  | ***Date:*** | | | 2024-05-23 |
|  |  | | | |  | |  | | |  |
| ***Category:*** |  |  | | | | | ***Release:*** | | |  |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-17 (Release 17) Rel-18 (Release 18) Rel-19 (Release 19)  Rel-20 (Release 20)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Document the FS\_AMD key topic “c) Multi-Access and Multi-CDN Delivery” description and collaboration scenarios. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Addition of new clause 5.x Multi-CDN and Multi-Access Media Delivery including sub-clause structure and headings. Addition of prose for Description and Collaboration Scenarios. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | FS\_AMD objectives not achieved. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 2, 5.x (NEW) | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | |  | | |
| ***affected:*** | |  | **X** | Test specifications | | | |  | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | |  | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | | S4-240591: SA4#127-bis-e, description and collaboration scenarios  S4-240844: updates with comments received at SA4#127-e-bis  S4aI240052: updated to match proposed template and address pending comments.  S4al240894: S4al240052 endorsed at SA4 post 127-bis-e and comments/changes accepted for ease of reading. Updates with comments received at SA4 post 127-bis-e and introduction of candidate solutions. More background on candidate solution is contained in S4-240895. Multi-access media delivery overview and collaboration scenarios have been moved to S4-241082.  S4-241230: updates to address pending comments.  S4-241273: updates to address pending comments.  S4al240095: S4-241273 endorsed at SA4#128. Comments/changes accepted for ease of reading. CMMF candidate solution architecture and reference point descriptions added.  S4al240107: CMMF candidate solution architecture and reference point descriptions replaced with options to incorporate CMMF within the existing 5GMS architecture.  S4al240114: Edits from BBC. Endorsed by MBS SWG.  S4-241588: Addition of CMMF background information and addresses outstanding comments.  S4-241665: Reorginization of content to fit within the existing template and updates to address received comments.  S4aI240120: updates to CMMF architecture and high-level call flows.  S4aI240138: Updates to CMMF’s introduction to include more details concerning the expected overhead CMMF induces as well as an example architecture and call flow used to deliver CMMF content for the trial where the included performance metrics were collected. Further updates to the CMMF architecture and high-level call flows to address outstanding comments.  S4al240145: Incorporates edits and captures additional BBC comments.  S4aI240160**:** Updates to CMMF’s architecture mapping and high-level flowcall sections. Addition of CMMF gap analysis. | | | | | | | | |

## FIRST CHANGE

## 2 References

…

[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

…

[15] 3GPP TS 26.501: "5G Media Streaming (5GMS) General description and architecture (Release 18)".

[16] 3GPP TS 26.512: "5G Media streaming (5GMS) protocols (Release 18)".

…

[32] IETF RFC 9000: "QUIC: A UDP-Based Multiplexed and Secure Transport", May 2021.

…

[DASH9] Draft Text of ISO/IEC FDIS 23009-9: "Information technology - Dynamic adaptive streaming over HTTP (DASH) – Part 9: Redundant Encoding and Packaging for segmented live media (REaP)", ISO/IEC JTC 1/SC 29/WG 3 NO 1165, Jan. 26, 2024. [Online]. Available: https://www.mpeg.org/standards/MPEG-DASH/9/

[UNPKG24] Emma Roth, "A popular open-source content delivery network went down for hours", The Verge, Apr. 12, 2024. [Online]. Available: https://www.theverge.com/2024/4/12/24128276/open-source-unpkg-cdn-down (accessed May 9, 2024).

[NET23] Sebastian Moss, "Cloudflare recovers from service outage after power failure at core North American data center", Data Center Dynamics, Nov. 3, 2023. [Online]. Available: https://www.datacenterdynamics.com/en/news/cloudflare-recovers-from-service-outage-after-power-failure-at-core-north-american-data-center/ (accessed May 9, 2024).

[FSLY21] Brian Barrett, "How an Obscure Company Took Down Big Chunks of the Internet", Wired, Jun. 8, 2021. [Online]. Available: https://www.wired.com/story/fastly-cdn-internet-outages-2021/ (accessed May 9, 2024).

[AKAM21] Josh Fomon, "CDN Provider Akamai Takes Down Popular Internet Services During Outage", Ookla, Jul. 22, 2021. [Online]. Available: https://www.ookla.com/articles/akamai-outage-july-22-2021 (accessed May 9, 2024).

[NET22] Charlotte Trueman, "Cloudflare outage brings hundreds of sites, services temporarily offline", Computer World, Jun. 21, 2022. [Online]. Available: https://www.computerworld.com/article/1627967/cloudflare-outage-brings-hundreds-of-sites-services-temporarily-offline.html (accessed May 9, 2024).

[VZ19] Jim Salter, "The Internet broke today: Facebook, Verizon, and more see major outages", Ars Technica, Jul. 3, 2019. [Online]. Available: https://arstechnica.com/information-technology/2019/07/facebook-cloudflare-microsoft-and-twitter-suffer-outages/ (accessed May 9, 2024).

[DEMX01] Marc Hoppner, "A content owner, a CDN and a player walk into a bar", (Jan. 6, 2023). Accessed: May 9, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=S9EdoQFOQ9I&list  
=PLkyaYNWEKcOf98lZxnCcL6y7ZIVU3oSYO&index=12

[DEMX02] Guillaume du Pantavice, "Improving streaming experience with Bayesian optimization, from AB to AZ test", (Dec. 25, 2021). Accessed: May 9, 2024. [Online Video]. Available: https://www.youtube.com/  
watch?v=t4nRrLygVwo&list=PLkyaYNWEKcOfD1GYFxFbZXDP03XM-cZPg&index=19

[IEEE01] E. Ghabashneh and S. Rao, "Exploring the interplay between CDN caching and video streaming performance", IEEE INFOCOM 2020 – IEEE Conference on Computer Communications, Toronto, ON, Canada, 2020, pp. 516-525.

[ACM01] K. Vermeulen, L. Salamatian, S. H. Kim, M. Calder, and E. Katz-Bassett, "The central problem with distributed content: common CDN deployments centralize traffic in a risky way", In Proceedings of the 22nd ACM Workshop on Hot Topics in Networks (HotNets ’23). Association for Computing Machinery, New York, NY, USA, 70-78.

[MHV01] A. Bentaleb, R. Farahani, F. Tashtarian, C. Timmerer, H. Hellwagner, and R. Zimmermann, "Which CDN to Download From? A Client and Server Strategies", (Jan. 6, 2024). Accessed: May 9, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=xCZmCnWgQRE

[VAS01] Will Law, "Content steering with MPEG DASH", (May 4, 2023). Accessed: May 9, 2024. [Online Video]. Available: https://www.youtube.com/watch?v=o9Pa5y-Usxw

[MWS23] W. Law and Y. Reznik, "DASH content steering at scale", Media Web Symposium (MWS’23), June 2023.

[DIFCS] ETSI TS 103 998: "DASH-IF: Content steering for DASH".

[CMMF] ETSI TS 103 973: "Coded multisource media format (CMMF) for content distribution and delivery".

[TS26510] 3GPP TS 26.510: "Interactions and APIs for provisioning and media session handling (Release 18)".

[RAPTORQ] IETF RFC 6330: "RaptorQ forward error correction scheme for object delivery", August 2011.

[RSFEC] IETF RFC 5110: "Reed-Solomon forward error correction (FEC) schemes", April 2009.

## SECOND CHANGE

## 5.19 Multi-CDN media delivery

### 5.19.1 Description

#### 5.19.1.1 Introduction

Media streaming applications conventionally obtain content from a single source or endpoint over a single path within a network. This imposes several limitations:

1. Performance is constrained to that of the source/endpoint and path chosen. Whatever the limits on network bandwidth and latency between the client and that source/endpoint are directly translated to the client’s achievable Quality of Service (QoS) and Quality of Experience (QoE).

2 Disruptions or degraded performance caused by the source/endpoint in use or on any of the network links between the client and source/endpoint can lead to poor user experience, often in the form of lower playback quality, rebuffering, or complete playback failure.

This study considers integration of different technologies into the 5G Media Streaming System that addresses these, and similar, issues by allowing media streaming applications to efficiently access content across multiple Content Delivery Networks (CDNs) and/or multiple access networks. Different client implementations may then beneficially use the content on these multiple sources or networks concurrently, potentially guided by service or network provider. In addition, formats and techniques for generating content for multiple CDN or multiple access network delivery such as MPEG-DASH Part 9 (ReAP) [DASH9] may be considered. Further extensions include the ability for a client to use multiple access networks at the same time to support media delivery. Study of integration of different technologies into the 5G Media Streaming System is of relevance to address content provisioning, content hosting, impacts on user plane reference points M2 and M4, and on media session handling at reference point M5 as well as potential benefits in terms of quality and resource usage.

#### 5.19.1.2 Challenges Multi-CDN deployments aim to address

CDNs are often used by content distributors to globally scale delivery of their content to end-users. These networks consist of a number of Points of Presence (PoPs) located at various locations around the networks’ edge. These PoPs help load-balance delivery of content as well as improve Quality of Service (QoS) by reducing the distance/latency between every client and the content they are accessing. In many cases, content distributors employ multiple CDNs to leverage the strengths of one CDN over another in every location those CDNs have a PoP. For example, a client experiencing degraded performance while using one CDN may switch to another that is offering better performance at that time and location. As another example, a content distributor may prefer one CDN over another at a given time to reduce delivery costs and/or meet monthly contractual commitments. These multi-CDN deployments aim to solve content delivery issues that exist when only one CDN is used; but the benefits they provide may not be fully realized because of the various challenges experienced and underlying methods used to stream content to every client.

Challenges multi-CDN deployments and architectures aim to address may include:

1. *Sustained CDN-/network-wide service disruptions* where network access, connectivity or QoS is severely degraded. Examples may include cases where an entire CDN’s network is degraded because of a network-wide misconfiguration or power failure. The duration of these events may last minutes to hours and affect a majority of the client population. Examples of recorded instances can be found in [UNPKG24], [NET23], [FSLY21], [AKAM21], [NET22], and [VZ19].

2. *Intermittent or short-term disruptions affecting QoS for an individual or small group of clients.* Examples include short periods of congestion within the network, isolated HTTP request/response failures or delays caused by application server congestion, etc. The following discusses these in greater detail: [DEMX01], [DEMX02], [IEEE01], [ACM01], [MHV01], [VAS01], and [MWS23]

3. *Augmentation of one CDN's performance with that of another to achieve a level of performance that neither can provide on its own.* An example is a peer-to-peer CDN where each peer has limited uplink capacity and is unable to satisfactorily service client demand on its own.

#### 5.19.1.3 Coded Multi-source Media Format (CMMF)

##### 5.19.1.3.1 Functional description of CMMF

Coded Multi-source Media Format (CMMF) [CMMF] is an extensible container format designed to facilitate the management and interchange of audio-visual media and metadata in one or more coded representations (e.g., encoded with application-layer, forward error correction (FEC), linear, network, or channel codes). The coded media representations supported by CMMF enable the efficient use of multi-source, multi-path, and multi-access connectivity for network-delivered media applications. The use of CMMF does not replace the basic media streaming architectures and procedures already defined. Rather, it is intended to supplement them to provide additional capabilities.

A typical Video-on-Demand (VoD) MPEG-DASH HTTP adaptive streaming system is set up similarly to the non-shaded blocks shown in figure 5.19.1.3.1-1. Source media (e.g. audio/video elementary streams) are segmented and encoded into multiple representations, each with a different quality and bit rate. These segments are packaged together using MPEG-DASH (or HLS) and stored on an origin server located within the network. One or more CDNs are set up to distribute and deliver this content to an OTT service provider’s customer base. These CDNs obtain every requested MPEG-DASH (or HLS) segment from an origin server, caches these segments at their respective network edges, and deliver these segments to clients.

Enabling multisource delivery using CMMF within this existing delivery architecture can be accomplished through the addition of a CMMF Bitstream Generator/Source before segments are delivered to the CDNs, and a CMMF Receiver on each client, as illustrated by the shaded boxes in figure 5.19.1.3.1-1.

A diagram of a diagram of a computer

Description automatically generated

Figure 5.19.1.3.1-1: MPEG-DASH with CMMF Delivery System Example

While Figure 5.19.1.3.1-1 shows the CMMF Bitstream Generator/Source between the Origin Server and the CDNs (i.e., CMMF bitstreams are created on demand), the CMMF Bitstream Generator/Source can just as easily be located between the MPEG-DASH Packager and the Origin Server. In the former case, the original MPEG-DASH segments are stored on the Origin Server and CMMF representations of those segments are cached on each CDN. In the latter case, the CMMF Bitstream Generator/Source creates multiple CMMF representations of each segment produced by the MPEG-DASH Packager and stores them on the Origin Server for later retrieval by a CDN.

In this system, the Application Service Provider is responsible for segmenting, encoding, and packaging the media. It is also responsible for creating a presentation manifest (MPD, shown in figure 15.19.1.3.1-2) that contains relative URLs to the files/segments (shown in a dotted outline) that make up the adaptation sets. This information is transferred from the Application Service Provider to the media player (using a method preferred by the Application Service Provider) and is used by the media player to determine which segments are to be downloaded (via the CMMF Receiver) and played.

A diagram of a data flow

Description automatically generated with medium confidence

Figure 15.19.1.3.1-2: Example MPEG-DASH MPD

The Application Service Provider in this example is also responsible for determining and setting up the delivery system (i.e., the CDNs) to distribute CMMF-encoded media. Information about this setup is captured within the CMMF Configuration Information as a list of host URLs to each of the CDNs. It is important to note that both the MPEG‑DASH master manifest and the CMMF Configuration Information is required to download the media. In this example, it is assumed that the Application Service Provider is utilizing two CDNs and each segment listed in the MPEG-DASH MPD is encoded and packaged into two unique CMMF bitstreams/objects, one intended for the first CDN and the other for the second CDN according to the list of host URLs provided in the CMMF Configuration Information. An example of this is shown in figure 15.19.1.3.1-3. Furthermore, example bitstream/object constructions can be found in annex C of [CMMF].

A diagram of a computer

Description automatically generated

Figure 15.19.1.3.1-3: CMMF bitstreams/objects generated  
to deliver the MPEG-DASH packaged content

The process for streaming this content is shown in figure 15.19.1.3.1-4. An Application Service Provider first configures and provisions the coded multi-source (or CMMF) service. The first steps are configuration of multiple CDNs. This configuration includes defining each CDNs’ service domain names, origin host (e.g., the domain name of the CMMF Bitstream Generator/Source), caching policies, etc. The Application Service Provider also configures the CMMF Bitstream Generator/Source. This configuration includes the definition of necessary information for the CMMF Bitstream Generator/Source to properly encode and generate CMMF objects. Per [CMMF], this may include: the definition of a code type and code construction parameters, definition of a CMMF profile, information about source object construction (e.g., fragmented HLS/MPEG-DASH segments, etc.), the location where source media can be found, etc. The final step of this process is to publish MPEG-DASH and/or HLS content to an origin server (e.g., AWS S3 bucket).

Upon client initialization, the CMMF Receiver requests the CMMF Configuration Information (in this case it is stored as a JSON file) from the Application Service Provider. This information communicates a list of host URLs specifying the locations for which the client can retrieve CMMF encoded content for every segment listed within the master manifest file. In this case, this list includes the domain names used during the CDN configuration step.

Upon media player/delivery session initialization, the media player retrieves and parses the presentation manifest (MPD) from the Application Service Provider and chooses the adaptation set(s) it wishes to stream. Once selected, the media player requests the appropriate segment from the CMMF Receiver using the relative URL communicated in the master manifest. The CMMF Receiver joins the relative URL with each of the host URLs from the CMMF Configuration Information file and requests two distinct CMMF bitstreams of that segment from the two available CDNs. Assuming the appropriate CMMF bitstream is cached, the CDN begins delivery. Otherwise, the CDN requests the segment from the CMMF Bitstream Generator/Source. At which point, the original segment is pulled from storage, encoded, and delivered to the CDN and to the client. The CMMF Receiver downloads the two CMMF bitstreams of that segment until it is capable of decoding. These downloads are performed using an appropriate strategy based on the underlying network protocols in use and network conditions. Once enough CMMF-encoded content has been received, the CMMF Receiver decodes the segment and delivers it to the media player. The media player selects the next segment to be downloaded and the process repeats.

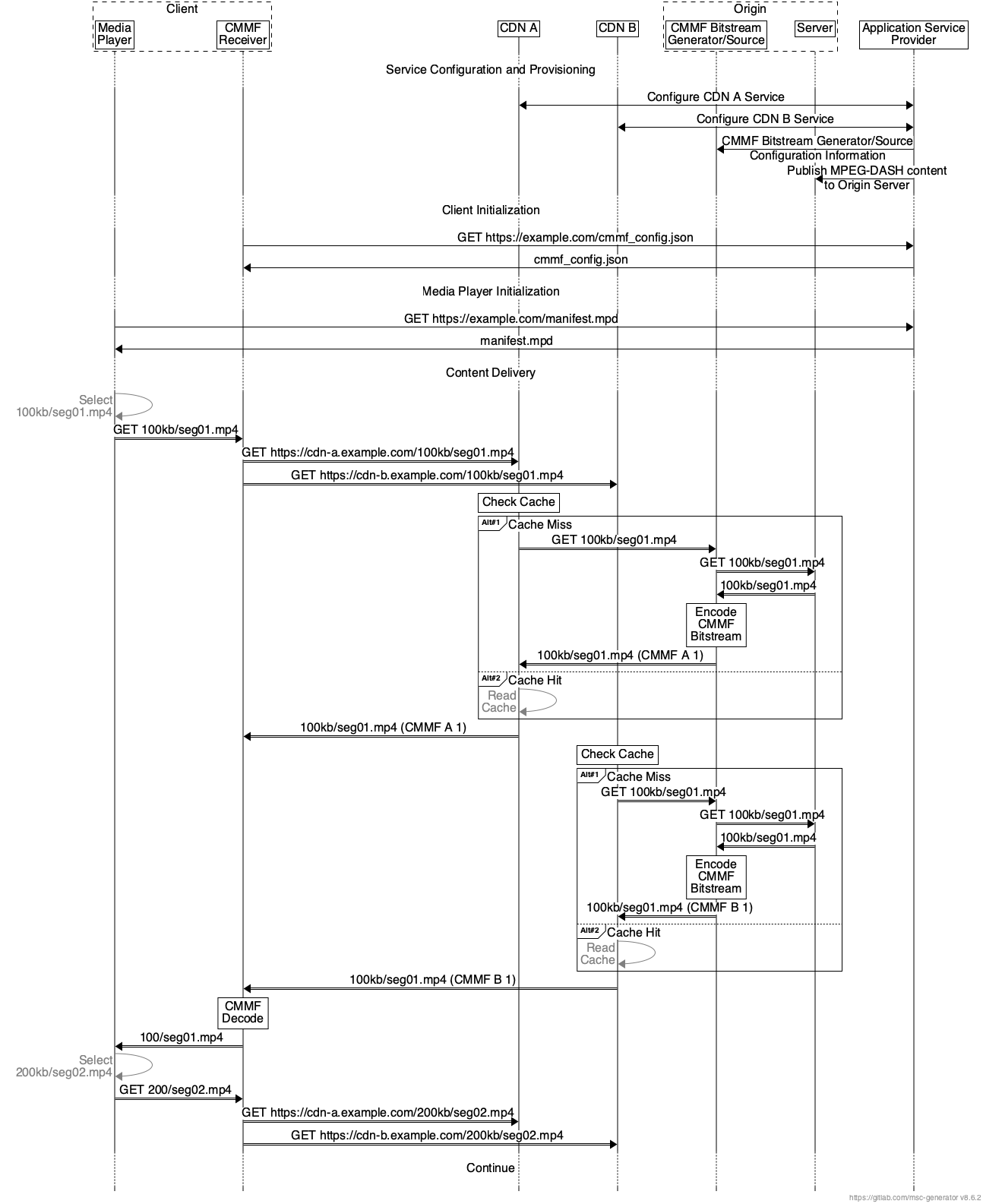


Figure 15.19.1.3.1-4: CMMF request and content delivery example for MPEG-DASH

In the case of a third CDN being introduced, a new CMMF bitstream can be generated and cached without replacing or modifying the existing CMMF bitstreams already cached in the initial two CDNs. All that is required is an update to the host URL list managed by the Application Service Provider.

##### 5.19.1.3.2 CMMF requirements on multi-CDN delivery

Based on the description in the previous clause, multi-CDN and/or multi-access media delivery using CMMF requires preparation of the content to be served to a population of clients and clients that can access and download from multiple sources in parallel. Specifically,

1. The ability to create CMMF-encoded media objects and distribute/stripe these (in addition to possibly distributing the original source media such as MPEG-DASH or HLS media segments) across multiple client-accessible network locations (e.g., 5GMS Application Servers, CDNs, etc.).

Accessing content from multiple sources/endpoints within the network simultaneously requires that each network source/endpoint be populated with a unique CMMF bitstream/object containing the content being requested. A CMMF network source/endpoint is one that can be individually addressable or reachable (i.e., it is recommended that there exists a one-to-one mapping between the set of individually addressable or reachable sources/endpoints and the set of CMMF bitstreams/objects for each CMMF encoded piece of content). Source/endpoint types may be entire CDN distributions, single points-of-presence (PoPs) within a single CDN distribution, or standalone servers. For example, a single CDN which replicates content across their PoPs and uses DNS or anycast to route traffic to the PoPs within their network would be considered one source/endpoint. Alternatively, a CDN that enables clients to reach individual PoPs within their network may allow for each PoP to be an CMMF source/endpoint assuming each PoP can be populated with a unique CMMF bitstream/object.

Various methods for creating unique CMMF bitstreams/objects for each CMMF network source/endpoint exist. The necessary CMMF bitstreams/objects can be created offline (e.g., at the time the video/audio is encoded and packaged) and stored on an origin server for later retrieval by the CMMF network sources/endpoint. They can also be created on demand using a cloud-based or edge-based just-in-time encoder as client requests are received.

Detailed examples for preparing original source media for delivery from multiple serving endpoints using CMMF are provided in [CMMF]. In general, the processing required to create CMMF bitstreams/objects is minimal (and scalable) allowing for a wide range of available implementation options.

2. The capability for clients to access, efficiently download, and decode information collected from the partially received CMMF-encoded media bitstreams/objects obtained in parallel from multiple network endpoints where CMMF-encoded media objects (and possibly original source media) are stored/cached.

These capabilities can be implemented as a plug-in similar to the MBMS Client shown in clause 4.6.1 of [TS26510] to simplify integration into existing platforms and players, or they can be implemented directly within the streaming media player located on each client. When downloading content (e.g., a segment that is intended to be played), a CMMF client will connect to multiple sources/endpoints and request the CMMF bitstream/object associated with that content from each. Any one of these CMMF bitstreams/objects do not need to be obtained in their entirety, nor does any byte-level scheduling need to occur (e.g., each CMMF bitstream/object can be transmitted from their beginning to their end). Rather, a client only needs to obtain enough information from all of the transmitted CMMF bitstreams/objects so that it can decode the content those bitstreams/objects carry.

The methods used to efficiently download media using CMMF from multiple sources/endpoints are heavily dependent on the underlying network and transport protocols used to deliver CMMF-encoded bitstreams/objects, as well as the implementation of the CMMF-enabled client. For streaming use cases utilizing either HTTP/1.1 [3] or HTTP/2 [4] over TCP, total overhead (i.e., total amount of data egressed from all of the sources/endpoints (including HTTP and CMMF headers) with respect to the size of the original source media object) has been demonstrated to be between 0.5-3% when downloading 1080p, 4-second MPEG-DASH segments encoded at approximately 5 Mbps (it should be highlighted that this is roughly on-par with the overhead induced by in-market media players related to downloading multiple bit rates of the same segment and (obviously) rendering only one during playback). Overhead when using other network/transport protocols or different implementations may differ.

Likewise, the number of requests sent by a CMMF-enabled client to each CMMF endpoint can also be considered overhead. Various strategies can be employed depending on the capabilities or limitations of the underlying transport protocols and network. Cases where the client is limited (e.g., it is using HTTP/1.1 [3]to obtain CMMF-encoded content where the client can only make requests serially and cancel them by closing the underlying TCP connection) may require a different strategy than cases where a more advanced protocol is used (e.g., HTTP/2 [4]).

- In the case where HTTP/1.1 is used as the application protocol, one strategy to download CMMF-encoded content while limiting the amount of extra data downloaded as well as avoiding cancelling requests may be to make several byte range requests to each CMMF endpoint where these byte range requests are sized appropriately for the observed network conditions and the amount of data that is still required at that time. A current, state-of-the-art implementation of this strategy generates between three and six requests to each CMMF endpoint (depending on the network conditions) while downloading 1080p, 4-second MPEG-DASH segments encoded at approximately 5 Mbps.

- In cases where the client is using an underlying protocol suite with more features (e.g., HTTP/2 over TCP or HTTP/3 [5] over QUIC [32]), those features may allow for different strategies. A strategy that uses flow control features available within HTTP/2 achieves the same download as mentioned above while only sending one request to each CMMF endpoint.

Of course, other strategies exist, and the approach they take to obtain CMMF-encoded content from multiple CMMF endpoints is dependent on the network protocols used, the underlying networks’ capabilities, and the overall use case.

The complexity and client device impacts of decoding received CMMF bitstreams/objects has also been demonstrated to be minimal. While the decode complexity is dependent on the CMMF code type used (CMMF [CMMF] supports a variety of different code types including general deterministic and random linear codes (RLC), the 3GPP Raptor code specified in RFC 5053 [RFC5053] as defined in TS 26.346 [26346], RaptorQ as specified in RFC 6330 [RAPTORQ], and Reed-Solomon as specified in RFC 5110 [RSFEC]), CMMF has been demonstrated on over 4,000 unique client device models without issue.

##### 5.19.1.3.3 CMMF performance

To understand some of the advantages of using CMMF for streaming media, CMMF was implemented and trialed on a commercial streaming platform from approximately September 2022 through September 2023. This platform offers a large content library, streamed to a world-wide customer base where the majority of the content had a maximum bit rate of 5 Mbps or less (the median maximum bit rate available was approximately 3.5 Mbps while over 70% of all sessions had a maximum possible bit rate of 5 Mbps or less). Approximately 5% - 50% of the traffic on selected device types was streamed using CMMF while the remainder of the traffic was streamed using a popular conventional server-side switching/DNS-based multi-CDN implementation. Both the CMMF multi-source and the conventional multi-CDN approach used three tier 1 CDNs. CMMF clients downloaded content from each CDN in parallel, while the "conventional" clients switched between the three based on input from the multi-CDN switching platform. Performance measurements for all traffic were collected using an industry-leading performance measurement platform. This data includes session-level information about relevant QoE key performance indicators (KPIs). In addition, supplemental QoS information was collected for only those sessions using CMMF multi-source as a delivery method.

A summary of the amount of traffic measured for each delivery method during this trial is provided in table 5.19.1.3.3-1. This and subsequent tables only show traffic measured for Android clients streaming over cellular networks from January 1 through July 26, 2023. Furthermore, only those sessions where the mean edge cache hit rate is greater than 50% are considered. For CMMF traffic, this was determined using the supplemental QoS information collected for each CMMF session. For conventional traffic, this supplemental information was not available on a session-by-session basis since this traffic bypassed the functionality added when implementing CMMF within the player. Rather, it was confirmed, via querying each CDN utilized, that the mean edge cache hit rates for all conventional traffic was greater than 95%. This estimate of the edge cache hit rate was also validated in a separate experiment where conventional traffic was routed through the new CMMF player functionality so that QoS metrics (including cache hit status) could be collected. Unfortunately, the volume of CMMF traffic and the diversity of the content streamed during the trial made it very difficult to keep CDN caches warm with CMMF encoded content. Trying to match multi-source and conventional edge cache hit rates on a one-to-one basis was not possible. As a result, the threshold established above provides sufficient data to provide statistically significant results; but it also implicitly favors conventional delivery since those sessions were more often served by the CDNs’ edge.

Table 5.19.1.3.3-1: CMMF real-world multi-CDN trial summary. Only sessions measured on cellular networks and running Android are shown

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Delivery method | Hours watched | Number of playback sessions | Number of unique devices | Number of unique countries | Minutes watched per unique device |
| Conventional | 25,026.92 | 120,269 | 23,752 | 178 | 63.22 |
| CMMF | 14,013.76 | 44,081 | 12,534 | 141 | 67.08 |

An overview of the performance improvements multi-source delivery provided over conventional multi-CDN switching for various QoE KPI’s is shown in table 5.19.1.3.3-2. The table provides the mean value of the relevant KPI plus/minus one standard deviation. In general, double-digit gains were observed across all key QoE performance indicators showing that CMMF enabled multi-source delivery can drastically improve the quality of streamed media.

Table 5.19.1.3.3-2: Real-world multi-CDN QoE performance results

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Delivery method | Normalized average session bit rate (% of max session bit rate)  (NOTE 1) | Playback start-up time (s)  (NOTE 2) | Connection-induced rebuffering ratio (%)  (NOTE 3) | Start-up failure rate (%) | Playback failure rate (%) |
| Conventional | 83.70 ± 28.08 | 3.40 ± 10.08 | 0.28 ± 1.78 | 0.51 | 1.22 |
| CMMF | 94.31 ± 16.23 | 1.83 ± 9.34 | 0.19 ± 1.17 | 0.07 | 0.59 |
| Difference | 10.61 ± 32.43 | -1.57 ± 13.74 | -0.09 ± 2.13 | -0.44 | -0.63 |
| Gain | + 12.68% | - 46.18% | - 32.14% | -86.27 % | - 51.64% |
| NOTE 1: The normalized average session bit rate is defined as the average bit rate measured during a session divided by the maximum bit rate listed in the session’s content manifest. Only sessions without a start-up or video playback failure, a playing time greater than or equal to 60 seconds, playback completed at least 10% of the content, and the maximum bit rate as defined by the sessions’ corresponding manifest was available. These statistics are weighted using the sessions’ duration.  NOTE 2: Only sessions without a start-up or playback failure and a start-up time greater than 0 seconds.  NOTE 3: Only sessions without a start-up or playback failure, a playing time greater than or equal to 60 seconds, and playback completed at least 10% of the content. These statistics are weighted using the sessions’ duration. | | | | | |

The empirical CDFs for the content normalized average session bit rate, start-up time, and connection-induced rebuffing ratio are provided in figures 5.19.1.3.3-1, 5.19.1.3-2, and 5.19.1.3.3-3 respectively.

Figure 5.19.1.3.3-1 shows that 60% of the sessions, regardless of delivery method, experienced an average session playback bit rate close to the maximum possible based on the content being played. However, CMMF multi-source delivery was able to lift more of those clients that could not reach the highest bit rate further up the bit rate ladder than conventional delivery.

A graph showing a blue and red line

Description automatically generated

Figure 5.19.1.3.3-1: Empirical CDF of the content normalized average session bit rate.

Figure 5.19.1.3.3-2 shows that CMMF multi-source delivery was able to significantly reduce the playback start-up time as well. For example, only 10.4% of the CMMF sessions experienced a startup time greater than 3 seconds compared to 29.0% of the conventional sessions.

A graph of a line graph

Description automatically generated with medium confidence

Figure 5.19.1.3.3-2: Empirical CDF of the video startup time.

Finally, figure 5.19.1.3.3-3 shows that CMMF multi-source delivery reduced the number of sessions that experienced a connection-induced rebuffering event from 22.4% to 14.9% in addition to reducing the total duration of rebuffering given a rebuffering event occurred.

A graph showing the value of a product

Description automatically generated with medium confidence

Figure 5.19.1.3.3-3: Empirical CDF of the connection-induced rebuffering ratio (CIRR).

These results were collected using a system similar to that presented in clause 15.19.1.3.1.

### 5.19.2 Collaboration scenarios

#### 5.19.2.1 Multi-CDN media delivery

In this scenario, the 5GMSd Client requests adaptive media streaming content from two or more 5GMSd Application Servers. The Client may choose one 5GMSd AS or use multiple simultaneously. This allows the client to distribute network load across Application Servers and M4 downlink transports, optimize costs, as well as improve QoS.

The client’s Media Session Handler discovers the URLs of these Application Servers from the 5GMSd Application Function (AF), either through a Media Entry Point or from a separate piece of metadata. QoE metrics from the client may be used by the AF to determine the best Application Server(s) for each client to use when streaming media.

Figure 5.19.2.1-1 shows the client communicating with multiple Application Servers. Each AS has no direct communication with its peers; rather it communicates (minimally) with the Application Provider and with the 5GMSd AF (not depicted) via reference point M3d.

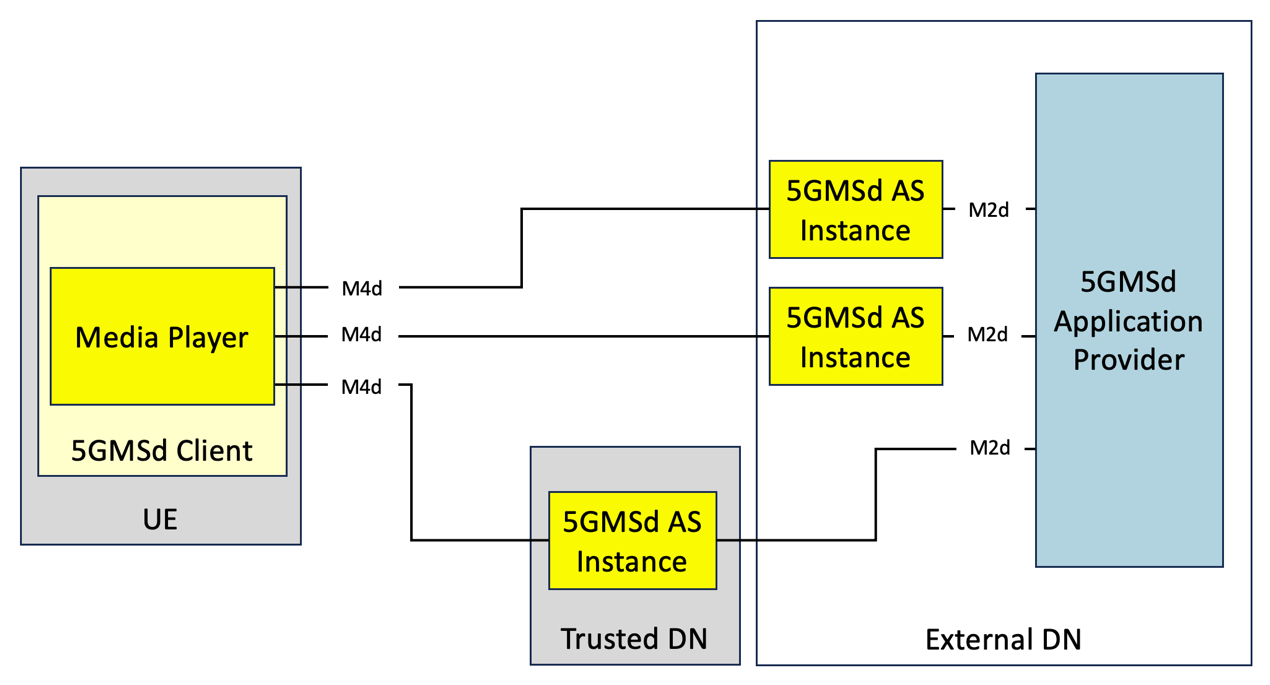


Figure 5.19.2.1-1: Multi-CDN media delivery within 5G system

#### 5.19.2.2 Joint multi-CDN and multi-access media delivery

In this scenario, the 5GMSd Client is directly connected to multiple data, or access, networks (e.g., an unmanaged Wi‑Fi network and the 5G network), as described in clause 5.18. The client requests adaptive media streaming content from two or more 5GMSd Application Servers. The Client may choose one or use multiple simultaneously. This allows the client to distribute network load across access networks and Application Servers, optimize costs, as well as improve QoS.

The client’s Media Session Handler discovers the URLs of these Application Servers from the 5GMSd Application Function (AF), either through a Media Entry Point or from a separate piece of metadata. QoE metrics from the client may be used by the AF to determine the best Application Server(s) for each client to use when streaming media.

Figure 5.19.2.2-1 shows the client communicating with multiple Application Servers through different data networks. Neither data network nor AS has direct communication with its peers. Rather each 5GMSd AS communicates (minimally) with the 5GMSd Application Provider at reference point M2 and with the 5GMSd AF (not depicted) via reference point M3d.

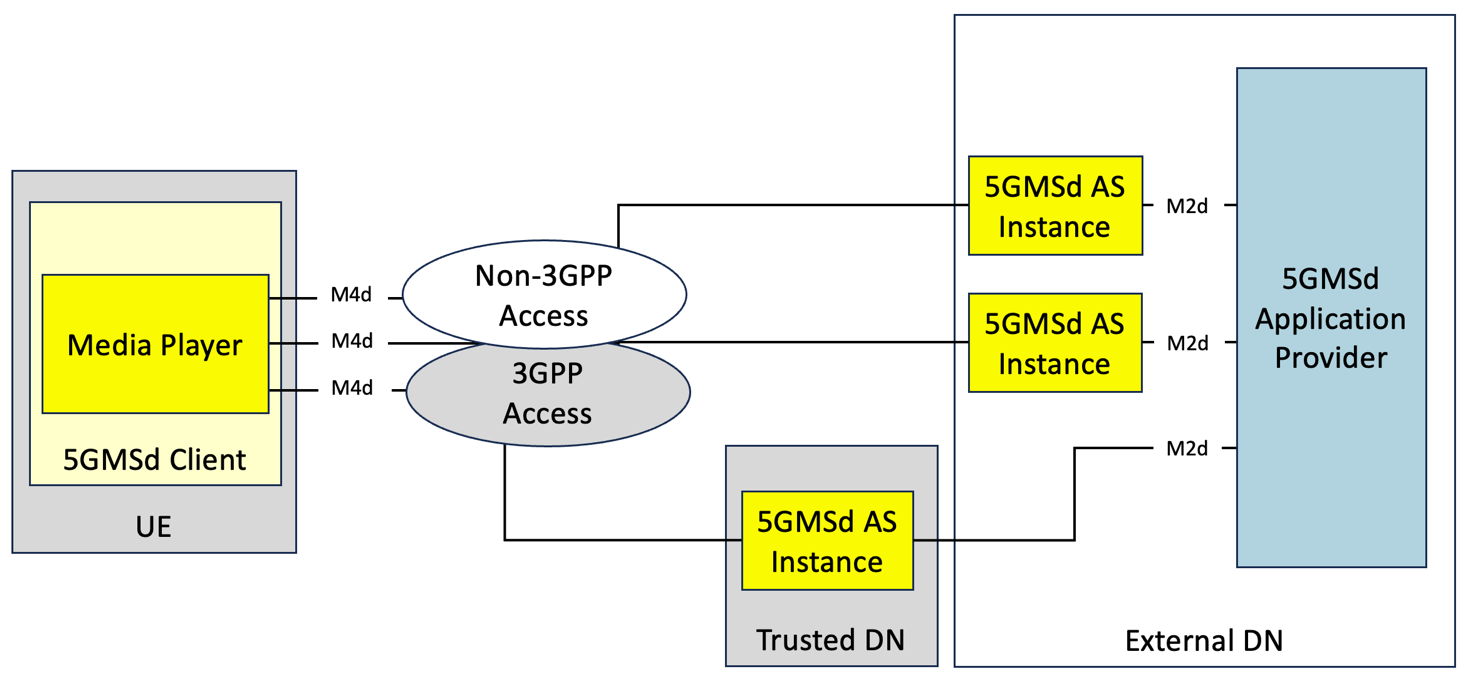


Figure 5.19.2.2-1: Multi-access media delivery within 5G system

### 5.19.3 Architecture mappings

#### 5.19.3.1 Server-side CDN switching

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.3.2 Client-side CDN switching

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.3.3 Concurrent CDN access using CMMF

Several options on the network side, client side and configuration exist when implementing CMMF within the 5GMS System. These are expanded upon further below.

##### 5.19.3.3.1 CMMF content preparation

The use of CMMF for delivering content from multiple sources/CDNs requires that the content be encoded into multiple coded representations (one per source/CDN). This requires the source content to be prepared (i.e., encoded within CMMF) somewhere within the network prior to a client attempting to access it. Various options exist for where these encoding procedures take place. These options include:

1. *5GMSd Application Provider Content Preparation.* In this option, it is the responsibility of the 5GMS Application Provider to encode and package source content within CMMF bitstreams/objects prior to delivery of that content separately to each 5GMSd AS instance via reference point M2d or to each external 5GMSd AS instance. This is illustrated in figures 5.19.2.1-1 and 5.19.2.2-1 above.

2. *Centralized 5GMSd Content Preparation*. In this option, a single, primary 5GMSd AS instance encodes and packages source content that has been ingested at reference point M2d into CMMF bitstreams/objects according to a (yet to be defined) configuration provided by a Content Preparation Template(s). The CMMF bitstreams/objects created during this media processing task may be delivered directly to the 5GMSd Client (via reference point M4d), delivered to another (secondary) 5GMSd AS instance (via reference point M10d), or delivered to a 5GMSd AS located in an external, possibly untrusted, Data Network. These CMMF bitstreams/objects may then be cached and/or forwarded onward. This primary 5GMSd AS instance is responsible for creation of all CMMF encoded representations used to deliver content from multiple sources. This is illustrated in figure 5.19.3.3.1-1 below. The secondary 5GMSd AS instances may be deployed either in the Trusted DN, in an Edge DN or in an External DN.

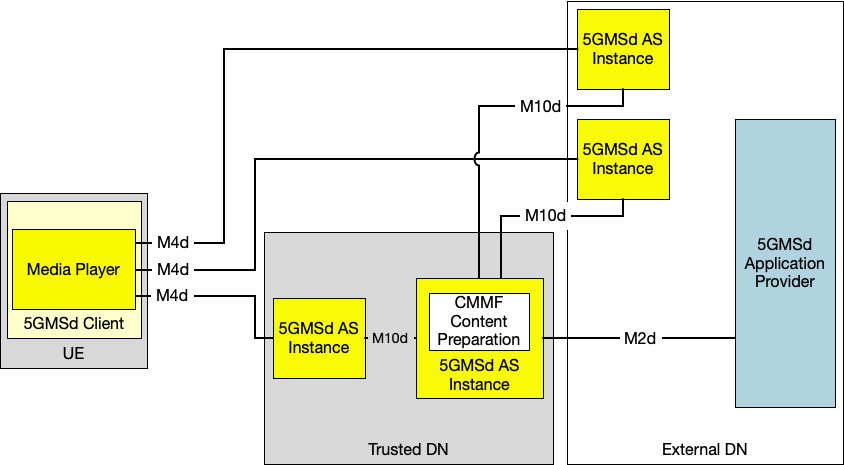


Figure 5.19.3.3.1-1: Option #2 for deploying CMMF within 5GMS  
where a single, primary 5GMSd AS performs all CMMF content preparation.

3. *Decentralized 5GMSd Content Preparation.* The possibility also exists to distribute the CMMF media processing across 5GMSd Application Servers such that each 5GMSd AS instance is only responsible for creation of a single CMMF representation for which it intends to cache and/or deliver to a 5GMSd Client via reference point M4d. In this option, each 5GMSd AS instance provisioned with the Content Preparation Template may receive original source content or CMMF-encoded content from either the 5GMSd Application Provider at reference point M2d or from another 5GMSd AS instance at reference point M10d. This received content is then processed to create a new, unique CMMF-encoded representation which can be used in conjunction with others during a multi-source download. Similarly, an externally deployed 5GMSd AS may be configured by the 5GMSd Application Provider (by private means) to perform a similar media processing task to create its own CMMF-encoded representation. This is illustrated in figure 5.19.3.3.1-2 below.

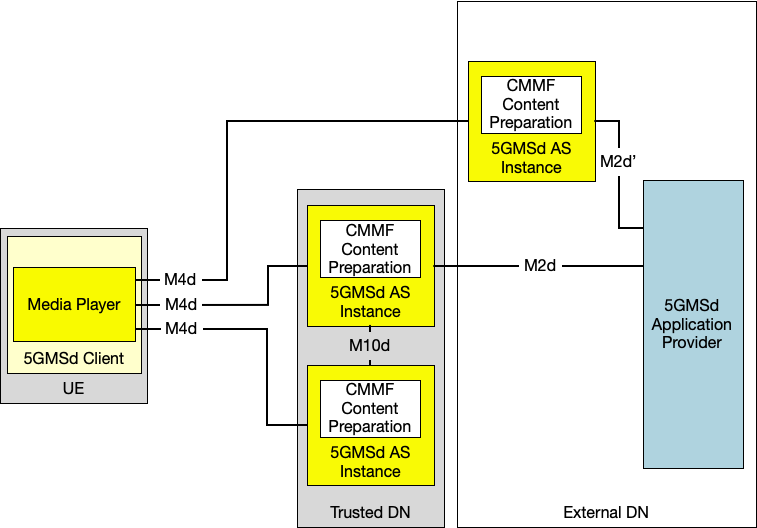


Figure 5.19.3.3.1-2: Option #3 for deploying CMMF within 5GMS  
where each 5GMSd AS instance performs independent CMMF content preparation.

A combination of options 1-3 is also possible where some aspect of all three exist within a physical realization of the network.

##### 5.19.3.3.2 CMMF-enabled 5GMS client architecture

Implementing multi-source/CDN delivery using CMMF requires modifications to the 5GMSd Client. At a minimum, a 5GMSd Client must be able to download CMMF bitstreams/objects from multiple 5GMSd Application Server instances simultaneously and decode the received bitstreams/objects. Options for implementing multi-source/CDN delivery using CMMF within the 5GMSd Client include:

1. *CMMF Client Proxy.* This option implements multi-source/CDN using CMMF within the client as a proxy between the Media Player and each 5GMSd AS. The proxy consists of a CMMF Client and a Media Server. Once the Media Session Handler of the 5GMSd Client has configured the CMMF Client via reference point CMMF-2, the Media Player may request source content via the Media Server using reference point CMMF-3. Once a request is received, the CMMF Client downloads different CMMF encoded representations of the requested content via reference point(s) CMMF-1 (this reference point is functionally equivalent to reference point M4d despite terminating on a different logical function in the 5GMSd Client), decodes the received CMMF bitstreams/objects, and replies to the Media Player with the requested source content via CMMF-3. This option is illustrated in figure 5.19.3.3.2-1.

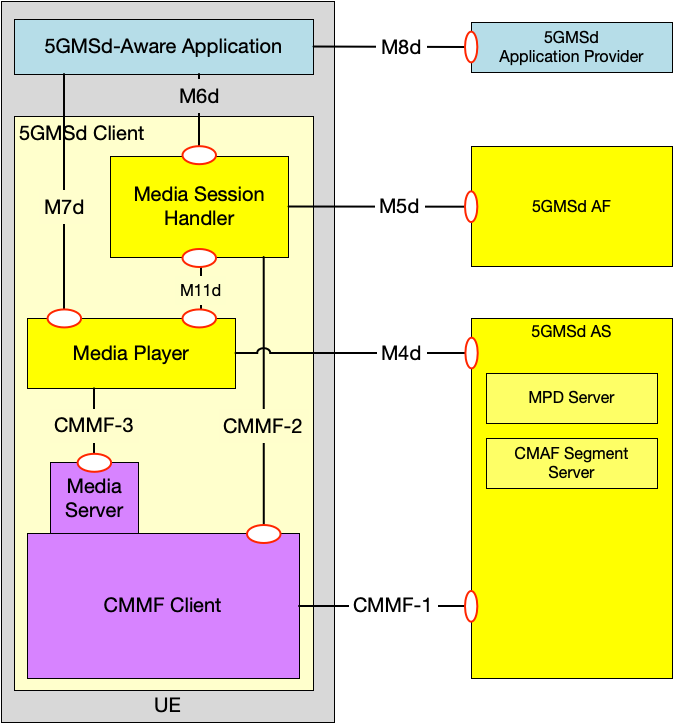


Figure 5.19.3.3.2-1: Option #1 for integration of CMMF within the 5GMS Client where CMMF is implemented as a client proxy.

2. *CMMF decoder integrated in Media Player.* This option implements CMMF within the Media Player itself. An example is provided in figure 5.19.3.3.2-2 depicting CMMF integrated within the DASH-based 5GMSd Client specified in clause 13.2 of TS 26.512 [16]. The architecture and operation of the 5GMS Client is similar to that in [16] with the following exceptions:

a. *Download*: Downloads source content objects and/or CMMF bitstreams/objects from one or more 5GMSd AS instances in parallel.

b. *Request Scheduling:* Performs the same function as defined in clause 13.2 of [16] with the addition of managing the concurrent requests sent over reference point M4d during the download of content encoded within CMMF.

c. *Throughput Estimation:* Estimates the throughput from each individual 5GMSd AS instance in addition to estimating the aggregated throughput from all 5GMSd AS instances.

d. *CMMF Receiver/Decoder:* Temporarily stores and jointly decodes CMMF bitstreams/objects as they are downloaded. Once decoded, the source content objects are moved to the Media Playback Management and Protection Controller. The CMMF Receiver/Decoder also provides status updates containing decode progress to each active download function for the purposes of managing/terminating in-process downloads.

A yellow diagram with many different symbols

Description automatically generated with medium confidence

Figure 5.19.3.3.2-2: Option #2 for integration of CMMF within the 5GMS Client where CMMF is integrated directly within the Media Player.

##### 5.19.3.3.3 CMMF configuration, provisioning, hosting, and Service Access Information

###### 5.19.3.3.3.1 General

CMMF configuration, provisioning, and hosting is the overall responsibility of the 5GMSd Application Provider. The 5GMSd Application Provider may configure and provision resources to deliver media using CMMF across both external and trusted data networks. Options for configuring CMMF media delivery include:

1. CMMF-encoded media delivery is transparent to the 5GMS System. In these cases, the 5GMSd Application Provider prepares all CMMF bitstreams/objects intended to be distributed across every 5GMSd AS instance (whether located within an external or trusted DN). Hosting of CMMF encoded media within the 5GMS System is performed according to established downlink 5G Media Streaming architectures (see clause 4.2 of TS 26.501 [15]) and procedures (see clause 5.2 of [15]). CMMF-enabled 5GMSd Clients accessing CMMF-encoded media hosted on one or more 5GMSd AS instances where the media is replicated rather than different CMMF-encoded representations of that media will treat the set of 5GMSd AS instances as a single CMMF endpoint. CMMF endpoints established in external Data Networks configured and provisioned by the 5GMSd Application Provider can be used to supplement 5GMS-delivered media. In this case, a CMMF-enabled 5GMSd Client is configured to use CMMF via reference point M8d.

2. CMMF content preparation and/or hosting is performed by the 5GMS System. In these cases, the 5GMSd Application Provider configures and provisions 5GMS resources (e.g., 5GMSd AF, 5GMSd AS instances, etc.) via reference point M1d to prepare and/or deliver CMMF-encoded media. The 5GMSd Application Provider may specify CMMF content preparation (e.g., CMMF encoding, etc.) and/or hosting by network-side components of the 5GMS System according to a provisioned CMMF Content Preparation Template. In such cases, individual 5GMSd AS instances may operate as separate CMMF endpoints for the purposes of CMMF-enabled media delivery assuming the system has been appropriately configured and provisioned to ensure CMMF-encoded media is not replicated between available 5GMSd AS instances. 5GMSd Client configuration information is provided by the 5GMS AF via reference point M5d.

The following clauses provide additional details for option 2 above, where configuration, provisioning and hosting CMMF media streaming is supported by the 5GMS System.

###### 5.19.3.3.3.2 CMMF provisioning and service information

The provisioning step allows a 5GMSd Application Provider to configure information about its CMMF content preparation and hosting requirements for media streaming sessions using 5GMS resources. The following information may be made available and/or configurable over reference point M1d:

- Endpoint locations or base URLs outside of the 5GMS System (i.e., within external data networks) that support CMMF delivery.

- Information necessary for configuring the preparation of CMMF-encoded media on 5GMSd AS instances within Trusted DNs.

- Necessary CMMF bitstream/object preparation parameters as defined in [CMMF] required to properly encode and package media within CMMF bitstreams/objects. Examples may include CMMF code type, CMMF profile, etc.

- Information necessary for configuring the hosting of CMMF-encoded media on 5GMS Application Servers within trusted DNs.

- This may include Edge Computing requirements and EAS profile information as defined in clause 4.5.3 of TS 26.501 [15] for each 5GMSd EAS instance that will be used to create CMMF bitstreams/objects.

- CMMF object and original source object URL handling parameters describing the method and/or format used to assign URLs to CMMF objects created within the 5GMS System. More detail is provided in clause 15.19.3.3.4.

Once provisioned, the 5GMSd AF allocates and manages the set of 5GMSd AS instances (including Edge Computing resources, if required) needed for the creation and/or hosting of CMMF bitstreams/objects generated from media provided by the 5GMSd Application Provider at reference point M2d. The 5GMSd AF ensures that each addressable set of 5GMSd AS instances acting as a single individual CMMF endpoint is configured to distribute to CMMF-enabled 5GMSd Clients a unique CMMF bitstream/object created from the original source media.

The 5GMSd AF is also responsible for communicating necessary configuration information (see clause 5.19.4.3.1) with the Media Session Handler in the 5GMSd Client via reference point M5d.

###### 5.19.3.3.3.3 5GMSd Client configuration for downlink media streaming using CMMF

CMMF supplements existing media streaming architectures (e.g., MPEG-DASH, etc.) to enable clients to obtain media from multiple endpoints in parallel. Enabling this functionality requires, at a minimum, information concerning where and how CMMF-encoded media can be accessed on one or more endpoints within the network(s).

This *CMMF client configuration information* may be conveyed to the CMMF Client (assuming client architecture 1 in clause 5.19.3.3.2) or to the Media Player (assuming client architecture 2 in clause 5.19.3.3.2) in the 5GMSd Client either:

1. Privately by the 5GMSd Application Provider as Service Access Information provided via a 5GMSd-Aware Application at reference point M8d. For client architecture 1, this information may be provided to the Media Session Handler via reference point M6d and then on to the CMMF Client via reference point CMMF‑2. For client architecture 2, this information may be provided directly to the Media Player via reference point M7d or routed through the Media Session Handler via reference point M6d followed by reference point M11d.

2. By the 5GMSd AF to the Media Session Handler as Service Access Information made available at reference point M5d. The Media Session Handler may then provide this information to the CMMF Client (assuming client architecture 1) via reference point CMMF-2 or to the Media Player (assuming client architecture 2) via reference point M11d.

3. Contained within a Media Entry Point specified by an existing media streaming framework that is understood by the Media Player (e.g., as additional XML elements or attributes within an MPEG-DASH MPD).

The CMMF client configuration information is the set of parameters and addresses which are needed by a 5GMSd Client to activate and control the reception of a CMMF-enabled media downlink media streaming session, primarily information concerning how the CMMF Client (client architecture 1) or Media Player (client architecture 2) accesses one or more CMMF-encoded objects, each containing a different encoded representation of the original source media to be played, from the available 5GMSd AS instances hosting this content.

- For options 1 and 2 above, additions to the Service Access Information defined in clause 4.2.3 of TS 26.501 [15] are needed to enable the use of CMMF This additional information can be used along with existing Service Access Information (e.g., a Media Entry Point document such as an MPEG-DASH MPD) to construct proper URLs that enable the download of CMMF-encoded content from each CMMF endpoint (5GMSd AS instance).

- For option 3 above, additional XML elements or attributes in the MPEG-DASH MPD are needed to enable the use of CMMF. This additional information can be used along with the Base URLs in the MPEG-DASH MPD to construct proper URLs that enable the download of CMMF-encoded content from each addressable set of 5GMSd AS instances (CMMF endpoint).

client

##### 5.19.3.3.4 CMMF object addressing and URL handling

The capability to locate and access multiple CMMF-encoded media objects (each containing different encoded representations of the same original source media) and/or the original source media within the 5GMS System is required by both 5GMSd Clients and 5GMS AS instances. Furthermore, the creation of these CMMF-encoded media representations may be dynamic where 5GMSd AS instances are creating and caching new CMMF-encoded media objects on demand. Having robust and extensible methods for addressing each CMMF-encoded media object, including the original source media, is necessary to ensure the proper operation of coded multi-source media delivery. At least two approaches exist for addressing these media. Each approach can be used independently or a hybrid of the two can be used in combination.

1. *Unique URL path names.* Each CMMF-encoded media object containing a different representation of the same original source media object, as well as the original source media object itself, are each assigned a unique URL path (i.e., there exists a one-to-one mapping between each CMMF-encoded media object or original source media object and each URL path). For example, every URL contains a unique path to a unique CMMF-encoded media object and original source media object. Table 5.19.3.3.4-1 provides examples of possible URL path formats where the differences in each URL emphasised in **boldface**. This approach enables multiple CMMF-encoded media objects, each containing a different representation of the same original source media object, to be co-located on a single 5GMSd AS instance (or CMMF endpoint).

Definition of the patterns used may be defined by a CMMF URL template that aids in mapping the URLs obtained from a content manifest (e.g., MPD) to CMMF-encoded media that is available via the 5GMS System. In addition to this CMMF URL template, the CMMF endpoint URLs may also be required by the CMMF Client (client architecture 1) or Media Player (client architecture 2) to construct a complete URL needed to fetch CMMF-encoded content.

Table 5.19.3.3.4-1: Approach #1 example URLs assigned to each unique CMMF-encoded media object and original source media object

|  |  |
| --- | --- |
| Object Description | Path Examples |
| Original source media object (e.g., media object referenced within an MPEG-DASH MPD or HLS manifest) | /1080p/4mbps/1.m4s |
| CMMF-encoded media object containing representation A of the original source media object | Example 1: /1080p/4mbps/1.m4s**.cmmf-a** Example 2: /1080p/4mbps/**cmmf-a/**1.m4s |
| CMMF-encoded media object containing representation B of the original source media object | Example 1: /1080p/4mbps/1.m4s**.cmmf-b** Example 2: /1080p/4mbps/**cmmf-b/**1.m4s |
| … | … |

2. *Unique URL authority names.* Each CMMF-encoded media object containing a different representation of the same original source media object, as well as the original source media object itself, are each assigned a unique URL where they all share a common URL path, but their URLs differ in the authority. Table 5.19.3.3.4-2 provides examples of possible URLs where the differences in each URL are emphasised in **boldface**; and the approach is also illustrated in figure 15.19.1.3.1-3.

URL assignment following this method requires that each 5GMSd AS instance (CMMF endpoint) contain one, and only one, representation of the original source media. That representation may be the original source media object itself or a CMMF-encoded media object created from the original source media object. While this approach limits how CMMF-encoded media is distributed within the 5GMS System, it may also simplify content preparation and hosting management by eliminating the need to track and manage creation and placement of CMMF-encoded media objects within the 5GMS System because it is implied that each and every addressable set of 5GMSd AS instances (CMMF endpoint) hosts a different representation of the original source media (whether that is the original source media object itself or a CMMF-encoded media object created from the original source media object).

Table 5.19.3.3.4-2: Approach #2 example URLs assigned to each unique CMMF-encoded media object and original source media object

|  |  |
| --- | --- |
| Object Description | Path Examples |
| Original source media object (e.g., media object referenced within an MPEG-DASH or HLS manifest) | //**5gms-as-1/**1080p/4mbps/1.m4s |
| CMMF-encoded media object containing representation A of the original source media object | //**5gms-as-2/**1080p/4mbps/1.m4s |
| CMMF-encoded media object containing representation B of the original source media object | //**5gms-as-3/**1080p/4mbps/1.m4s |
| … | … |

The use of one approach over another may be influenced by the method in which CMMF content is prepared.

- For options #1 and #2 in clause 15.19.3.3.1, where CMMF content preparation is centralized within either the 5GMS Application Provider or on a single 5GMSd AS instance, assigning unique URL path names to each original source media object and each CMMF-encoded media object can be easily implemented and managed according to Approach #1 above.

- Approach #2, where each CMMF-encoded media object representation is assigned a different URL authority name, may be appropriate when CMMF content preparation is performed in a decentralized manner as shown in option #3 of clause 5.19.3.3.1. In these cases, no central management is necessary once the 5GMS System has been configured to prepare CMMF-encoded content or assign URLs and track the location of that content.

### 5.19.4 High-level call flows

#### 5.19.4.1 Server-side CDN switching

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.4.2 Client-side CDN switching

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.4.3 Concurrent CDN access using CMMF

##### 5.19.4.3.1 Overview

The intent of CMMF is to supplement existing downlink streaming procedures, rather than replace them entirely. As such, minimal changes to the procedures provided in clause 5 of TS 26.501 [15] are necessary. Enabling multi-CDN media delivery using CMMF within existing downlink media delivery workflows can generally be realised through the following:

1. CMMF-encoded media objects, and possibly original source media (e.g., MPEG-DASH or HLS media segments), are striped across multiple 5GMSd Application Server instances. The uniquely addressable set of one or more 5GMSd AS instances containing the same CMMF-encoded media object (or stripe) of the media (i.e., the CMMF-encoded media object is replicated across multiple 5GMSd AS instances) is considered a single CMMF endpoint rather than multiple CMMF endpoints. The 5GMSd Application Provider may make the CMMF-encoded media objects, and possibly original source media (e.g., MPEG-DASH or HLS media segments), available at reference point M2d or they may be created by a 5GMSd AS instance performing content preparation on either regular media objects (e.g., MPEG-DASH or HLS media segments) or already created CMMF-encoded media objects.

2. Upon initialization of a playback session, the 5GMSd Media Client’s Media Session Handler obtains relevant Service Access Information from the 5GMSd AF at reference point M5d. At a minimum, this includes details concerning the location of each set of 5GMSd AS instances from which a stripe of CMMF-encoded and possibly original media (e.g., MPEG-DASH or HLS media segments) may be obtained, as well as appropriate signalling to indicate whether the media at each location is CMMF-encoded. Each of these locations should be considered a CMMF endpoint as defined above.

3. The 5GMSd Media Client connects to and downloads CMMF-encoded media objects, and possibly the original source media (e.g., MPEG-DASH or HLS media segments), from each CMMF endpoint simultaneously via reference point CMMF‑1 (5GMSd Client architecture Option #1) or M4d (5GMSd Client architecture Option #2), terminating the download from each 5GMSd AS instance early upon obtaining enough of the CMMF-encoded objects to recover the source media (e.g., MPEG‑DASH or HLS media segment). Once decoded, the source media is delivered to the Media Player in the 5GMSd Media Client for presentation.

The procedures for enabling multi-CDN or multi-5GMSd AS instance delivery using CMMF for MPEG-DASH streaming sessions and provisioning the 5GMS network to deliver CMMF-encoded media objects are provided in the following clauses.

##### 5.19.4.3.2 CMMF provisioning, hosting, and processing procedure

The workflow for provisioning network resources and/or hosting CMMF-encoded media is dependent on the 5GMSd Application Provider requirements and network configuration. Assuming the 5GMSd Application Provider creates CMMF-encoded media, provides that media over reference point M2d, provisions CMMF endpoints external to the 5GMS System, and manages the configuration of the 5GMSd Client over reference point M8d, existing procedures as outlined in clause 5 of TS 26.501 [15] are sufficient to host and deliver CMMF-encoded media. However, cases exist where the 5GMS System may take a more active role in delivering CMMF-encoded media. These cases are outlined in the procedure below. Differences from the baseline procedure in clause 7.2 of TS 26.501 [15] are highlighted in **boldface**.



Figure 5.19.4.3.2-1: Media provisioning, hosting, and processing procedures for downlink media streaming using CMMF

Steps:

1. Upon setting up a **CMMF Provisioning and Content Hosting Configuration**, the 5GMSd Application Provider requests **CMMF** media processing and hosting to be set up. **The 5GMSd Application Provider provides a description of the type and placement of the processing used to encode CMMF objects, including the required number of CMMF endpoints to be exposed as addressable 5GMSd AS instance sets, and defines the flow of content through the 5GMS System.** The 5GMS System may only allow a shortlisted set of media processing functions to be used by the 5GMSd Application Provider.

4. The 5GMSd AF confirms the successful creation of the **CMMF Provisioning and Content Hosting Configuration** with the requested media processing to the 5GMSd Application Provider.

5. The 5GMSd AF exchanges relevant **CMMF client configuration information** with the 5GMSd Client to enable the 5GMSd Client to access and download **both CMMF-encoded media and original source media**. **This CMMF client configuration information may consist of URLs to each CMMF endpoint established in step 2, a CMMF URL template allowing for translation of the URLs provided in content manifests (e.g., MPD) into URLs that can be used to access CMMF-encoded media within the 5GMS System, etc.**

6. A 5GMSd Client sends one or more requests for **CMMF-encoded media content** to one or more of the 5GMSd AS instance(s) (established as CMMF endpoints) listed in the provisioned Content Hosting Configuration (see clause 5.4 of TS 26.501 [15]).

7. If it does not already have a copy of the requested media cached, the 5GMSd AS addressed by the 5GMSd Client fetches the media **from a back-end 5GMSd AS instance** or from the 5GMSd Application Provider.

NOTE 1: Multiple options are available for distributing CMMF-encoded media to the addressed 5GMSd AS instance. These options are dependent on how the 5GMS System is provisioned and configured during steps 1-4. Clause 5.19.4.3.4 enumerates these options and provides more details on the call flows involving cache misses.

8. Depending on the media content received **from a back-end 5GMSd AS instance** or from the 5GMSd Application Provider, **the addressed 5GMSd AS may be required to create a unique CMMF-encoded representation of the requested media using content processing provisioned in step 1 and referenced by the Content Hosting Configuration**.

NOTE 2: Additional details are provided in clause 15.19.4.3.4.

9. The addressed 5GMSd AS instance serves **the requested CMMF-encoded media or original source media** to the 5GMSd Client.

Different variants of these procedures may be possible, depending on the placement of the processing, the placement of the CMMF endpoints, and the characteristics of the CMMF Content Provisioning and Hosting Configuration.

##### 5.19.4.3.3 Procedures for CMMF downlink media streaming

The procedure depicted below illustrates how CMMF can supplement downlink media delivery using MPEG-DASH as defined in clause 5.2.3 of TS 26.501 [15]. Differences from the baseline procedure are highlighted in **boldface**.



Figure 5.19.4.3.3-1: High-level procedure for delivering DASH content using CMMF

Prerequisites:

- The 5GMSd Application Provider has provisioned the 5G Media Streaming System and has set up content ingest. **In the case where the 5G Media Streaming System is responsible for CMMF media object preparation, the 5GMSd Application Provider has also provisioned 5GMS resources to encode CMMF media objects and host those objects on 5GMSd Application Server instances.**

- The 5GMSd-Aware Application has received the service announcement from the 5GMSd Application Provider.

Steps:

4. When the 5GMS-Aware Application has received only a reference to the Service Access Information (see step 1), the Media Session Handler interacts with the 5GMSd AF to acquire the whole Service Access Information. **The Service Access Information includes relevant CMMF client configuration information. This CMMF client configuration information may consist of URLs to each CMMF endpoint, a CMMF URL template allowing for translation of the URLs provided in content manifests (e.g., MPD) into URLs that can be used to access CMMF-encoded media within the 5GMS System, etc.**

4a. **The Media Session Handler provides the CMMF Client or CMMF-enabled Media Player with the necessary information to construct complete URLs to CMMF-encoded media with the 5GMS System.**

9. The Media Player processes the MPD. **Based on the information contained within the MPD as well as the relevant CMMF client configuration Information contained in the Service Access Information, the CMMF Client or CMMF-enabled Media Player determines, for example, the number of needed transport sessions for media acquisition, complete URLs to CMMF-encoded media, etc.** The Media Player should use the MPD information to initialize the media pipelines for each media stream. The MPD should also contain information to initialize the DRM client, when DRM is used.

11. Optional: the Media Player acquires the necessary DRM information, for example a DRM License.

**NOTE: CMMF does not impede the use of DRM as long as DRM is applied to the original source media prior to creation of CMMF-encoded objects of that media.**

13. The Media Player establishes the necessary transport sessions for the **CMMF-encoded** content. For example, the **CMMF Client or CMMF-enabled** Media Player may establish one transport session for each media component (audio, video, etc) and possibly additional transport sessions for other media representations **to each CMMF endpoint communicated by the CMMF client configuration information.**

15. The Media Player requests initialization information. **In the case where this initialization information has been encoded within CMMF objects, the CMMF Client or CMMF-enabled Media Player requests the CMMF-encoded initialization information objects from each CMMF endpoint in parallel.** **The URLs of the CMMF-encoded initialization information objects are determined using the MPD and information contained within the CMMF client configuration information (e.g., base URLs to each CMMF endpoint, CMMF URL Template, etc.).** The **CMMF Client of CMMF-enabled** Media Player repeats this step for each required initialization segment.

16. The **CMMF Client or CMMF-enabled** Media Player receives the initialization information. **In the case where this initialization information has been encoded within CMMF objects, the CMMF Client or CMMF-enabled** **Media Player downloads these multiple CMMF-encoded initialization information objects from each CMMF endpoint in parallel until such time as the CMMF decoder has received enough information to successfully decode, at which point the download of any incompletely acquired CMMF object is abandoned.**

16a. **In the case where the initialization information has been encoded within CMMF objects, the received information from the CMMF objects containing the initialization information is decoded by the CMMF decoder in the CMMF Client.**

17. The **CMMF Client or CMMF-enabled** Media Player requests media segments according to the MPD. **In the case where these media segments have been encoded within CMMF objects, the CMMF Client or CMMF-enabled** **Media Player requests the CMMF-encoded media segment objects from each CMMF endpoint in parallel. The URLs of the CMMF-encoded media segment objects are determined using the MPD and information contained within the CMMF Configuration Information (e.g., base URLs to each CMMF endpoint, CMMF URL Template, etc.).**

18. The **CMMF Client or CMMF-enabled** Media Player receives media segments. **In the case where these media segments have been encoded within CMMF objects, the CMMF Client or CMMF-enabled Media Player downloads these multiple CMMF-encoded media segment objects from each CMMF endpoint in parallel until such time as the CMMF decoder has received enough information to successfully decode, at which point the download of any incompletely acquired CMMF object is abandoned.**

18a. **In the case where the media segments have been encoded within CMMF objects, the received information from the CMMF objects containing the media segment is decoded by the CMMF decoder in CMMF Client** **and the decoded information is placed** into the appropriate media rendering pipeline.

##### 5.19.4.3.4 CMMF content preparation and retrieval procedure

###### 5.19.4.3.4.1 Overview

The process for preparing and distributing CMMF-encoded media is dependent on the 5GMSd Application Provider’s requirements, as well as the capabilities and configuration the 5GMS System. The workflows provided below expand upon the processes that occur when the CMMF Client or CMMF-enabled Media Player requests CMMF objects from a CMMF endpoint (5GMSd AS instance) in steps 7 and 8 of clause 15.19.4.3.2 and in steps 15 and 16, as well as steps 17 and 18, within clause 5.19.4.3.3. These workflows focus on the processes a 5GMSd AS instance follows to locate and distribute CMMF objects upon receiving a request from the 5GMSd Client’s CMMF Client or CMMF-enabled Media Player. For the purposes of these procedures, it is assumed that each addressable set of 5GMSd AS instances is configured as a different CMMF endpoint. In the case where multiple 5GMSd AS instances belong to a single CMMF endpoint, the procedures provided below can be modified accordingly.

Different variants of these procedures may be possible, depending on the type of processing, the placement of the processing, and the characteristics of the Content Hosting Configuration. Furthermore, the introduction of CMMF to supplement the download of media in other scenarios (e.g., downlink streaming to Media Players with different presentation manifests per clause 5.2.4 of TS 26.501 [15]) can be applied in a similar fashion to that shown below.

###### 5.19.4.3.4.2 Content preparation by 5GMSd Application Provider

The procedure used when the 5GMSd Application Provider is responsible for encoding and packaging source media within CMMF bitstreams/objects prior to delivery of that content separately to each 5GMSd AS instance via reference point M2d, or to each externally deployed 5GMSd AS instance, is provided in figure 15.19.4.3.4.2-1. In these cases, the 5GMSd Application Provider provides a unique CMMF-encoded representation of original source media to each 5GMSd AS instance configured as a CMMF endpoint. (This case is illustrated in figures 5.19.2.1-1 and 5.19.2.2-1.)



Figure 5.19.4.3.4-1: 5GMSd Application Provider CMMF Content Preparation and media processing procedures for downlink media streaming

Steps:

**1. The 5GMSd Client sends request(s) for** **CMMF-encoded media to one or more 5GMSd AS instances where each 5GMSd AS instance is configured as a different CMMF endpoint**.

**2. If the 5GMSd AS instance does not have the required CMMF-encoded media cached, it requests and fetches a unique CMMF-encoded representation of the requested media from the 5GMSd Application Provider. In this case, it is the responsibility of the 5GMSd Application Provider to ensure that the CMMF-encoded representations of the requested media provided to each CMMF endpoint are unique. Upon receipt of the requested unique CMMF-encoded representation of the requested media, the 5GMS AS instance may cache the CMMF object locally to support subsequent requests for that content.**

**3. If the requested CMMF-encoded media is cached, the 5GMSd AS instance pulls the content from cache.**

**4. Once the CMMF-encoded representation of the requested media has been obtained by the 5GMSd AS instance, the 5GMSd AS instance responds to the 5GMSd Client’s request with a CMMF-encoded representation of the requested media that is unique to the CMMF endpoint for which the 5GMSd AS instance belongs.**

###### 5.19.4.3.4.3 Centralized 5GMSd content preparation

Figure 5.19.4.3.4.3-1 shows the procedure used when source content is ingested at reference point M2d by a single, primary 5GMSd AS instance that encodes and packages it into CMMF bitstreams/objects according to a configuration provided by a Content Preparation Template contained within a CMMF Provisioning and Hosting Configuration previously provisioned by the 5GMSd Application Provider.

In these cases, the placement of 5GMSd AS instances is hierarchical, as shown in figure 5.19.3.3.1-1 and the client-addressable 5GMSd AS instances (CMMF endpoints) accessed at reference point M4d by 5GMSd Clients ingest, via reference point M10d, CMMF-encoded media from an upstream back-end 5GMSd AS instance that is configured to encode and package source media into CMMF bitstreams/objects. The 5GMSd AS instance performing CMMF content preparation is responsible for providing a unique CMMF-encoded representation of the original source media to each client-addressable 5GMSd AS configured as a CMMF endpoint.



Figure 5.19.4.3.4.3-1: Centralized 5GMSd Content Preparation and media processing procedures for downlink media streaming

Steps:

**1. The 5GMSd Client sends request(s) for** **CMMF-encoded media to one or more addressable sets of 5GMSd AS instances where each set is configured as a different CMMF endpoint**.

**2. If the resolved 5GMSd AS instance does not have the required CMMF-encoded media cached, it requests, via reference point M10d, a unique CMMF-encoded representation of the requested media from an upstream 5GMSd AS instance**.

**3. Upon receipt of a request for CMMF-encoded media from a downstream 5GMSd AS instance via reference point M10d, the original source media is ingested by the upstream 5GMSd AS instance from the 5GMSd Application Provider via reference point M2d.**

**4. The original source media is encoded and packaged within a unique CMMF-encoded representation by the content preparation function of the upstream 5GMSd AS instance.**

**5. The CMMF-encoded media is provided to the downstream 5GMSd AS instance via reference point M10d. Upon receipt, the CMMF-encoded media may be cached by the downstream 5GMSd AS instance to support subsequent requests for that content.**

If the requested CMMF-encoded media is already cached by the downstream 5GMSd AS instance

**6. The downstream 5GMSd AS instance, retrieves the requested content from its cache.**

**7. The 5GMSd AS instance responds to the 5GMSd Client’s request at reference point CMMF‑1 or M4d (as appropriate) with a CMMF-encoded representation of the requested media that is unique to the CMMF endpoint to which the 5GMSd AS belongs.**

###### 5.19.4.3.4.4 Decentralized 5GMSd content preparation

Theprocedure shown in figure 15.19.4.3.4.4-1 is used when CMMF media processing and content preparation is distributed across 5GMSd AS instances where each 5GMSd AS instance is responsible for the creation of a single CMMF representation which it intends to cache and/or deliver via reference point CMMF-1 or M4d (depending on the assumed 5GMSd Client architecture described in clause 5.19.3.3.2. This case is illustrated in figure 15.19.3.3.1-2.



Figure 15.19.4.3.4.4-1: Decentralized 5GMSd Content Preparation and media processing procedures for downlink media streamingSteps:

**1. The 5GMSd Client sends request(s) for** **CMMF-encoded media to one or more addressable sets of 5GMSd AS instances where each set is configured as a different CMMF endpoint**.

**2. If the resolved 5GMSd AS instance does not have the required CMMF-encoded media cached and it is aware (through its configuration) that the original source media is available from an upstream 5GMSd AS instance, it requests and receives via reference point M10d a copy of the original source media**.

**3. The back-end 5GMSd AS instance returns the original source media, after possibly retrieving this from the 5GMSd Application Provider if it does not have a copy already cached.**

**4. The original source media is encoded and packaged within a unique CMMF-encoded representation.**

Alternatively:

**5. If the resolved 5GMSd AS instance does not have the required CMMF-encoded media cached and it is aware (through its configuration) that a copy of an already CMMF-encoded representation is available from a peer 5GMSd AS instance hosting a different CMMF endpoint, it requests and receives via reference point M10d a copy of that CMMF-encoded media from that other CMMF endpoint**.

**6. The peer 5GMSd AS instance returns the CMMF-encoded media, after possibly retrieving the original source media from the 5GMSd Application Provider and encoding it into the required CMMF representation according to its configured Content Preparation Template if it does not have a copy already cached.**

**7. The CMMF-encoded representation received from the peer 5GMSd AS instance is first decoded back to the original source media and then re-encoded and packaged within the unique CMMF-encoded representation configured for this CMMF endpoint in the Content Preparation Template for the resolved 5GMSd AS instance.**

If the requested CMMF-encoded media is already cached by the resolved 5GMSd AS instance:

**6. The resolved 5GMSd AS instance retrieves the requested content from its cache.**

**7. The resolved 5GMSd AS instance responds to the 5GMSd Client’s request at reference point CMMF‑1 or M4d (as appropriate) with a CMMF-encoded representation of the requested media that is unique to the CMMF endpoint to which the 5GMSd AS instance belongs.**

### 5.19.5 Gap analysis and requirements

#### 5.19.5.1 Server-side CDN switching

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.5.2 Client-side CDN switching

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.5.3 Concurrent CDN access using CMMF

Clauses 5.19.3.3 and 5.19.4.3 outline how concurrent multi-CDN media delivery can be accomplished using CMMF within the 5GMS System. The methods described range from CMMF multi-source delivery being largely transparent to the 5GMS System to fully integrated solutions where the 5GMS System performs an active role in preparation and delivery of CMMF-encoded media (additional detail is provided in clause 15.19.3.3.3.1). The following gaps in existing 5G Media Streaming specifications have not yet been addressed to support the capabilities discussed.

At a minimum, several modifications to the 5GMSd Client architecture are required to enable concurrent media delivery from multiple CMMF endpoints (addressable sets of 5GMSd AS instances), regardless of how the network has been configured by the 5GMSd Application Provider. These include:

- Support for additional signalling required to configure the 5GMSd Client to use CMMF multi-source delivery including the communication of CMMF endpoints where CMMF-encoded media can be accessed. This may require changes to provisioning information supplied to the 5GMSd AF at reference point M1d as well as to the Service Access Information consequently provided by the 5GMSd AF to the 5GMSd Client at reference point M5d.

- New 5GMSd Client functionality to locate, access, and download CMMF-encoded media concurrently from multiple CMMF endpoints via reference point M4d.

- New 5GMSd Client functionality to decode received (or partially received) CMMF-encoded media objects to recover original source media objects required for playback.

A 5GMS System that provides support for this minimum set of requirements can be used to stream CMMF-encoded media without additional modifications to other 5G Media Streaming specifications. In these cases, the 5GMSd Application Provider can enable CMMF multi-source delivery using a combination of 5GMS and non-5GMS resources. All media hosted by the 5GMS System is prepared by the 5GMSd Application Provider and delivery, with respect to the 5GMS network, is performed according to established downlink 5G Media Streaming architectures (see clause 4.2 of TS 26.501 [15] and procedures (see clause 5.2 of [15]). However, the following changes are necessary should the 5GMSd Application Provider need 5GMS System resources to perform CMMF content preparation and hosting:

- Functionality to configure and access non-overlapping sets of 5GMSd AS instances (i.e., CMMF endpoints) within the 5GMS System. Each set of 5GMSd AS instances needs to be individually addressable by 5GMSd Clients at reference point M4d to enable the concurrent download of CMMF-encoded content.

- Functionality to configure and assign unique CMMF-encoded media representations (or stripes) to each individually addressable set of 5GMSd AS instances (i.e., CMMF endpoints).

- Functionality to enable each individually addressable set of 5GMSd AS instances (i.e., CMMF endpoints) to obtain and/or create unique CMMF-encoded media representations (or stripes) from sources both external and internal to the 5GMS System.

- 5GMSd AF functionality to provide necessary CMMF client configuration information to the 5GMSd Client at reference point M5d.

### 5.19.6 Candidate solutions

#### 5.19.6.1 Server-side CDN switching

These candidate solutions include approaches where a media streaming client or population of clients changes or switches between two or more CDNs based on recommendations from a remote server. An example includes the DASH Industry Forum’s content steering architecture [DIFCS].

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.6.2 Client-side CDN switching

These candidate solutions include approaches where a media streaming client changes or switches between two or more CDNs based on decisions made locally.

Editor’s Note: Inclusion and expansion on these sets of solutions is dependent on interest from working group.

#### 5.19.6.3 Concurrent CDN access using CMMF

This candidate solution includes approaches where a 5GMSd Client accesses and downloads CMMF-encoded media objects [CMMF], and possibly original source media (e.g., MPEG-DASH or HLS media segments), from two or more addressable sets of 5GMSd AS instances simultaneously. Additionally, the 5GMSd Client may access these 5GMSd AS instances over different access networks (such as 3GPP and non-3GPP access networks).

### 5.19.7 Summary and conclusions

## END OF CHANGES