



Coded Multi-Source Media Format (CMMF) Introduction

—

JASON CLOUD, JEFF RIEDMILLER, FREDDIE SANCHEZ, FREDERIC GABIN, CHRISTIAN WOLFF, THOMAS STOCKHAMMER (QUALCOMM)



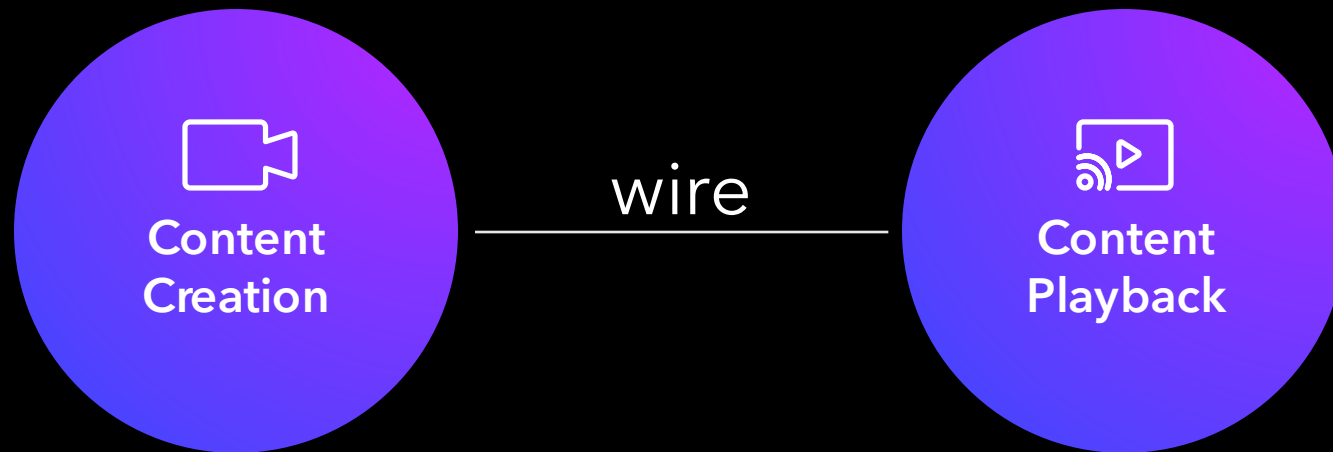
MOTIVATION FOR OUR WORK



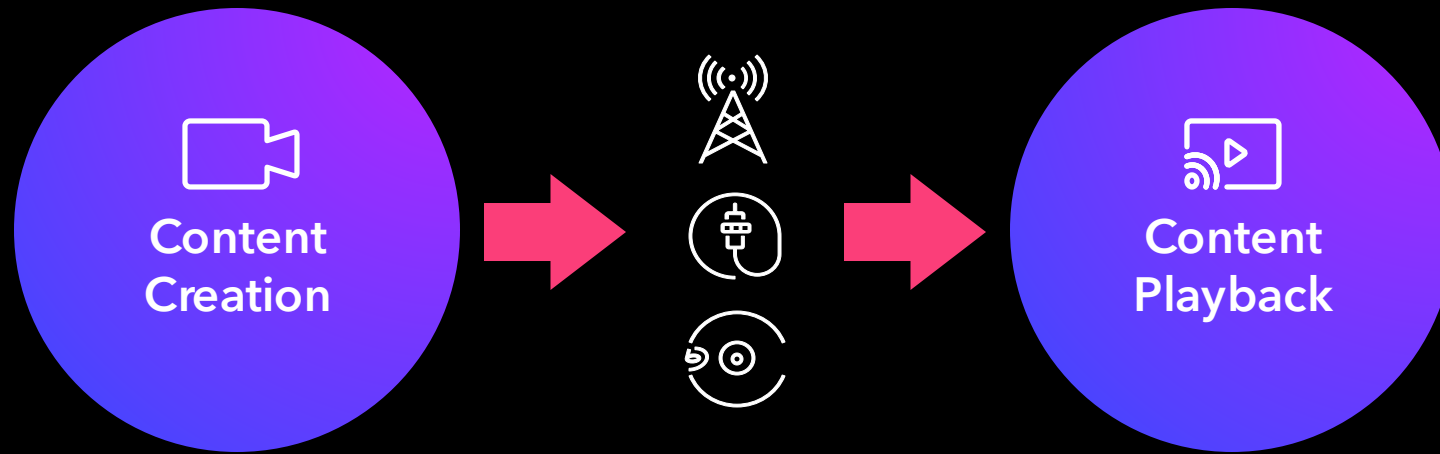
**Content
Creation**



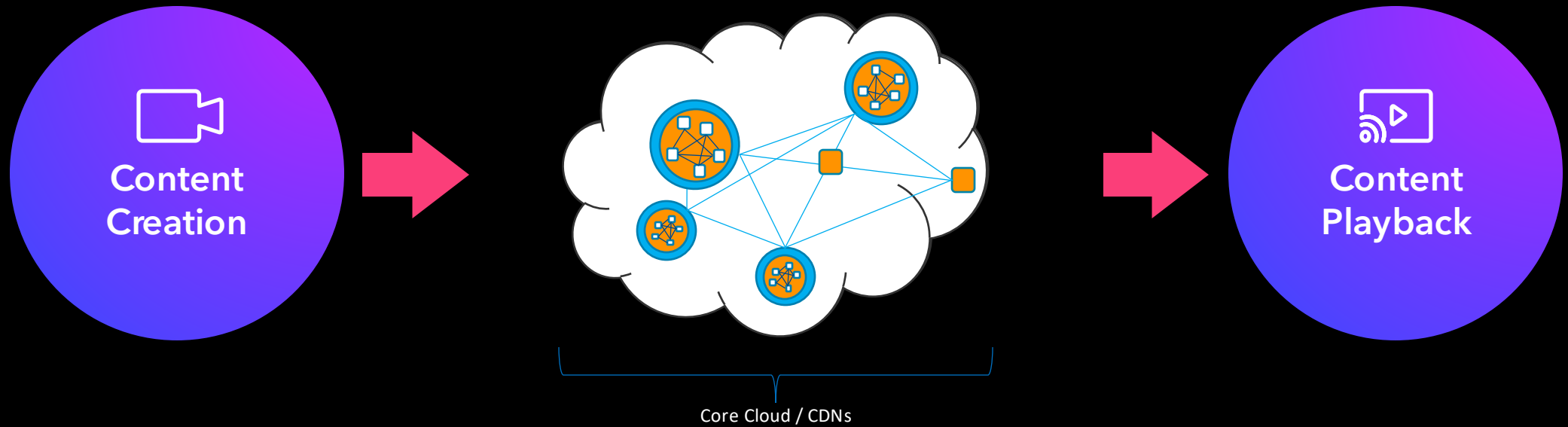
**Content
Playback**



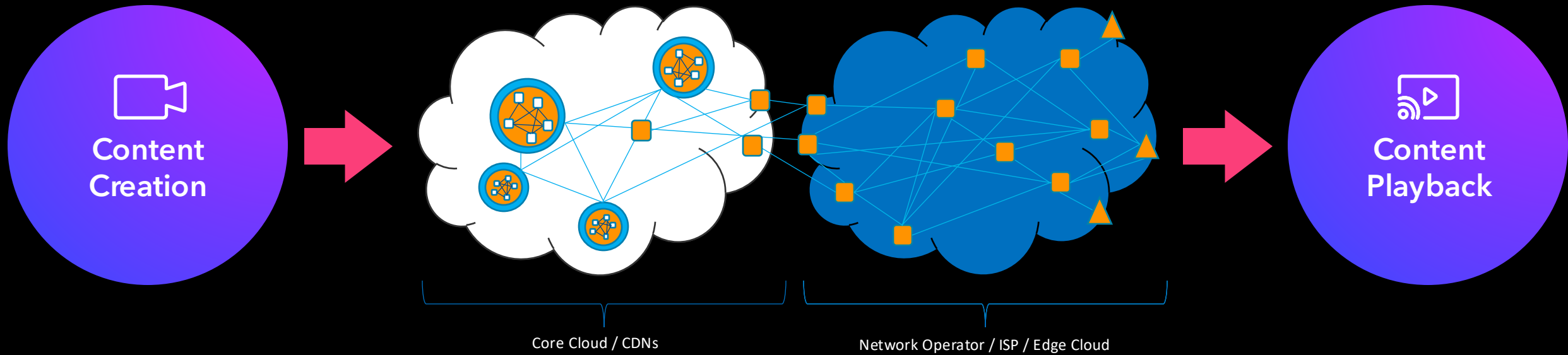
Legacy



Cloud & Network



Clouds & Networks

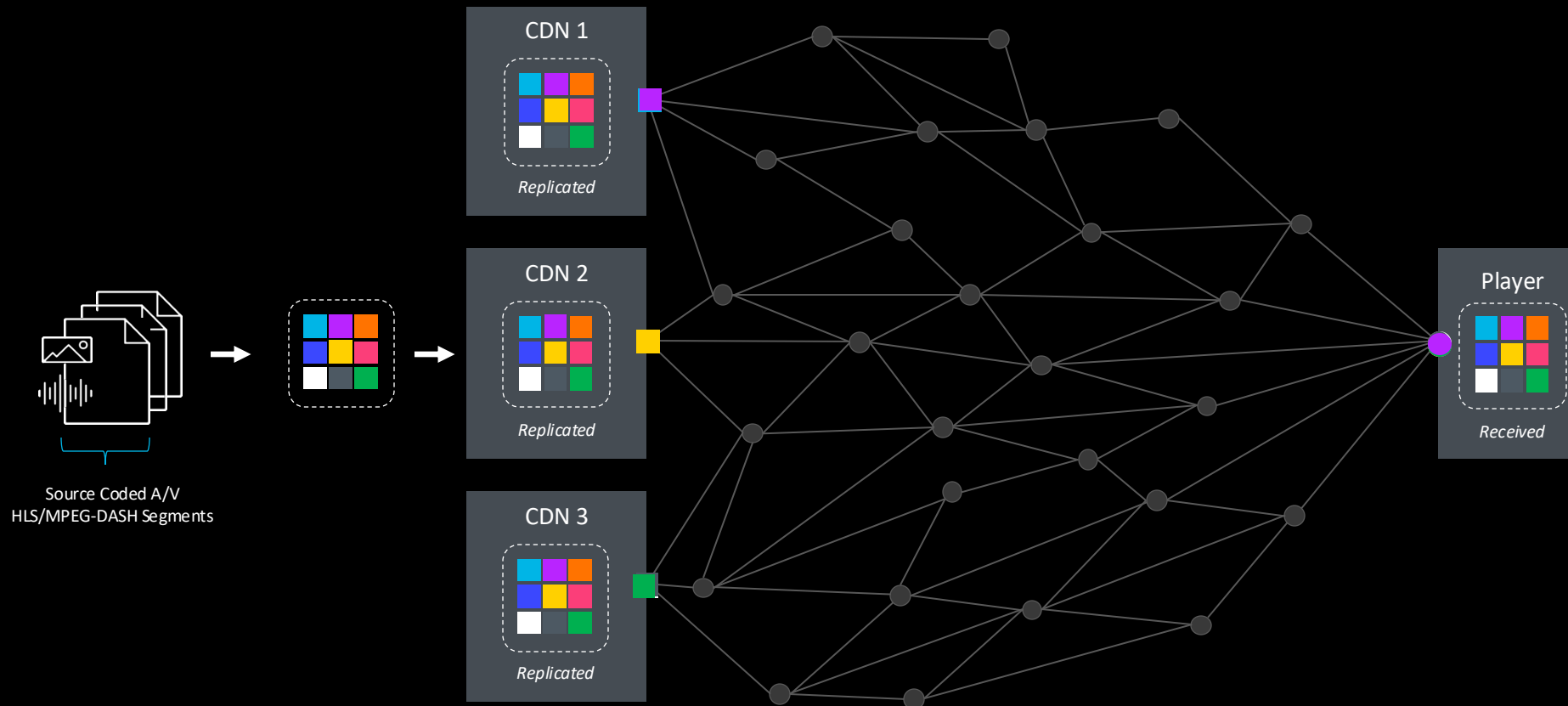


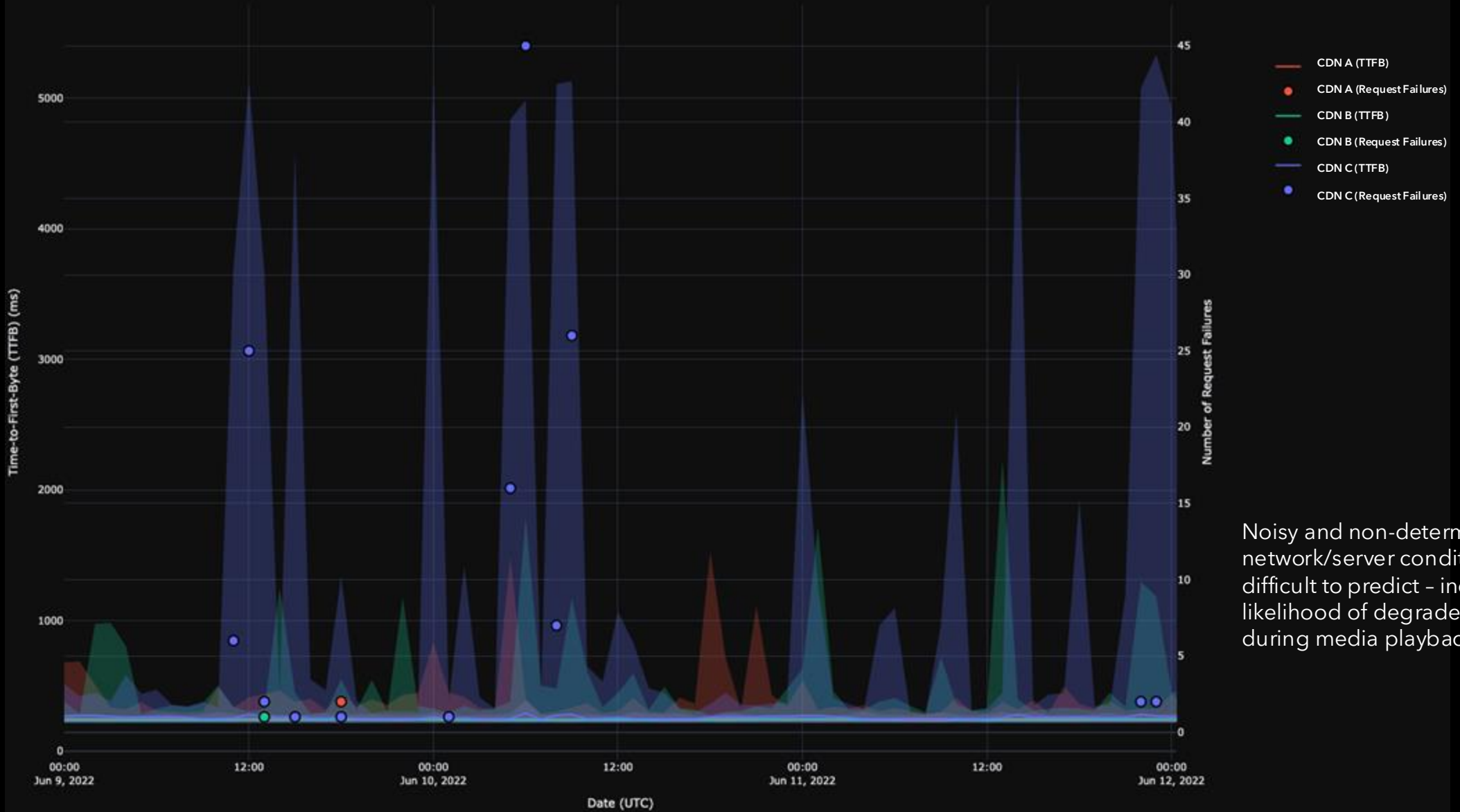
New Thinking: Media designed to work *with* the network

- Rearchitecting the media representation itself – to seamlessly work *with* the underlying network infrastructure and protocols – opens new operating and performance capabilities by enabling **efficient** distribution of media from a plurality of network sources.
- This new **multisource** media representation has capacity to provide higher average throughput, lower latency, and significantly higher reliability that:
 - Reduces/eliminates traffic scheduling complexities
 - Minimizes reliance on latent network performance data to make decisions
 - Requires little-to-no orchestration overhead

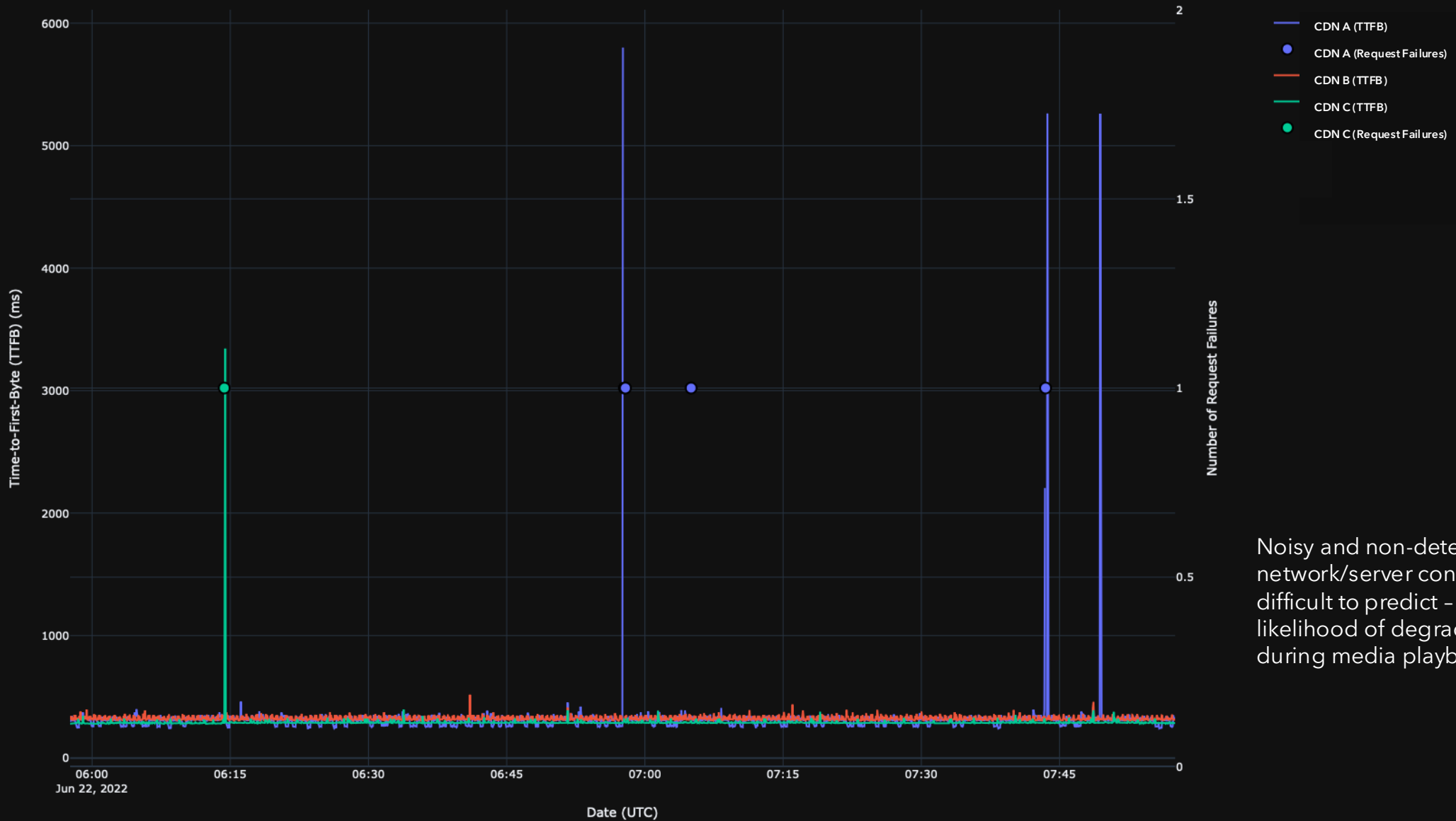


MEDIA CODING AND DISTRIBUTION - TODAY

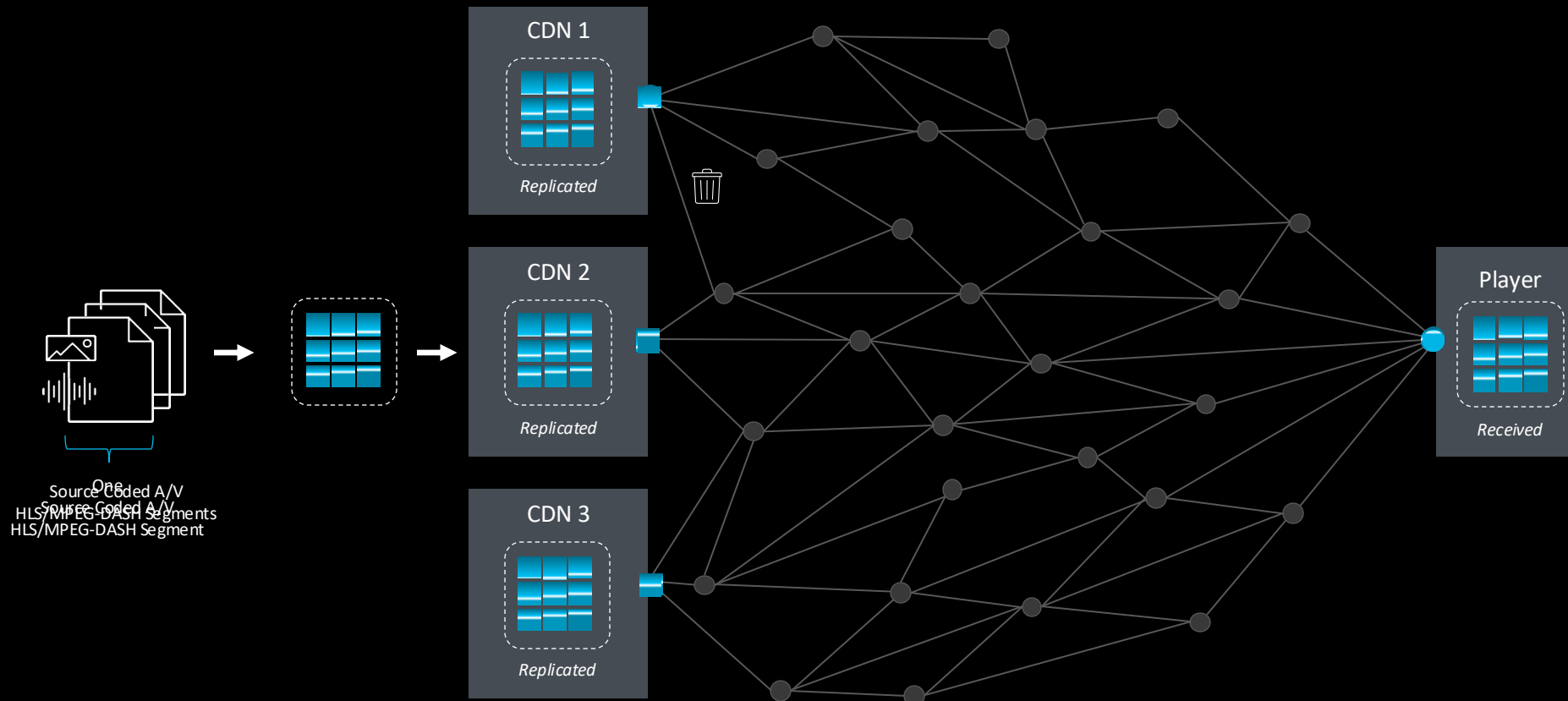


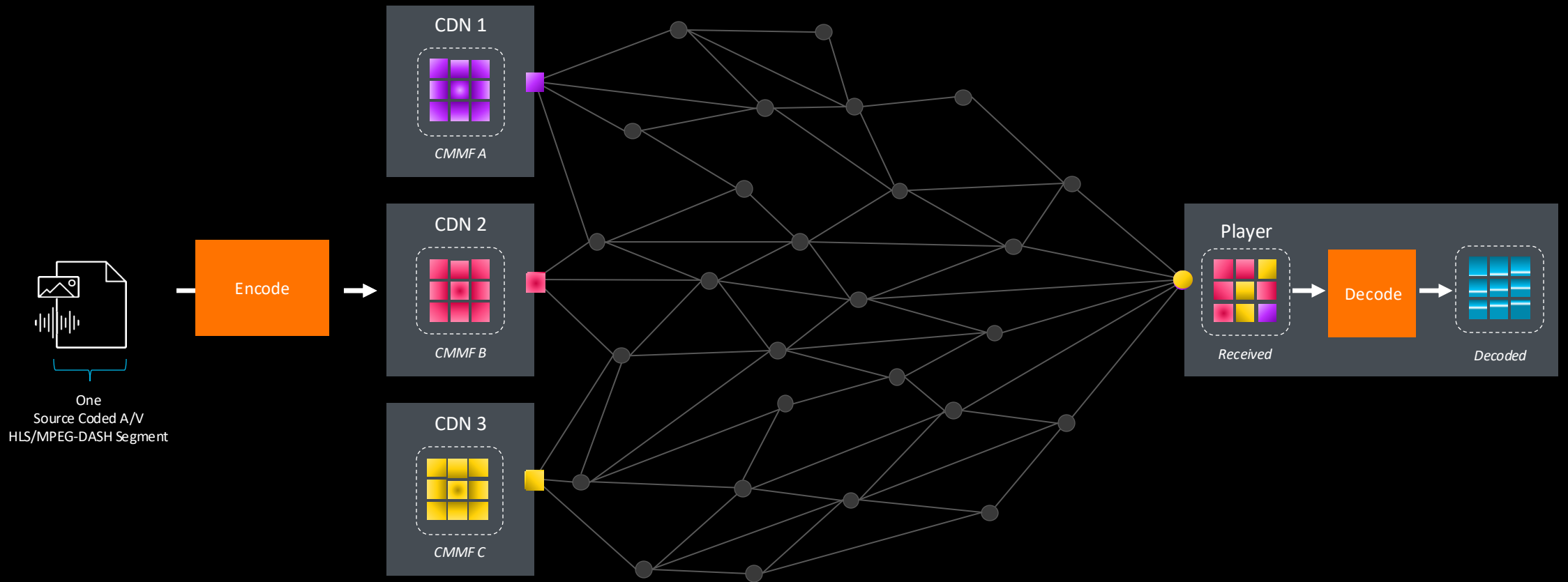


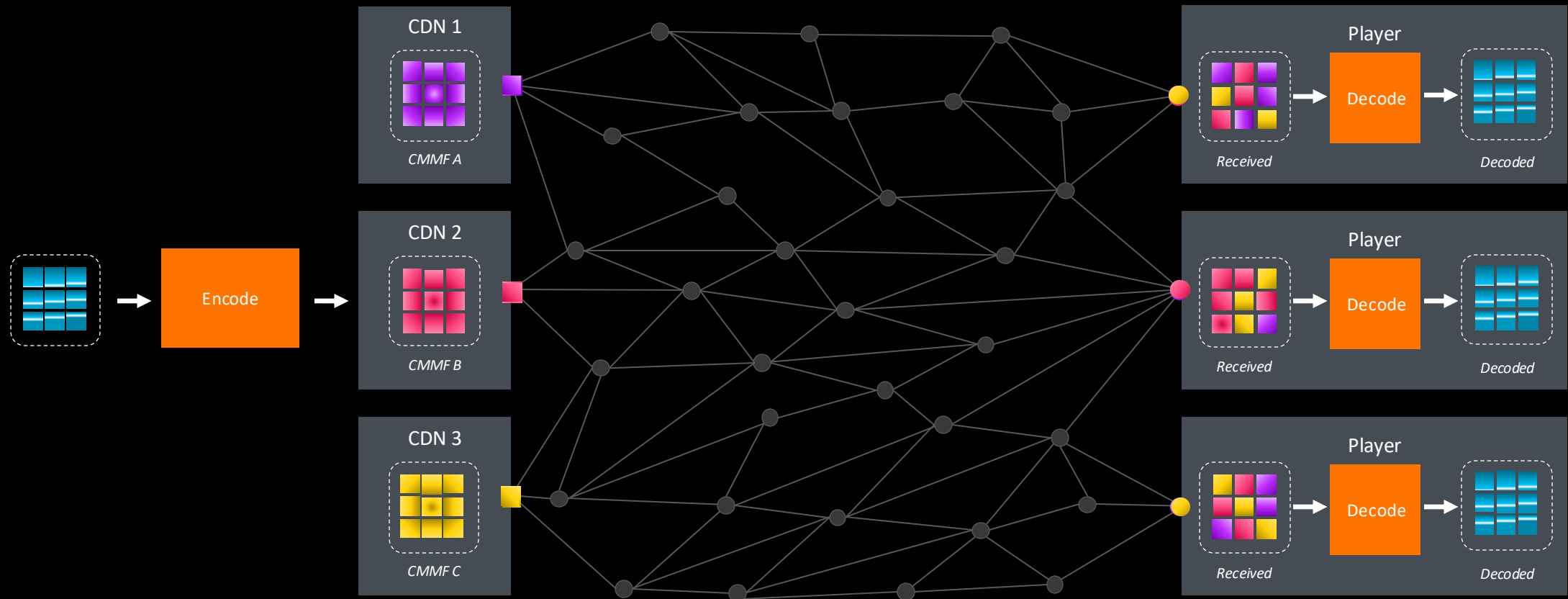
Noisy and non-deterministic network/server conditions are difficult to predict - increasing the likelihood of degraded QoE during media playback.

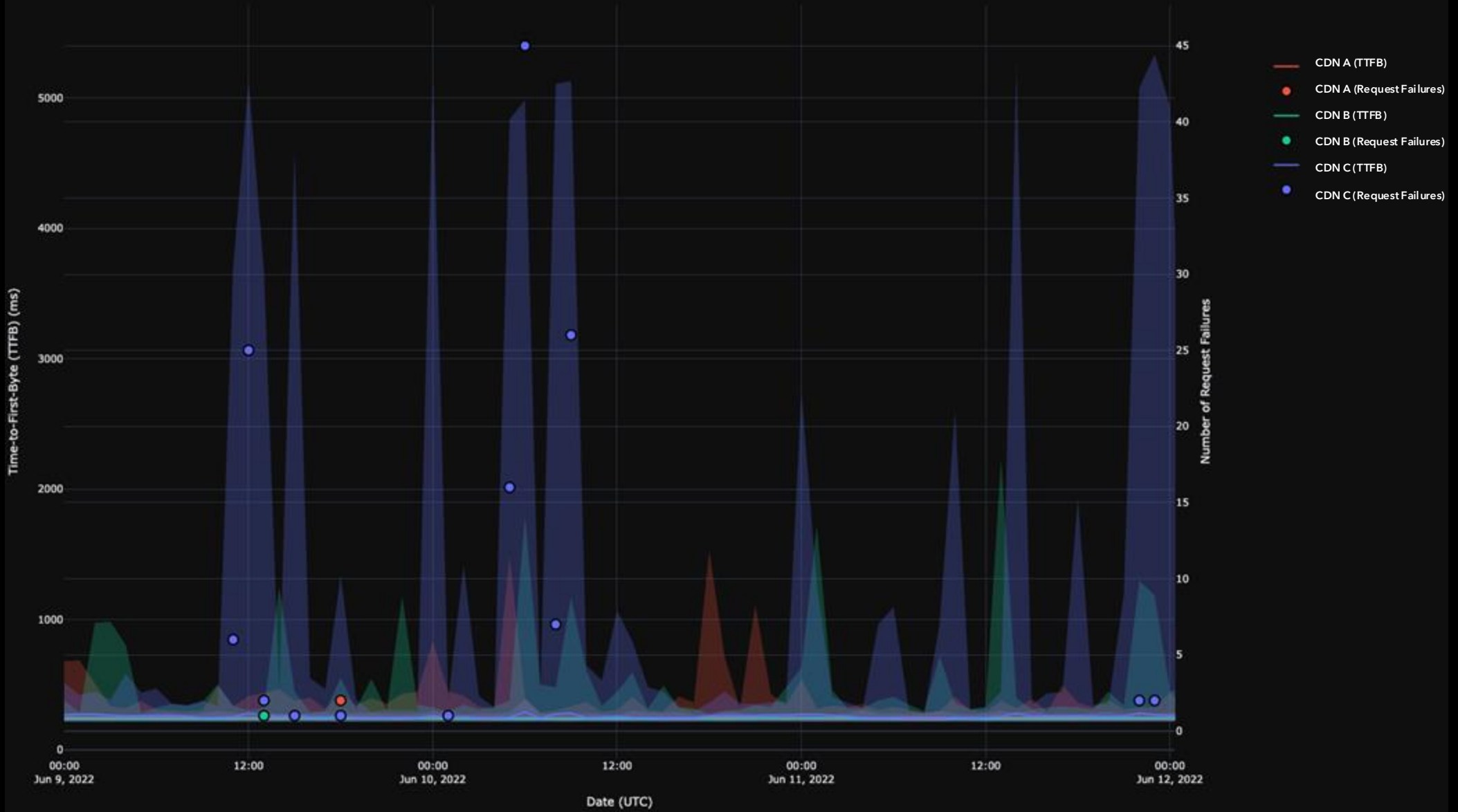


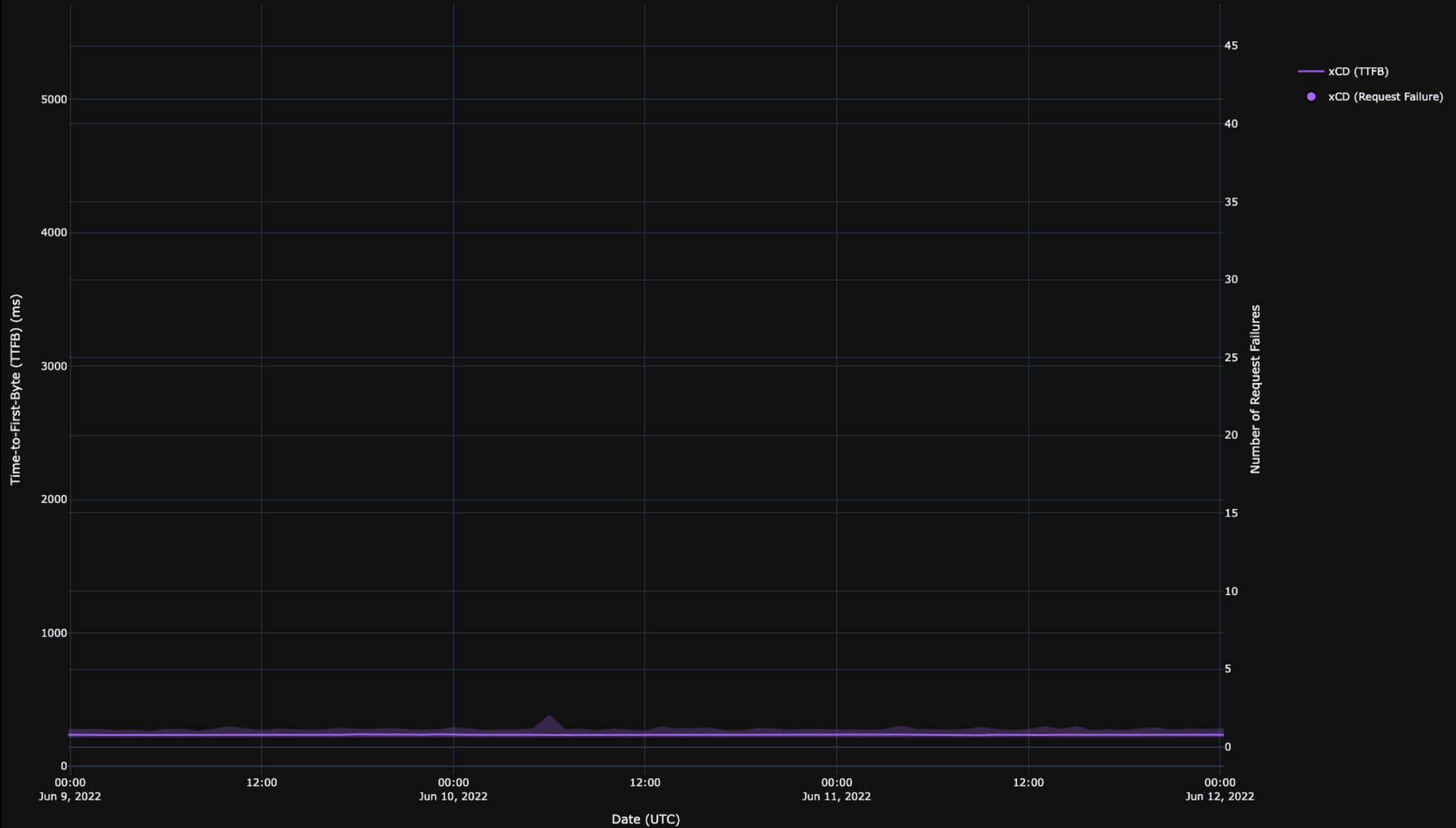
Noisy and non-deterministic network/server conditions are difficult to predict - increasing the likelihood of degraded QoE during media playback.

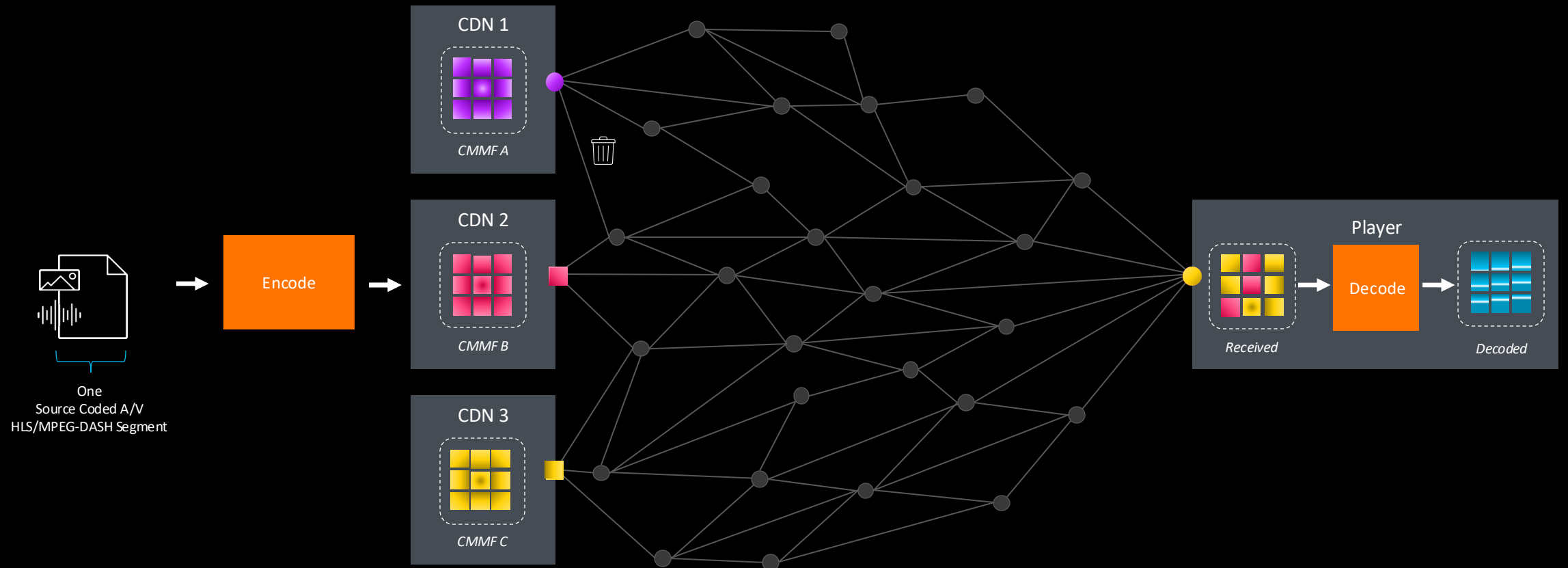




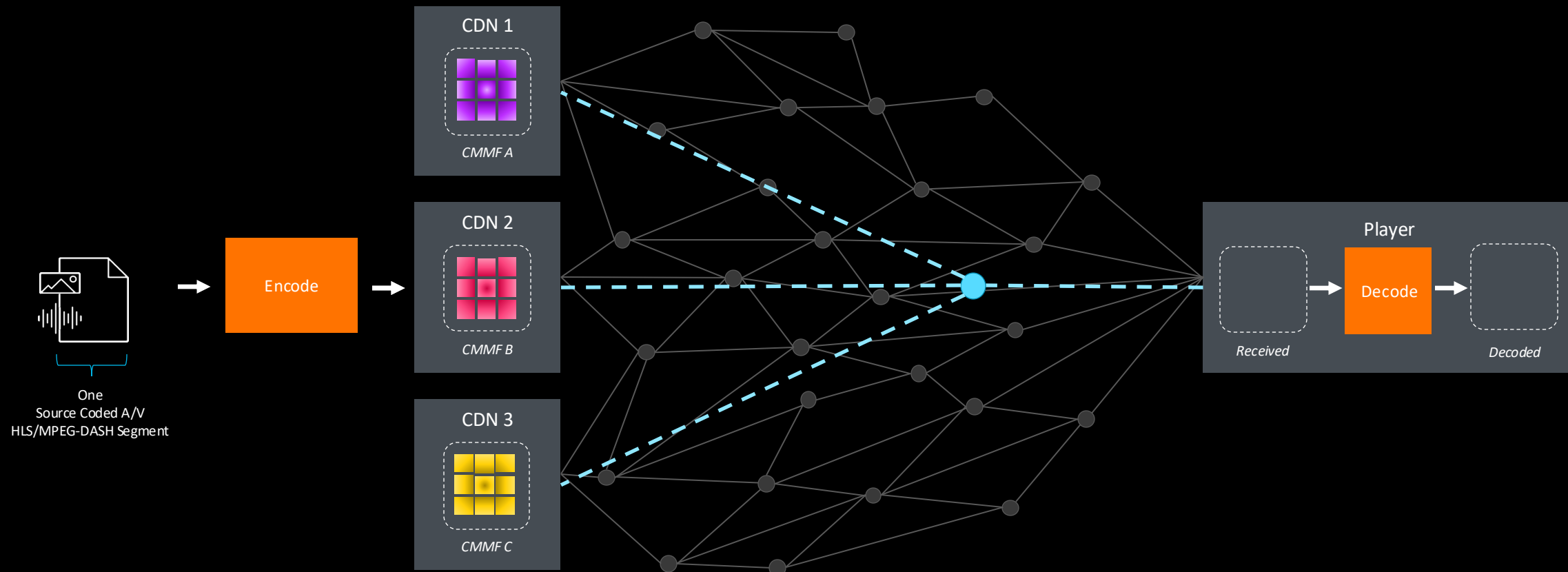


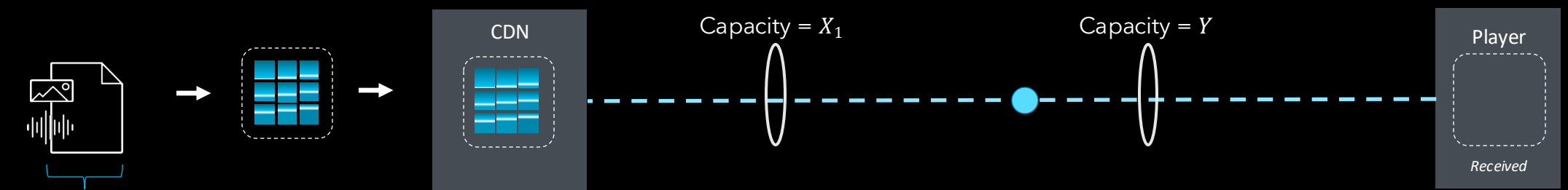
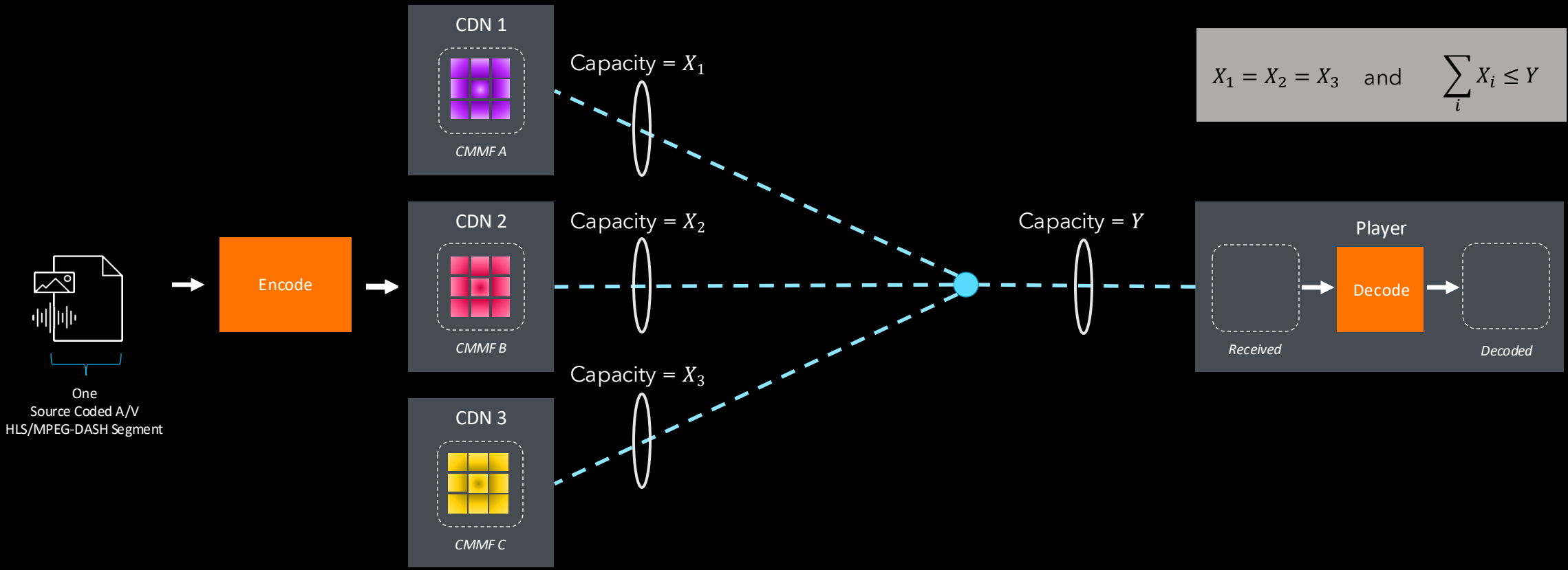




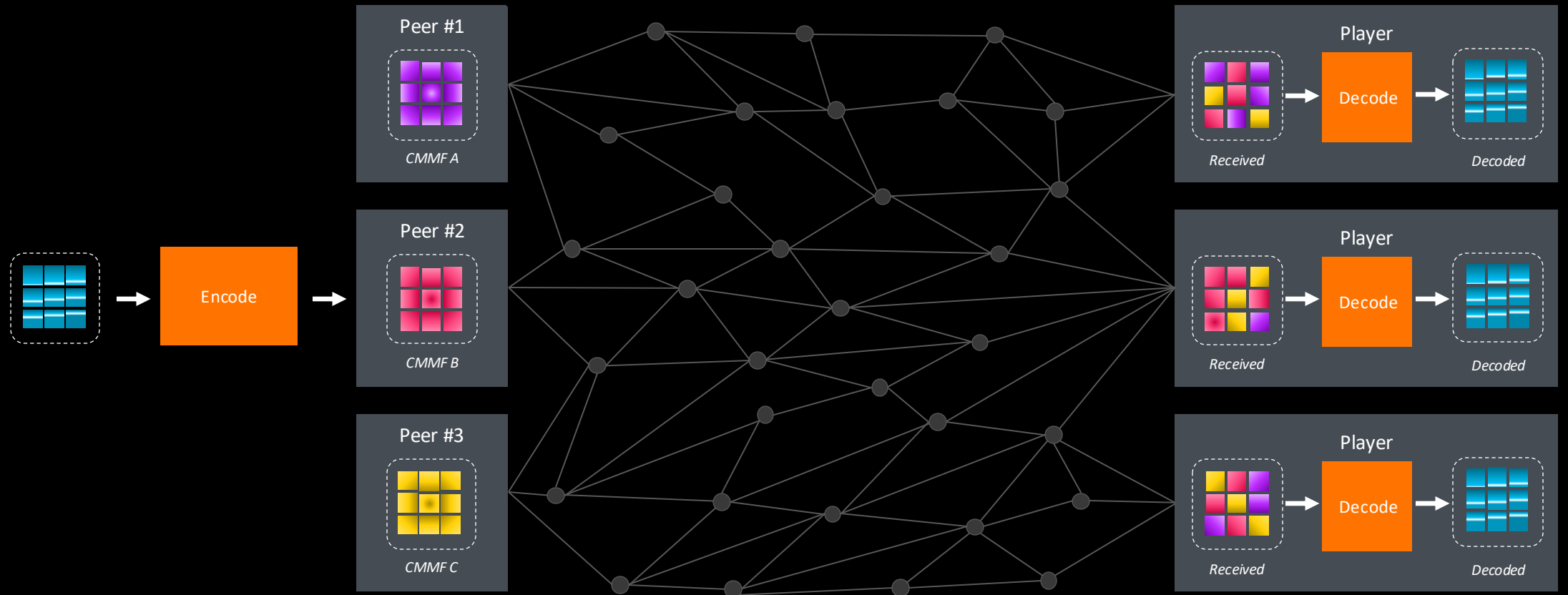


Example: One CDN fails to respond





Example: Path Bonding





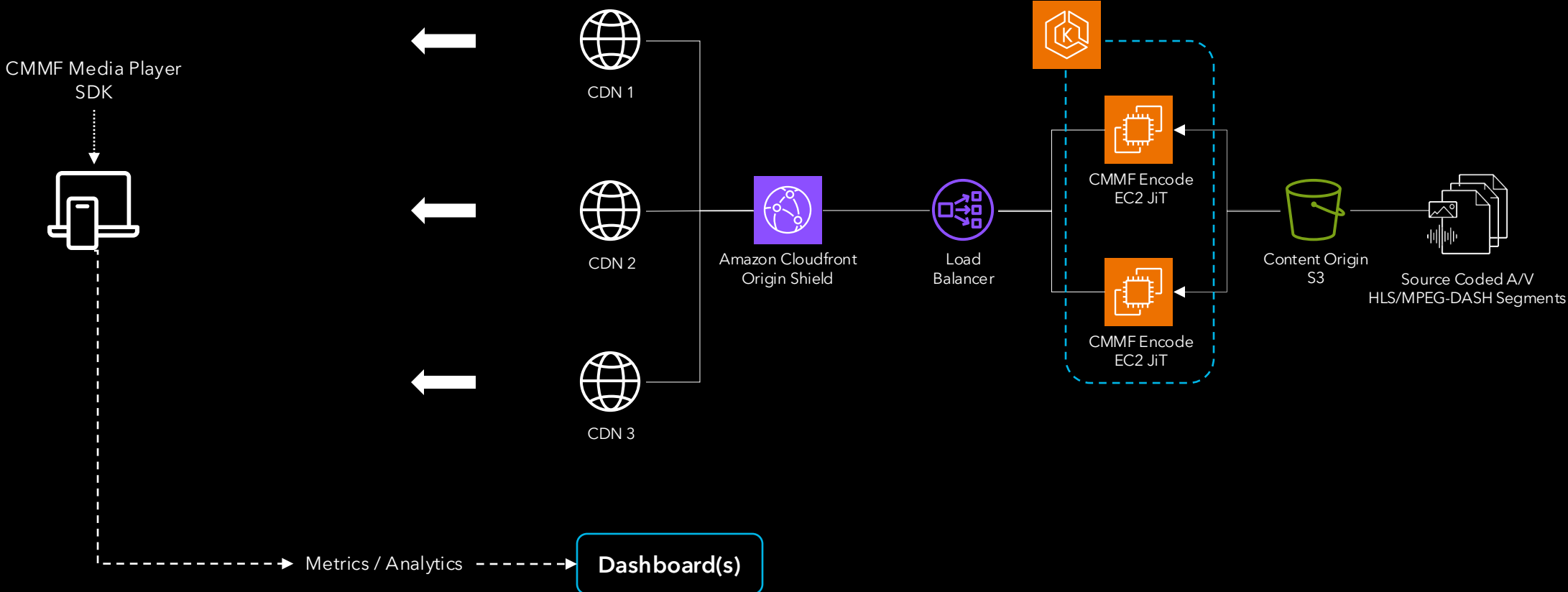
CODED MULTI-SOURCE DELIVERY IN THE REAL-WORLD

Overview

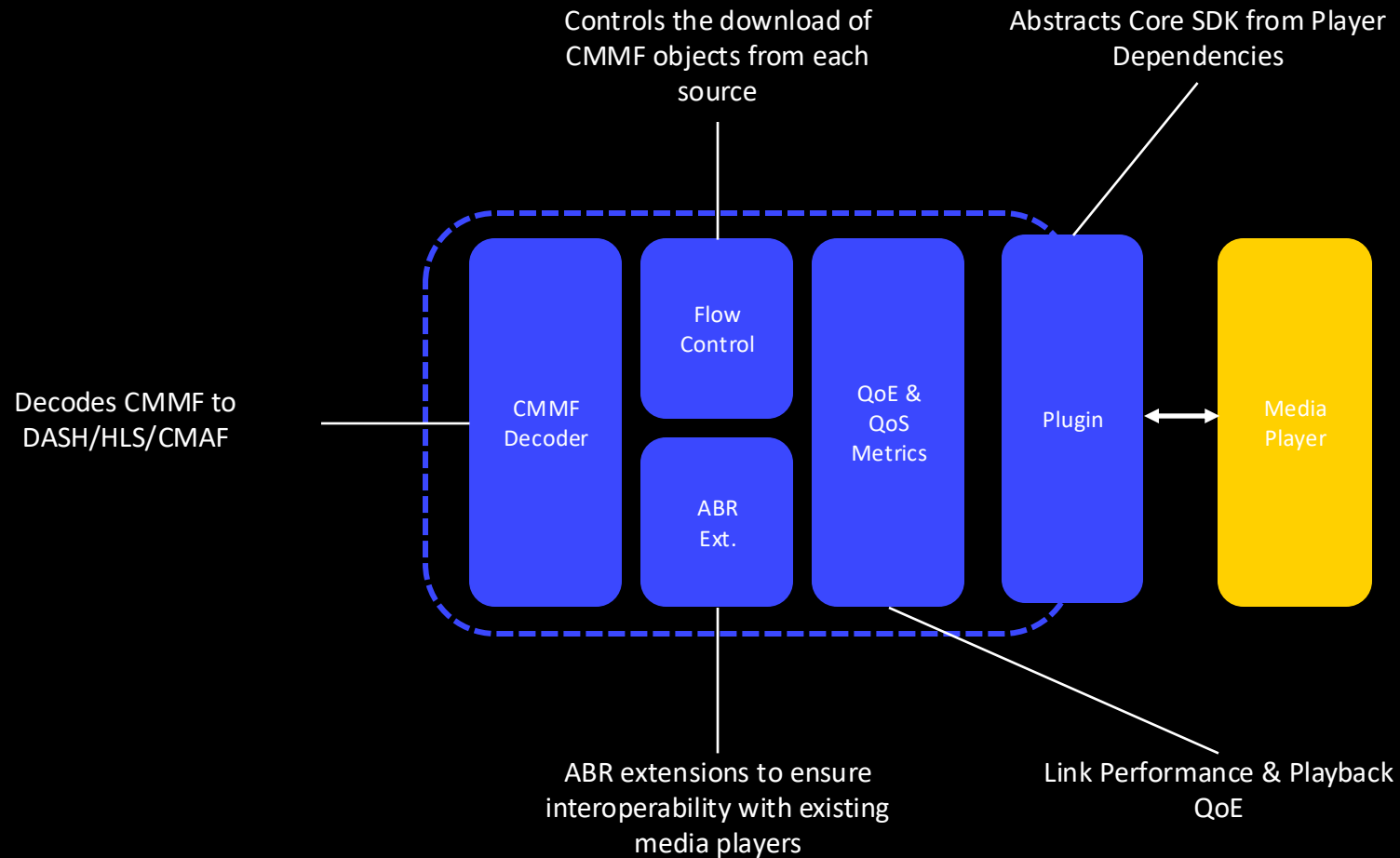
	Number of Sessions	Unique Viewers	Unique Countries	Unique ISPs	Hours Watched
Android/FireOS Mobile (WiFi)	477,623	35,794	194	3,643	126,798.61
Android/FireOS Mobile (Cell)	95,266	12,085	161	788	24,607.72
Android/FireOS TV	596,949	22,086	144	1,550	225,700.84
iOS Mobile (WiFi)	113,108	11,620	155	1,591	30,033.75
iOS Mobile (Cell)	8,173	1,569	99	308	1,728.13
Total	1,291,119	73,048	205	4,570	408,869.05

Coded multi-source delivery tested from Sep. 2022 – Sep 2023 within a Global OTT sVoD service with >20 million subscribers

E2E Architecture

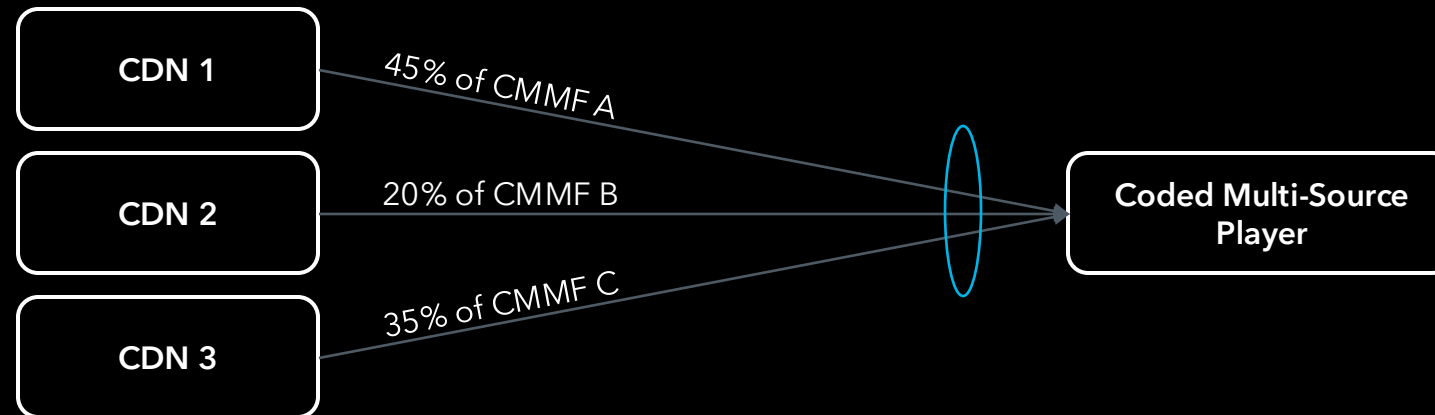


Client SDK Architecture



Flow Control

Goal: Download just enough CMMF-encoded content from any combination of sources to decode and limit excessive CDN egress



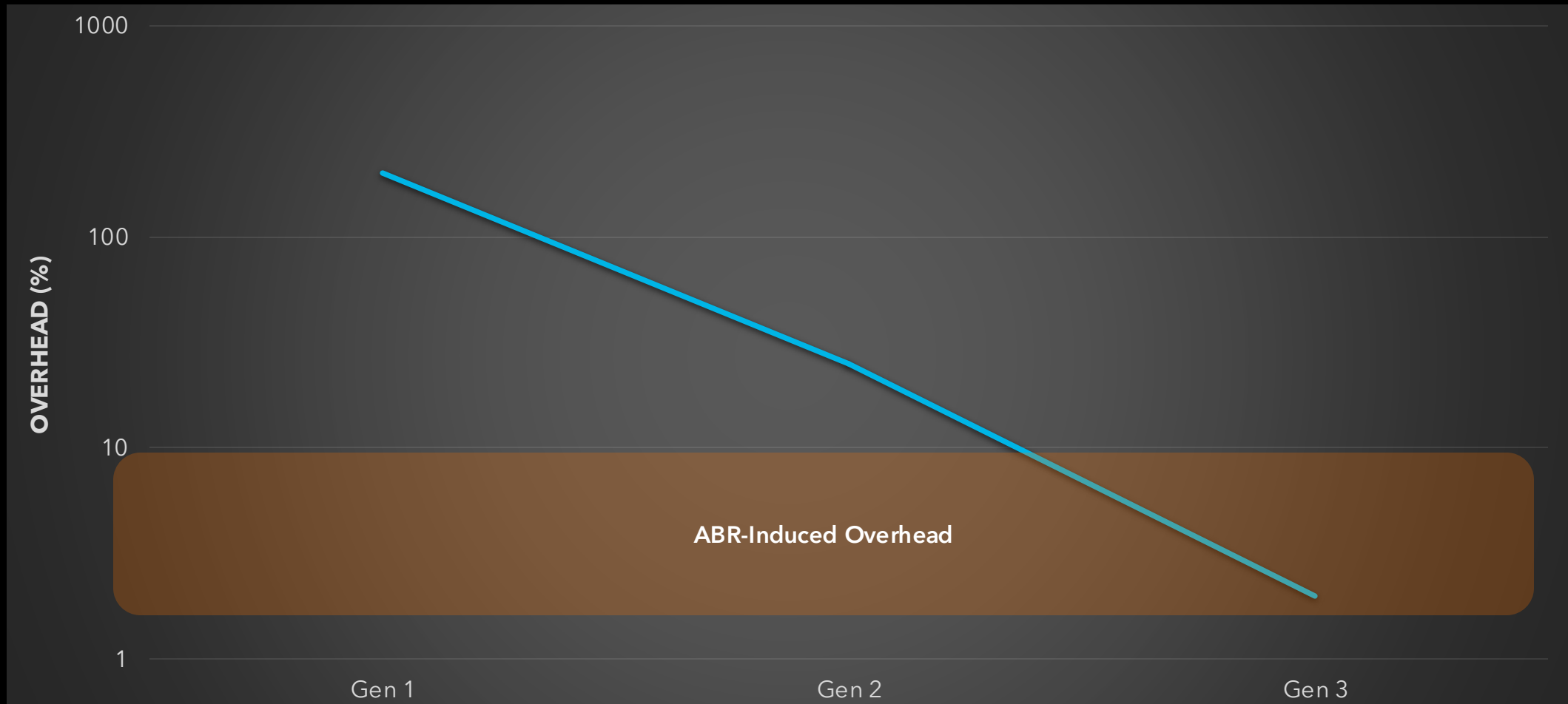
Amount of CMMF-encoded content required \approx Original size of segment

Flow Control

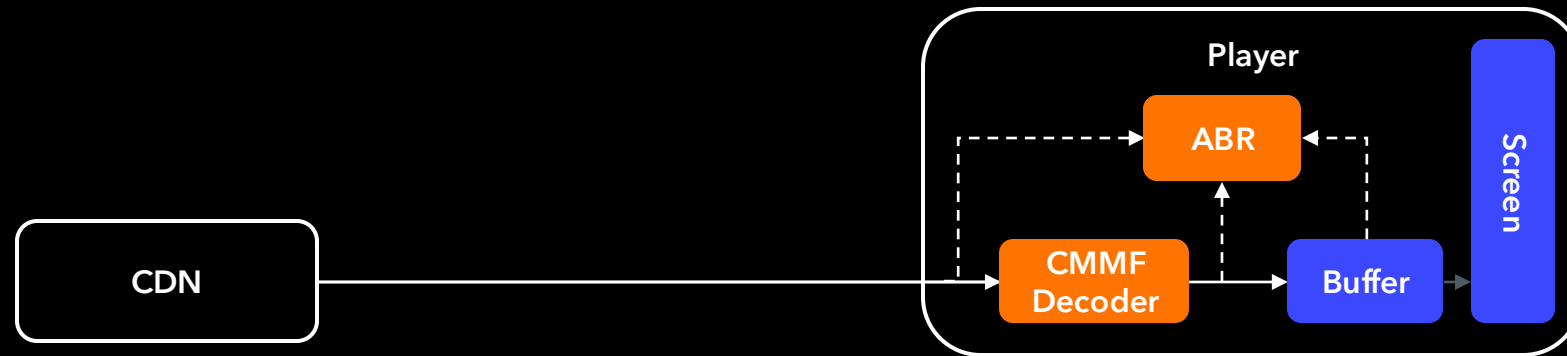
Challenges

- Existing protocols (e.g., HTTP, TCP, etc.) and HTTP servers generally do not react well when requests are cancelled
- Differences in how each protocol functions necessitate different strategies
- Each CDN has their own idiosyncrasies making a “one-size-fits-all” approach difficult

Flow Control



Adaptive Bitrate Policy Considerations



Real-World Performance

Android/FireOS Mobile (WiFi)

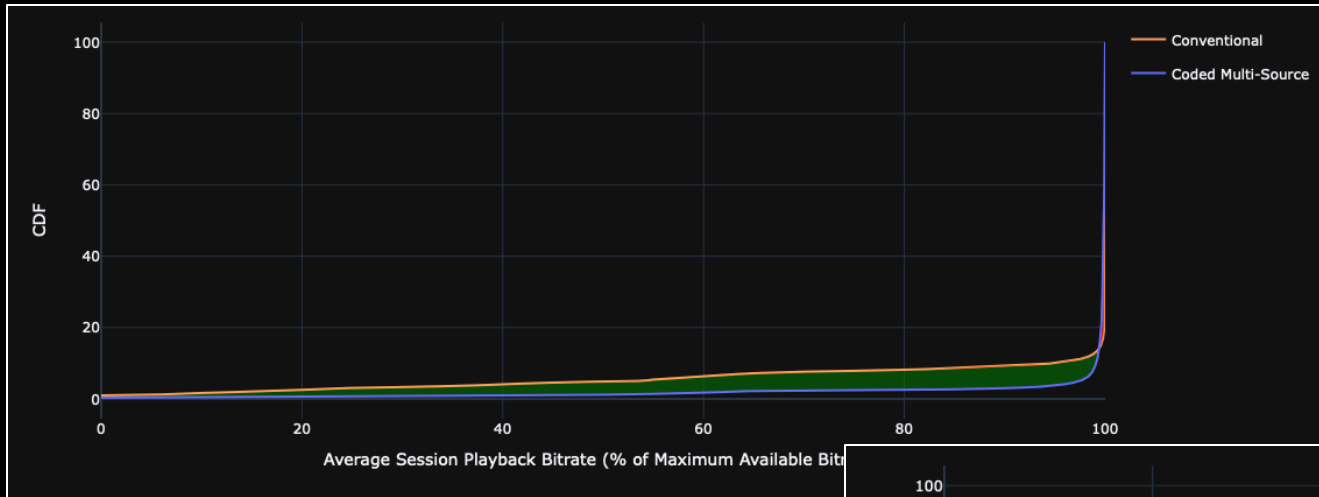
	Playback Completion (%)	Content Normalized Average Bitrate (%)	Startup Time (s)	Connection-Induced Rebuffering Ratio (%)	Startup Failure Rate (%)
Conventional Multi-CDN	48.2 ± 43.47	94.56 ± 18.55	2.01 ± 10.22	0.13 ± 1.15	0.9
Coded Multi-Source	55.78 ± 42.54	98.14 ± 9.95	1.4 ± 9.88	0.07 ± 0.55	0.16
Difference	7.59 ± 60.82	3.58 ± 21.05	-0.61 ± 14.21	-0.06 ± 1.28	-0.74
% Gain	15.74	3.79	-30.25	-43.65	-82.17

Android/FireOS Mobile (Cellular)

	Playback Completion (%)	Content Normalized Average Bitrate (%)	Startup Time (s)	Connection-Induced Rebuffering Ratio (%)	Startup Failure Rate (%)
Conventional Multi-CDN	46.3 ± 42.03	83.7 ± 28.08	3.32 ± 10.24	0.28 ± 1.78	0.46
Coded Multi-Source	56.23 ± 40.34	94.31 ± 16.23	1.83 ± 9.34	0.19 ± 1.17	0.07
Difference	9.93 ± 58.26	10.61 ± 32.43	-1.49 ± 13.86	-0.09 ± 2.13	-0.4
% Gain	21.45	12.67	-44.85	-32.23	-85.82

Tables show mean plus/minus one standard deviation, Conventional Multi-CDN Avg Session Cache Hit Rate: >95%, Coded Multi-Source Avg Session Cache Hit Rate: >50%

Content Normalized Average Playback Bitrate

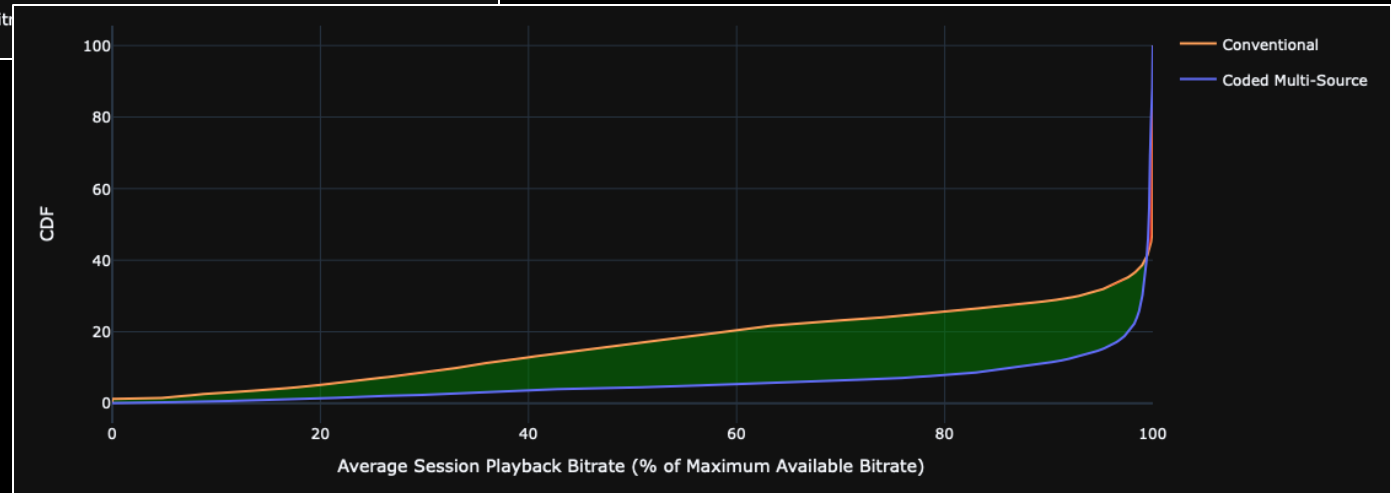


Android/FireOS Mobile (WiFi)

Maximum Possible Bitrate (from manifest):
 4.07 ± 2.26 Mbps

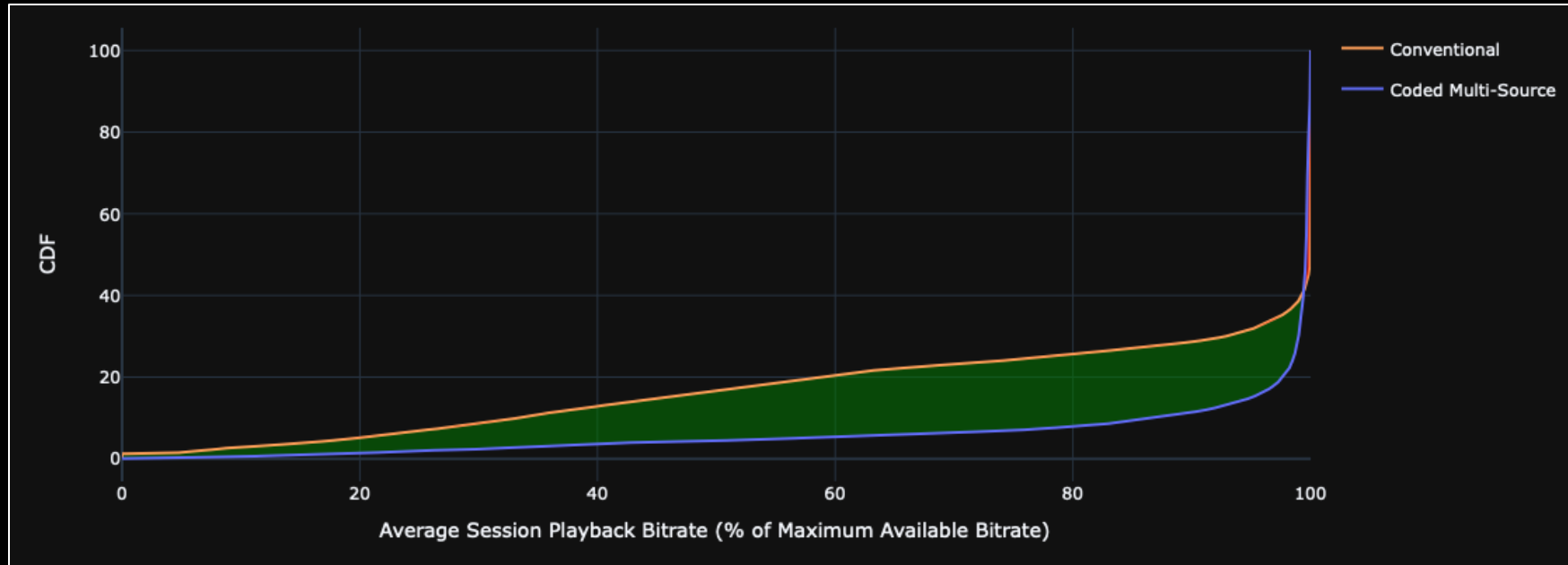
80% of the sessions had a maximum possible
bitrate less than 6 Mbps (per the sessions' manifest)

Android/FireOS Mobile (Cellular)



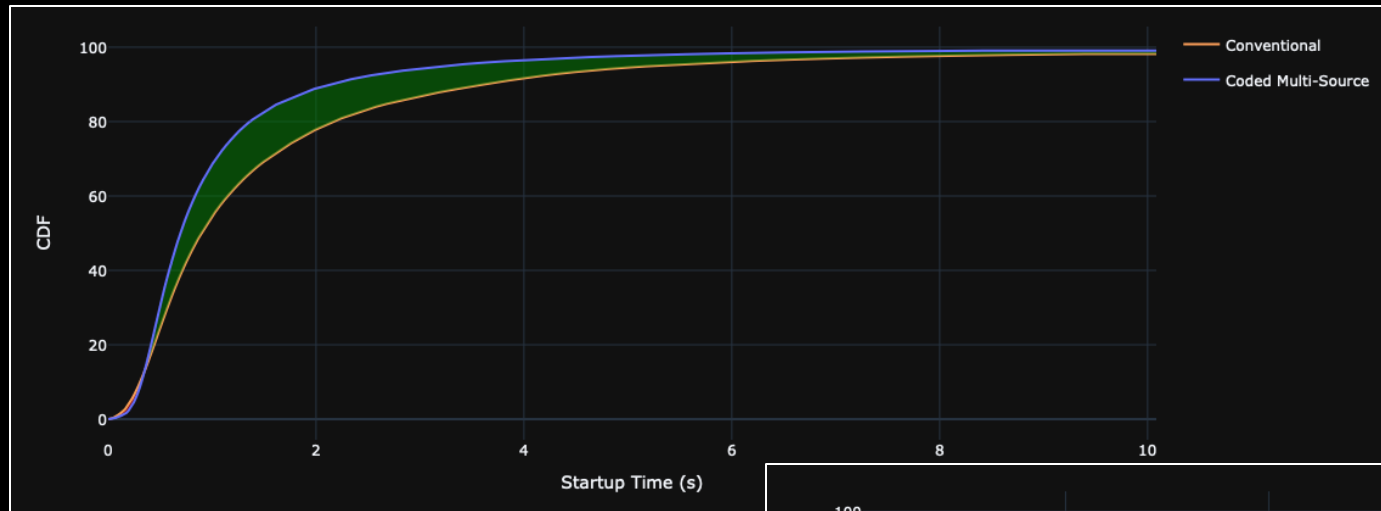
Content Normalized Average Playback Bitrate

Android/FireOS Mobile (Cellular)



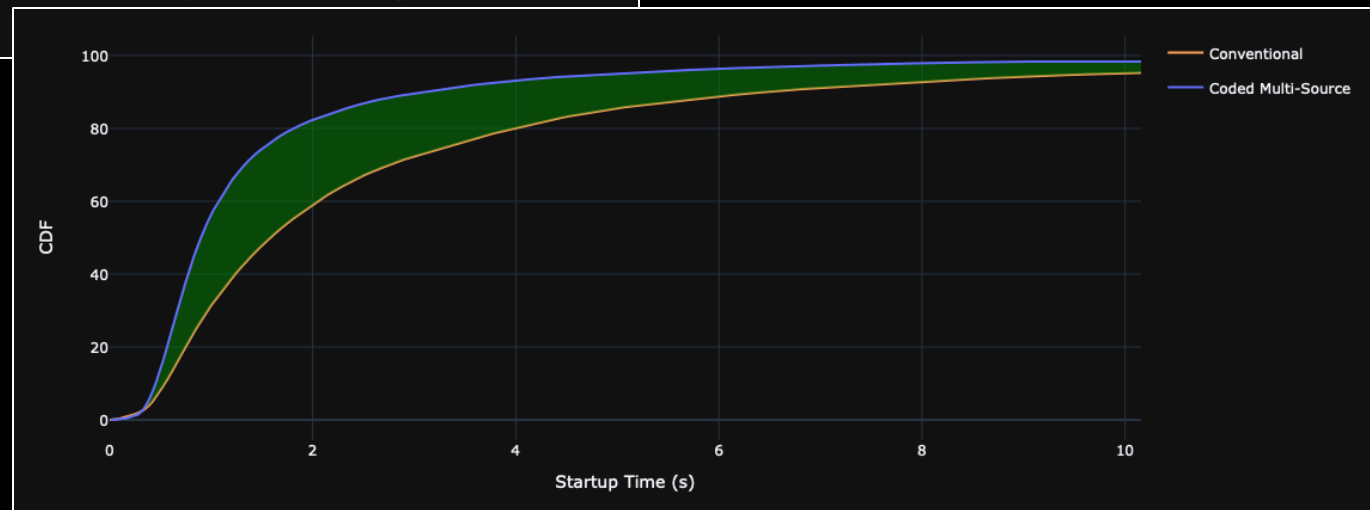
	< 20%	< 40%	< 60%	< 80%	< 90%	< 95%
Conventional Multi-CDN	5.13	12.85	20.56	25.58	28.66	31.65
Coded Multi-Source (xCD)	1.29	3.69	5.43	7.92	11.29	14.96
Difference	-3.84	-9.16	-15.13	-17.66	-17.37	-16.69
% Gain	-74.85	-71.28	-73.59	-69.04	-60.61	-52.73

Startup Time



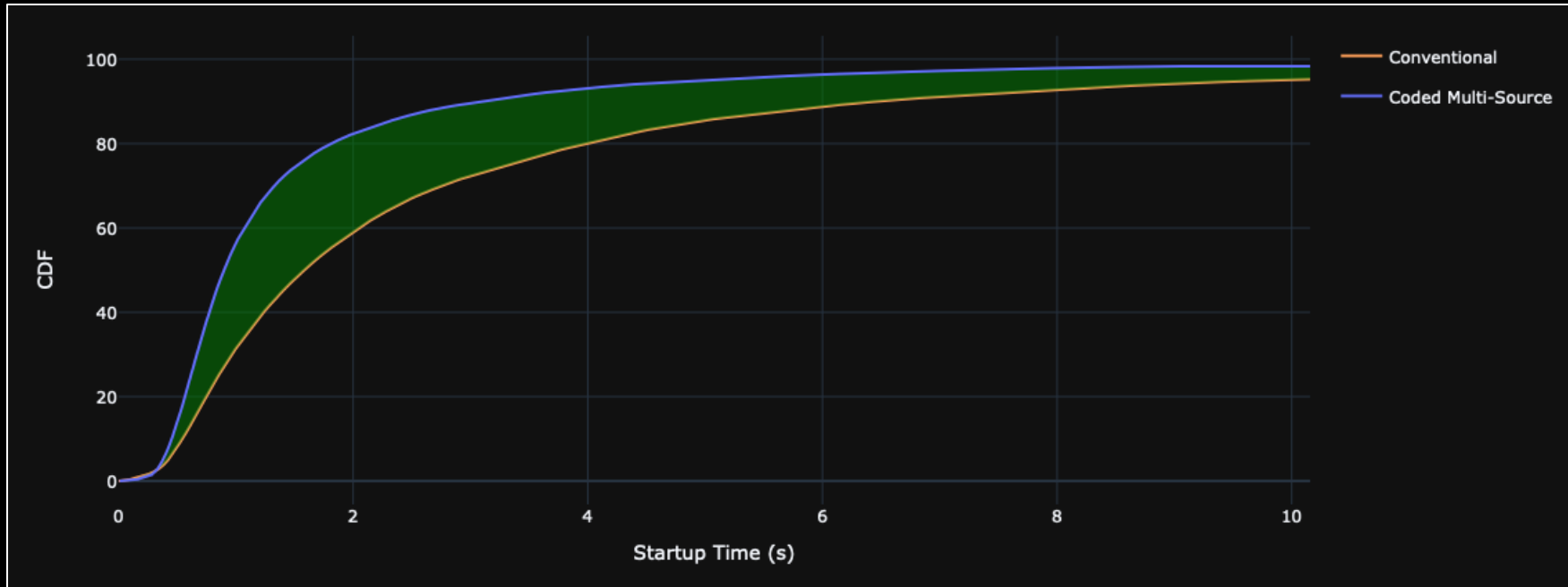
Android/FireOS Mobile (WiFi)

Android/FireOS Mobile (Cellular)



Startup Time

Android/FireOS Mobile (Cellular)



	> 500 ms	> 1000 ms	> 2000 ms	> 3000 ms	> 4000 ms	> 8000 ms
Conventional Multi-CDN	91.70	68.61	40.86	27.59	19.80	7.05
Coded Multi-Source (xCD)	85.96	43.66	17.60	10.36	6.89	2.11
Difference	-5.74	-24.95	-22.40	-17.23	-12.91	-4.94
% Gain	-6.26	-36.36	-54.82	-62.45	-65.20	-70.07

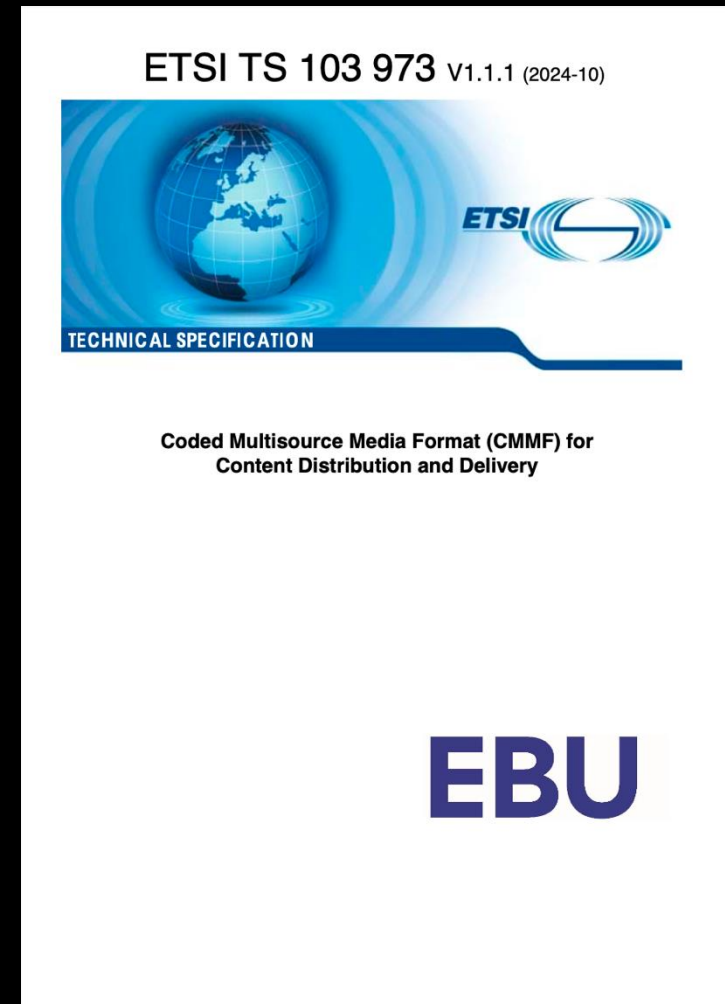


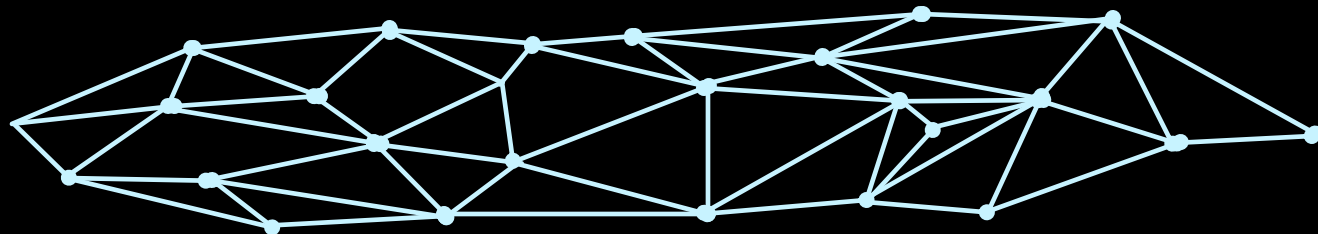
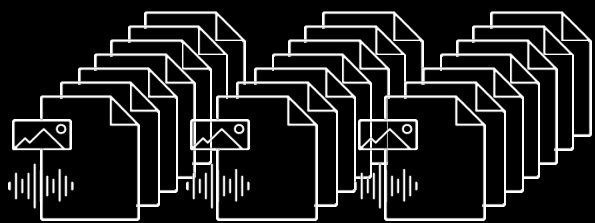
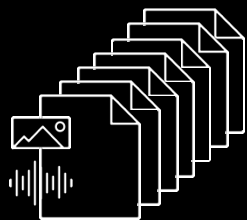
STANDARDIZATION

Coded Multi-Source Media Format (CMMF)

ETSI TS 103 973

- A reference architecture for the generation and distribution of multisource coded media to support delivery from multiple sources to a single receiver
- A definition of the necessary configuration information for the receiver to access the multiple sources and re-generate the original source content
- An extensible and coded media format including syntax and semantics (xCD-1, RaptorQ, etc.) that can be used by the receiver to collect data from multiple sources and reconstruct the original content
- Deployment examples and guidelines for the usage of the format in different scenarios, for example in video-on-demand streaming with DASH or large file distribution





Audio/Video Elementary Streams

Content Packaging

(e.g., ISO/BMFF/CMAF/HLS/MPEG-DASH)

Formats and schema to communicate how media should be treated. Little-to-no capability to communicate network-level information.

CMMF

Server/network and content packaging agnostic (works with any lower-layer protocol or higher-layer content packaging schema).

Network Delivery

(e.g., Network Caching, HTTP, TCP, QUIC, webRTC, FLUTE, etc.)

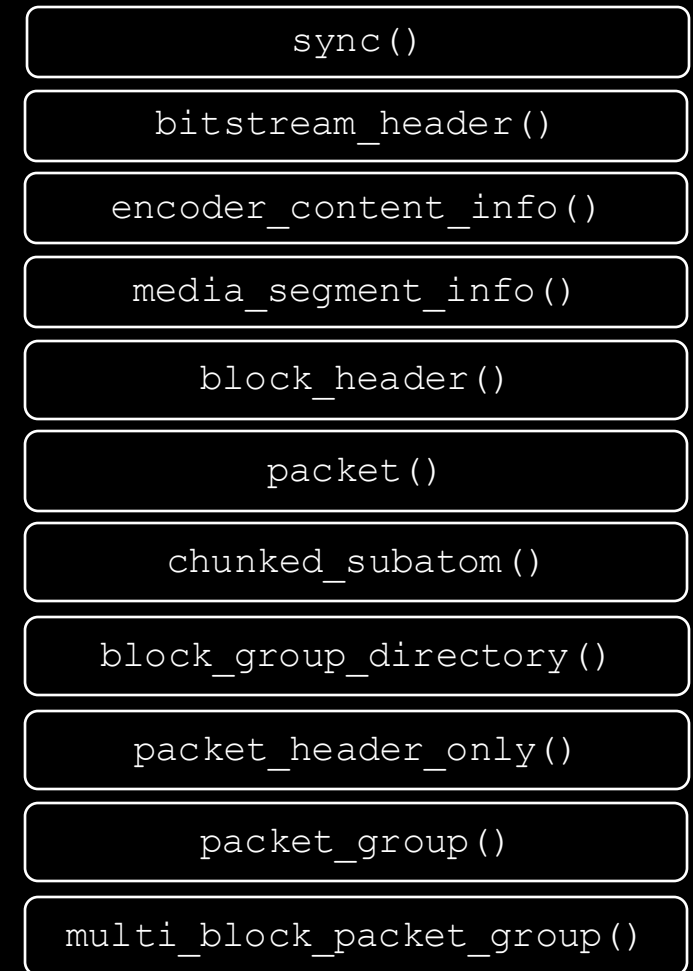
Caching and/or transport of content across networks. Systems and protocols may be agnostic of CMMF encoded content.

Conventional

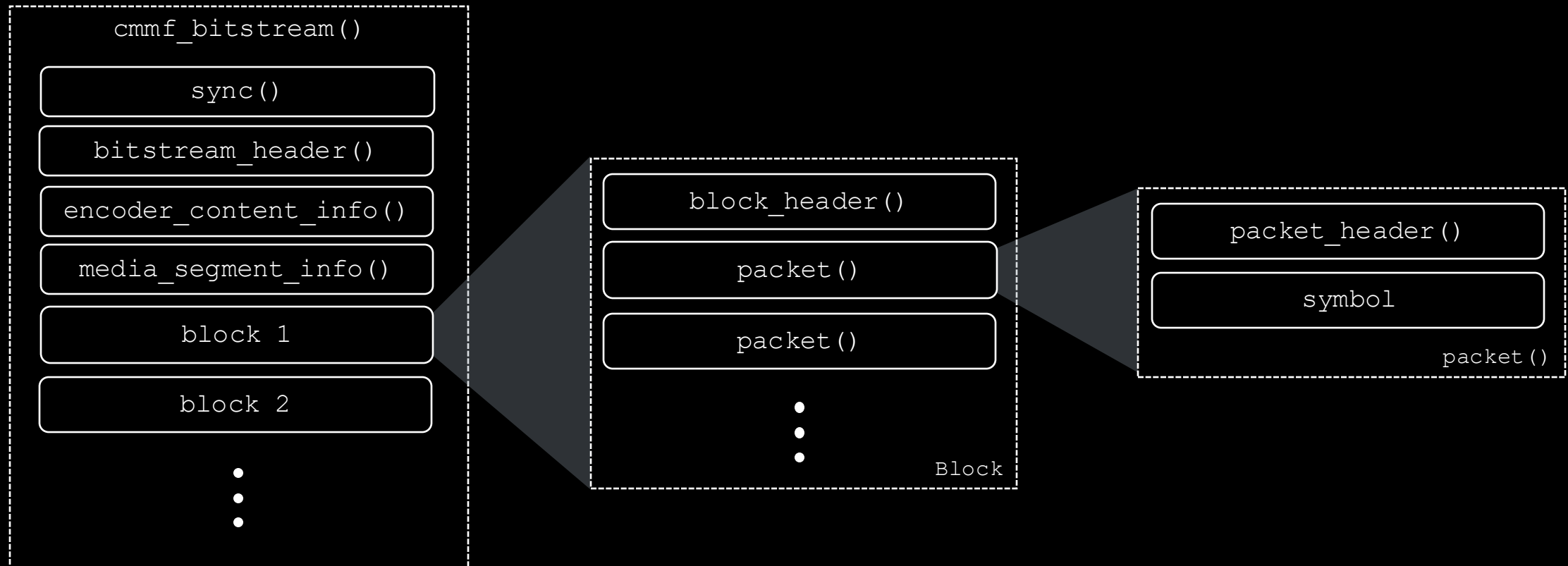
'replicates'

CMMF Bitstream / Object Construction

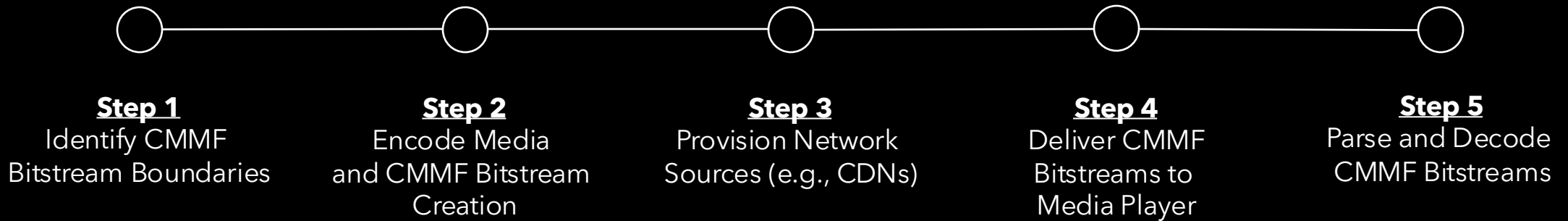
- Supports the transmission of media (e.g., MPEG-DASH, HLS, etc.) using network or FEC coding
 - xCD-1 (General Purpose/Build-Your-Own Linear Code)
 - Raptor (RFC 5053)
 - Reed-Solomon (RFC 5510)
 - RaptorQ (RFC 6330)
- Flexible and extensible atom structure that supports:
 - Different code types and code constructions
 - Multiple delivery scenarios and use cases
- 11 currently defined sub-atoms to support various use cases
 - Use of any one sub-atom is optional
 - Mix-and-match as necessary

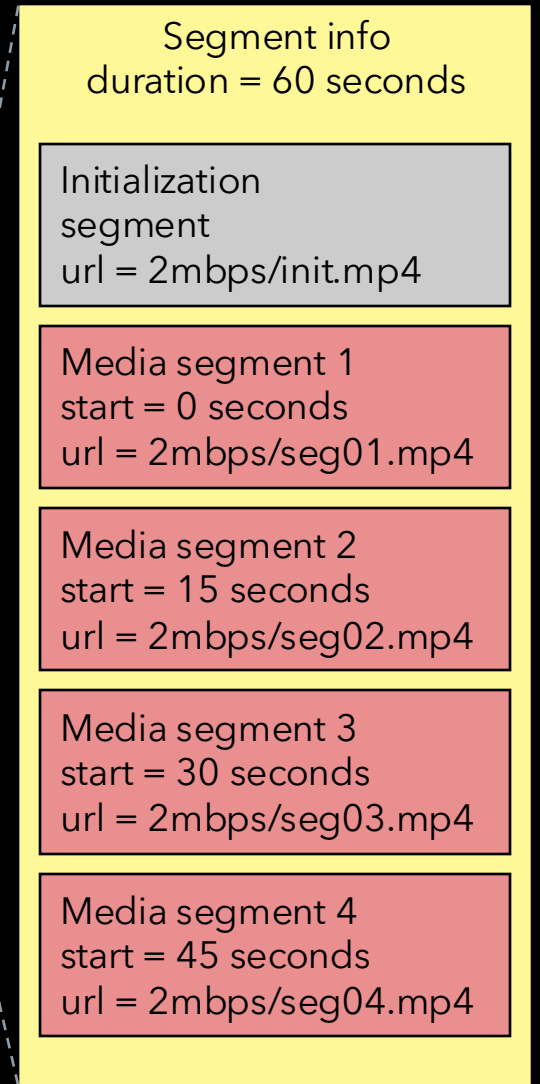
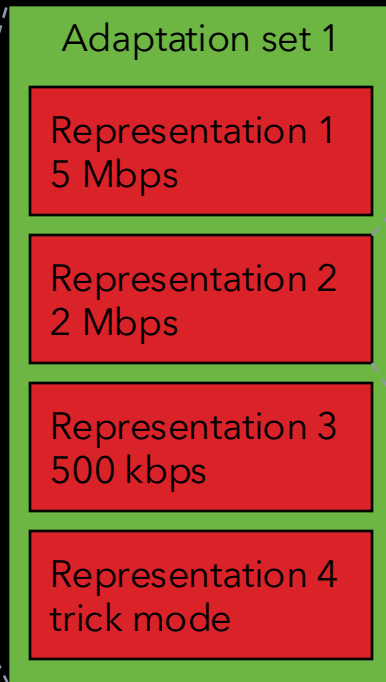
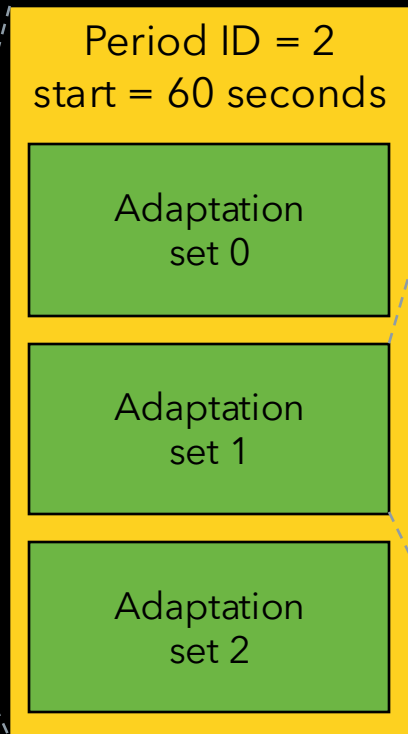
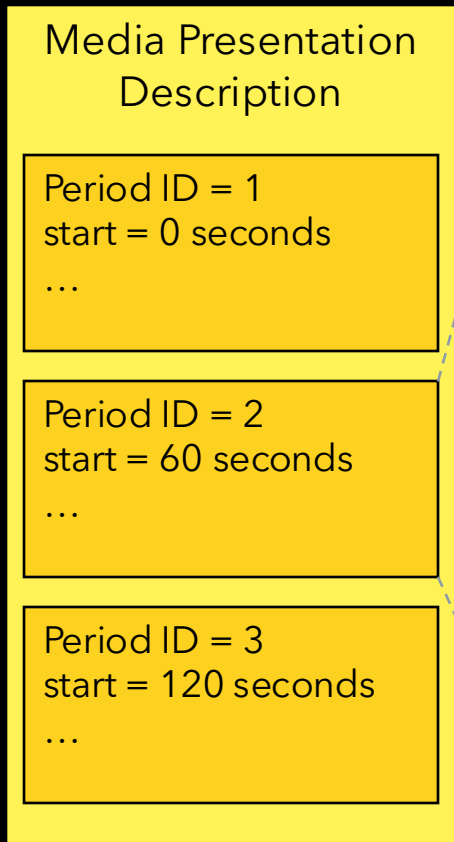


CMMF Bitstream / Object Data Model Example



MPEG-DASH + CMMF Example





Segment info
duration = 60 seconds

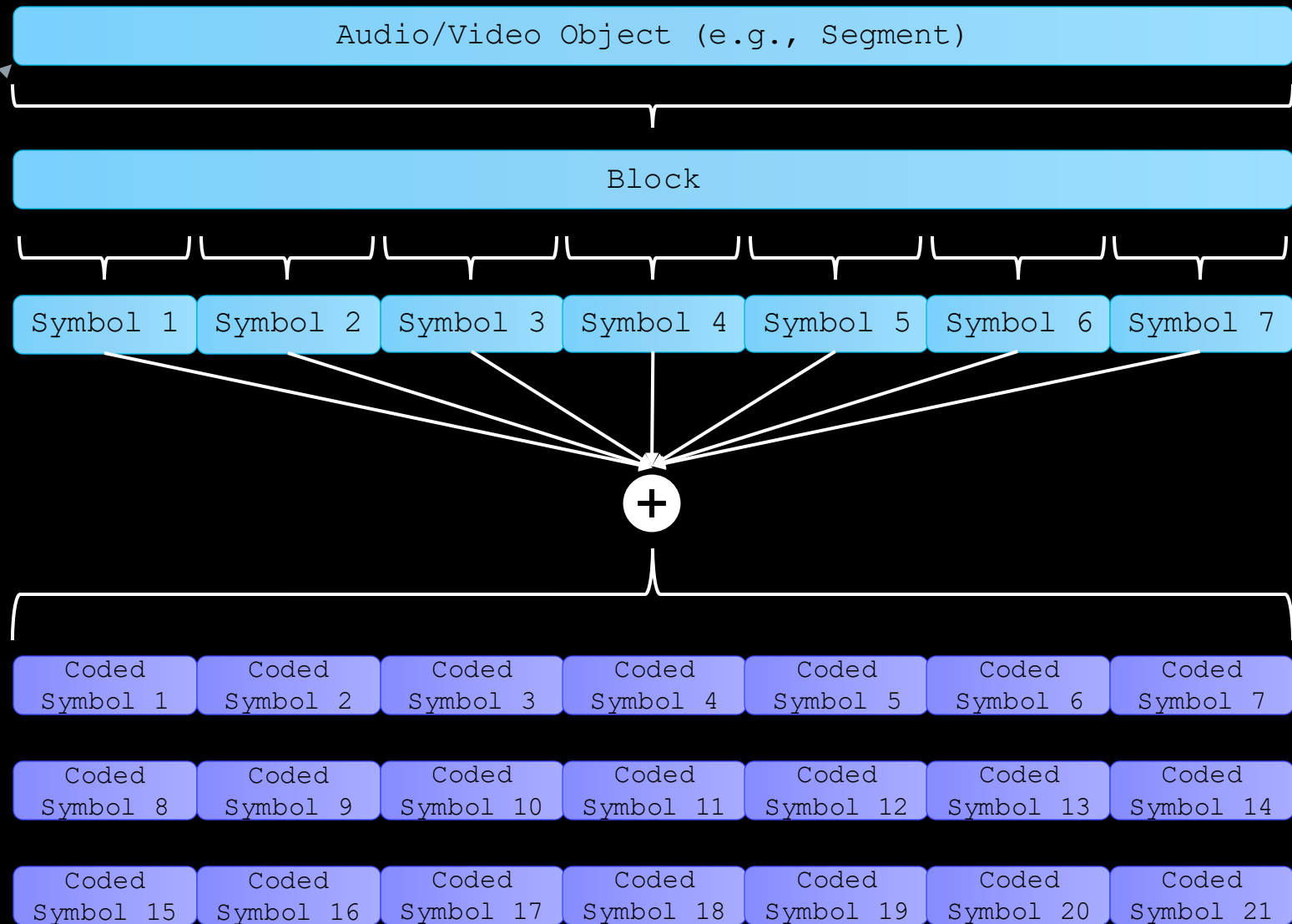
Initialization segment
url = 2mbps/init.mp4

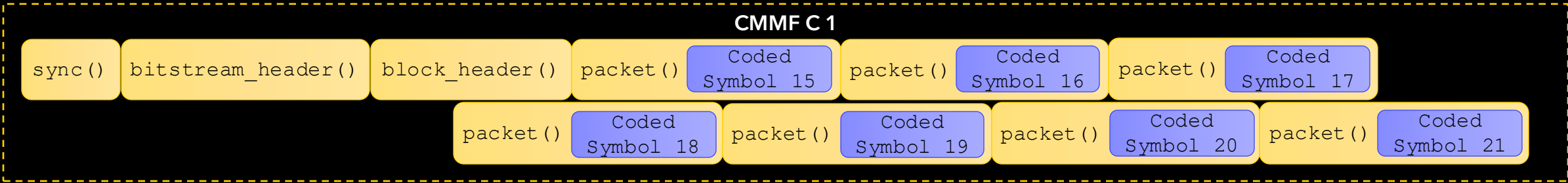
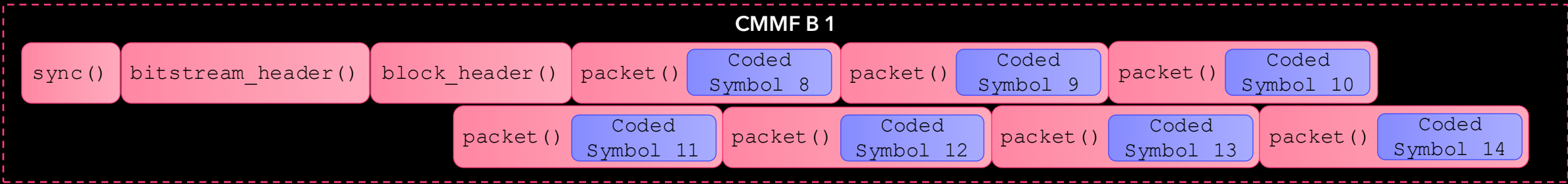
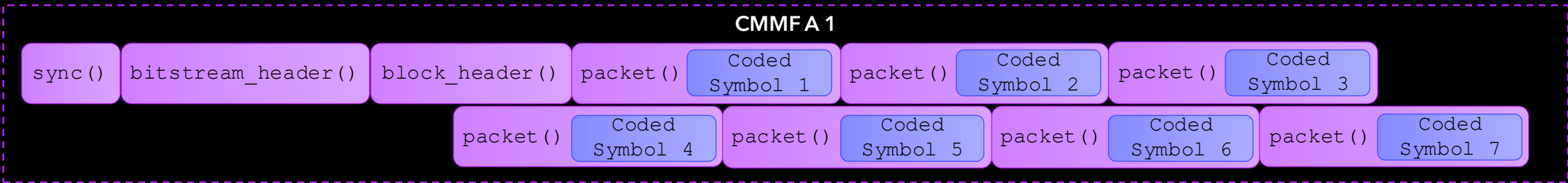
Media segment 1
start = 0 seconds
url = 2mbps/seg01.mp4

Media segment 2
start = 15 seconds
url = 2mbps/seg02.mp4

Media segment 3
start = 30 seconds
url = 2mbps/seg03.mp4

Media segment 4
start = 45 seconds
url = 2mbps/seg04.mp4





CMMF A 1

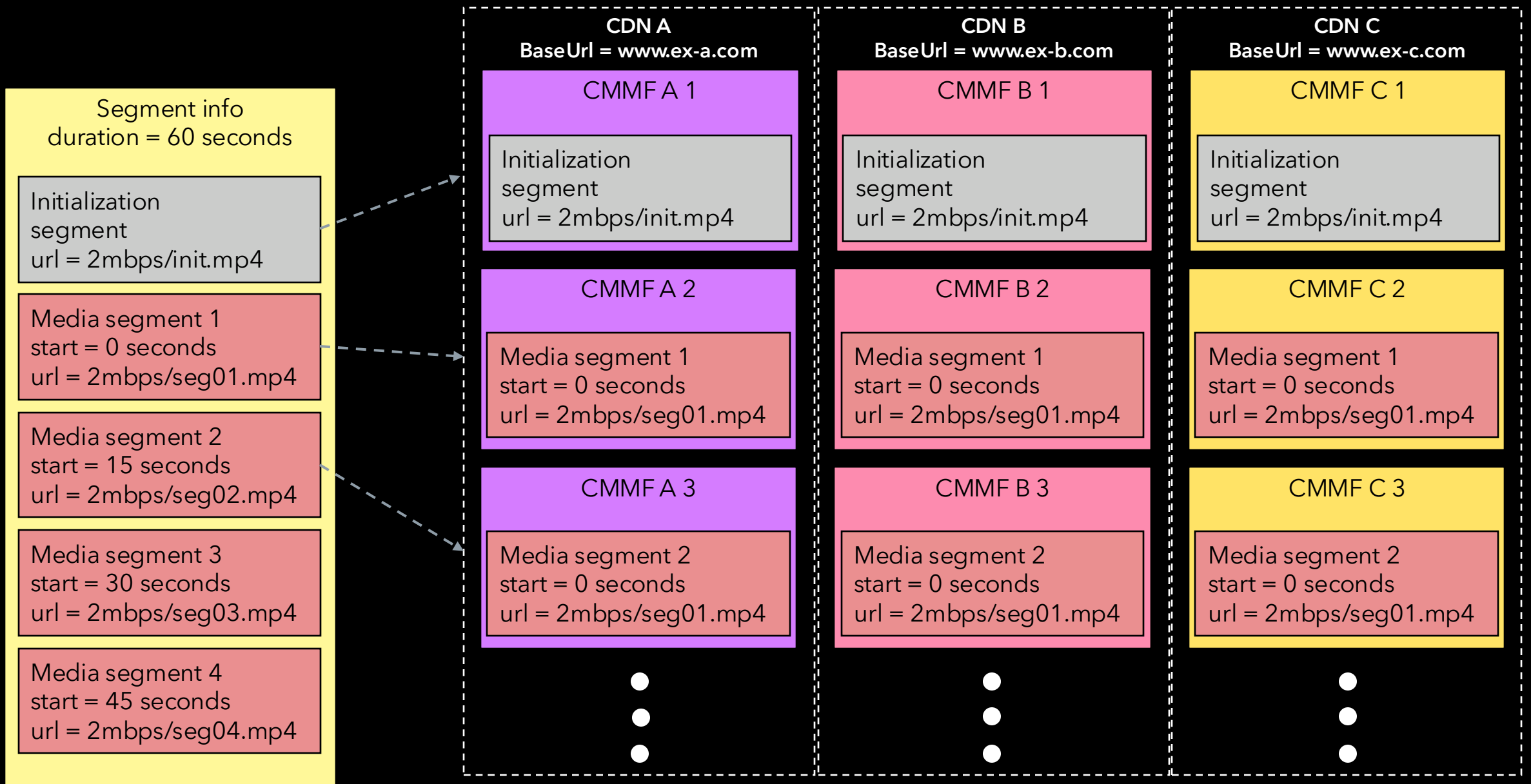
Initialization
segment
url = 2mbps/init.mp4

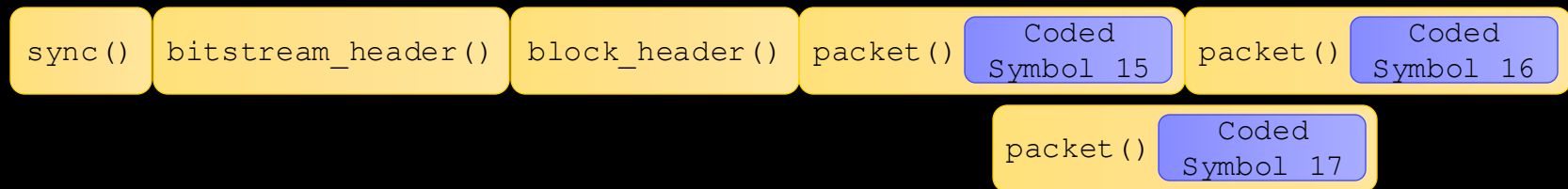
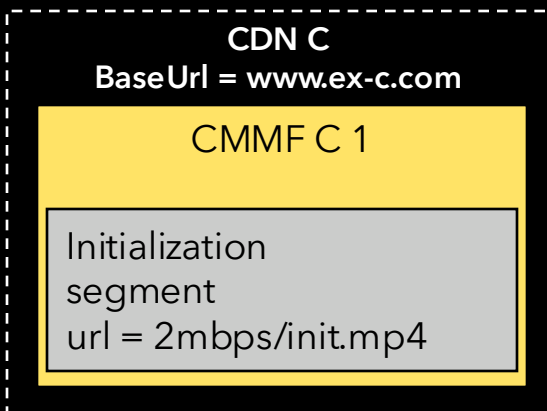
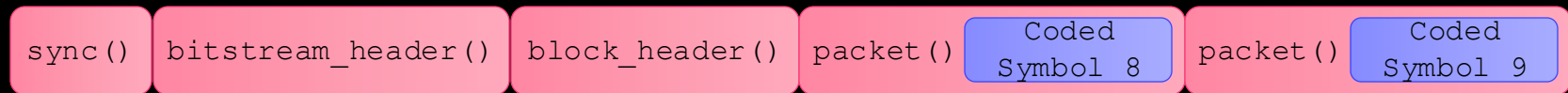
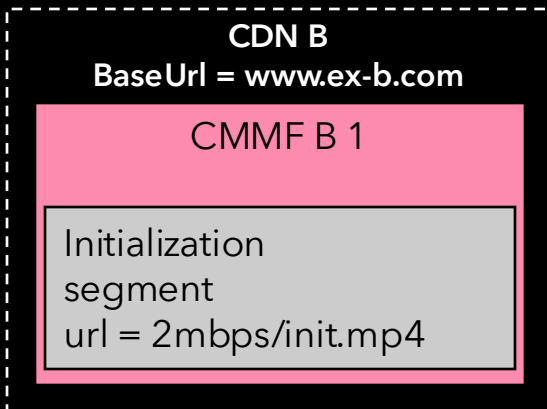
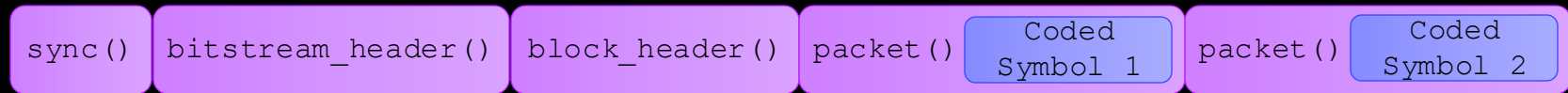
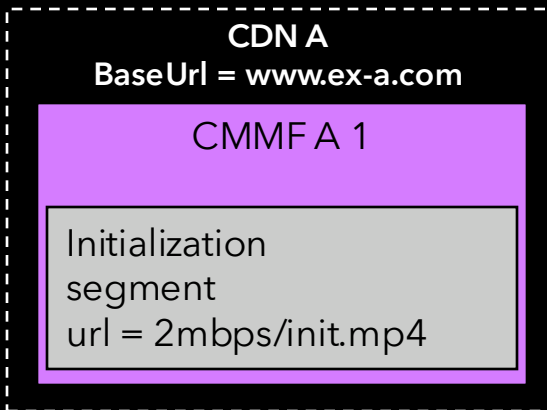
CMMF B 1

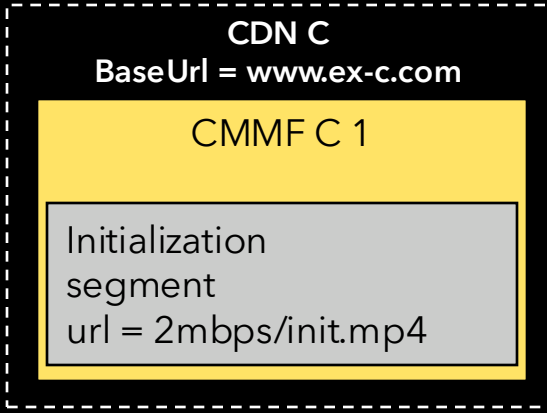
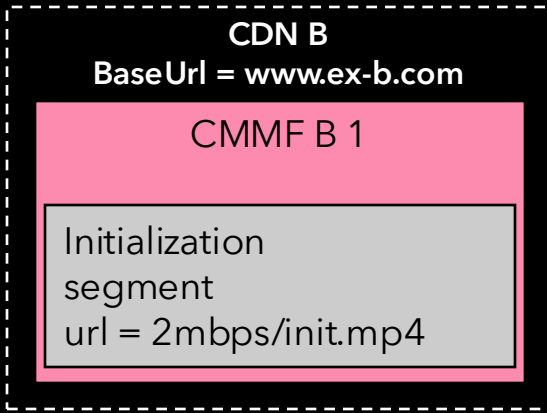
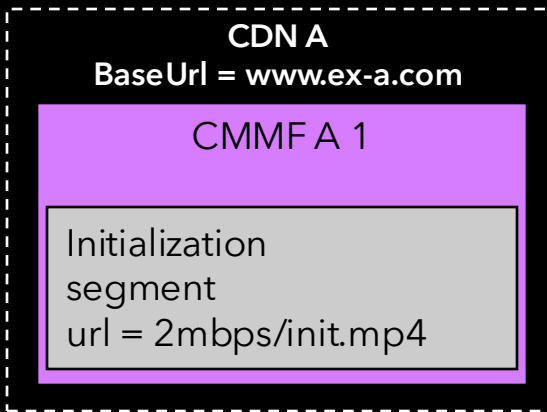
Initialization
segment
url = 2mbps/init.mp4

CMMF C 1

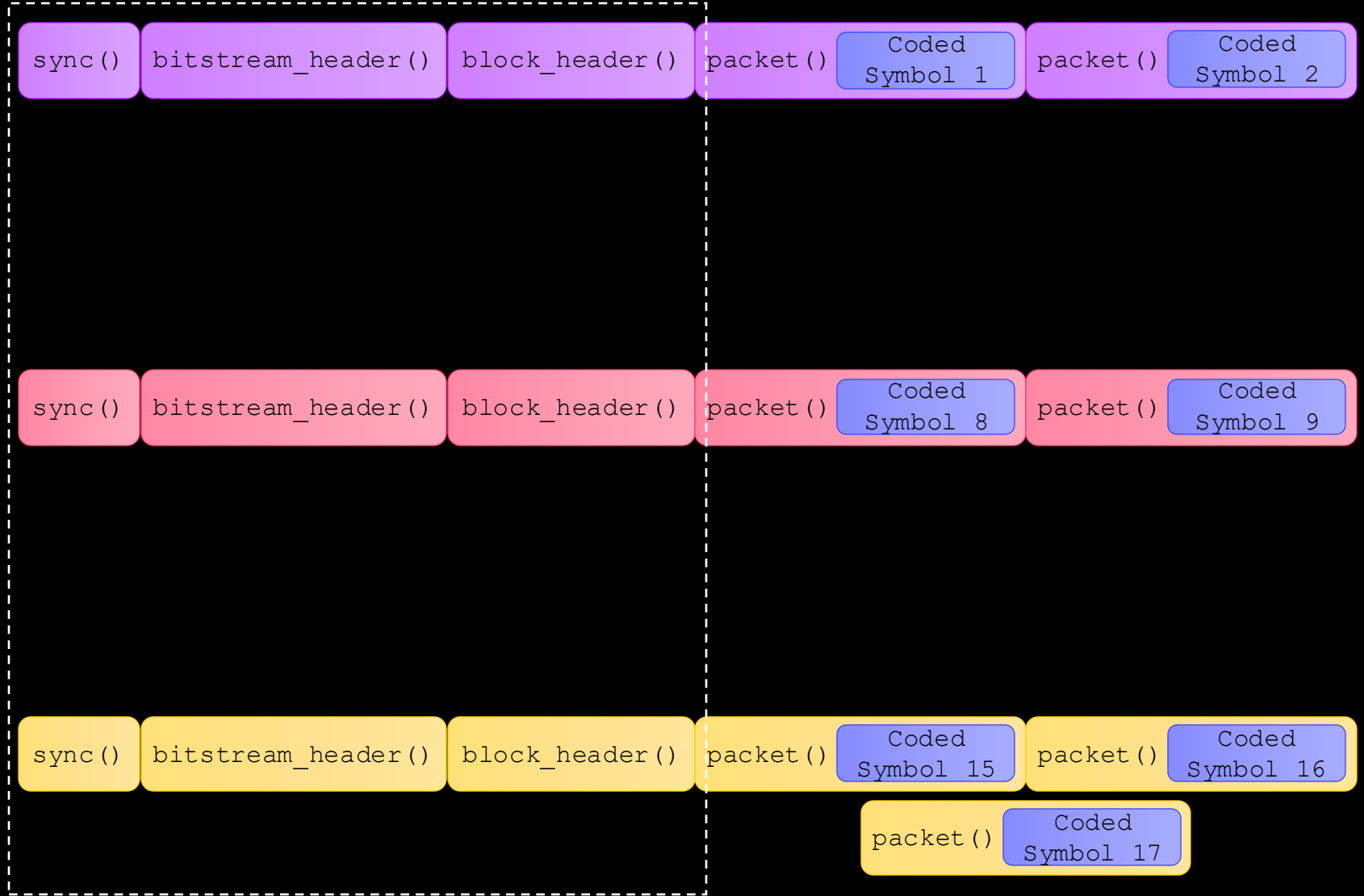
Initialization
segment
url = 2mbps/init.mp4

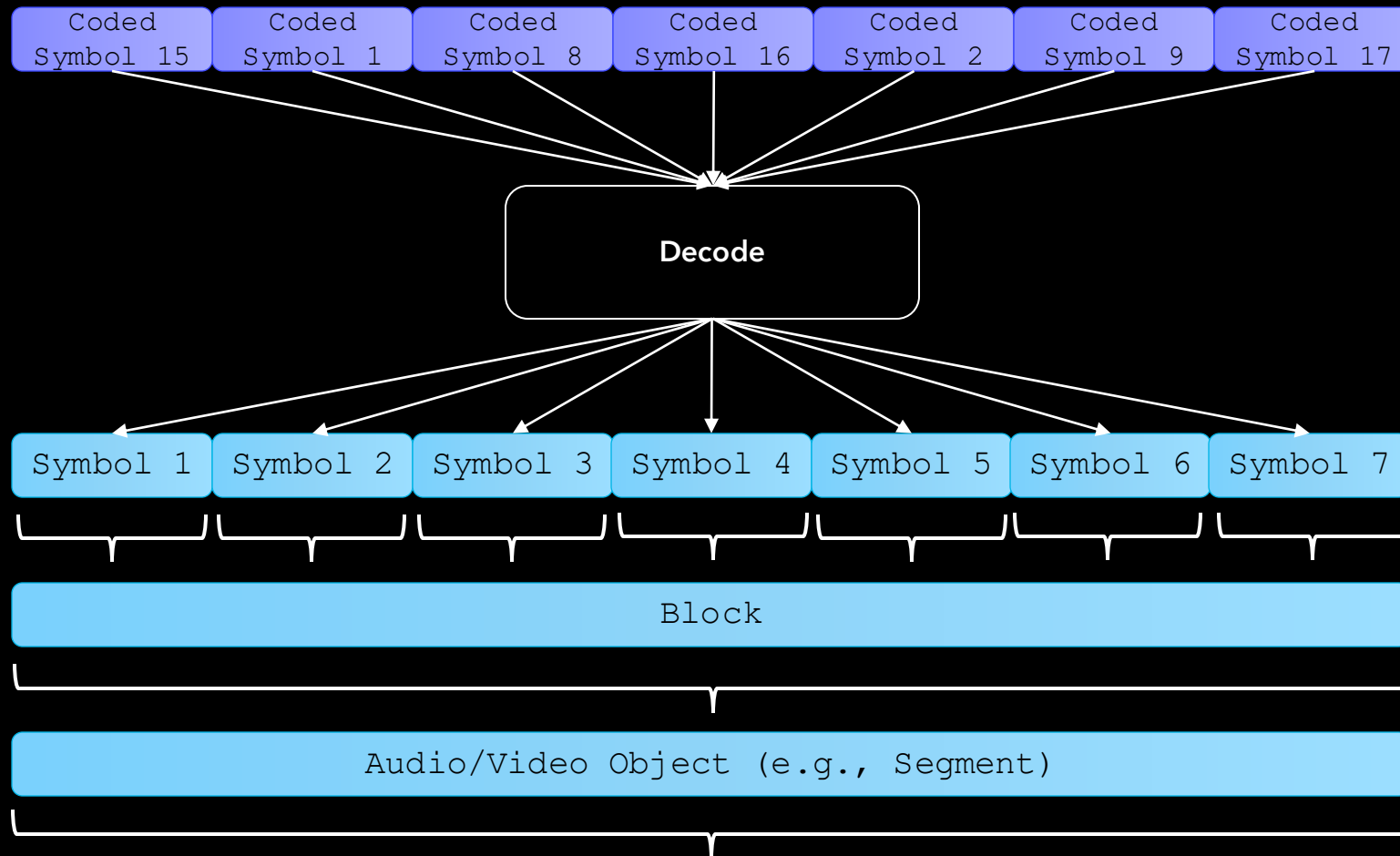






Validate each CMMF bitstream is unique and can be jointly decoded





Initialization segment
url = 2mbps/init.mp4





CMMF AS A CONTENT DELIVERY PROTOCOL

THOMAS STOCKHAMMER (QUALCOMM)

3GPP Advanced Media Delivery

Multi-CDN and Multi-Access Media Delivery

- Focus on integrations within the 5G Media Streaming System of technologies (including CMMF) that support media delivery from multiple locations (either internal or external to the 5G network)
- Recommendations developed for architectural, call flow, and procedural updates to 3GPP Release 19 specifications
- Study completed in November 2024



THANK YOU

CMMF as an IETF CDP Instantiation

Thomas Stockhammer
Qualcomm Incorporated

Qualcomm Technologies in Context



- **FEC Codes**

- Raptor
- RaptorQ

- **Frameworks**

- FLUTE
- ROUTE
- FECFRAME
- Video Delivery Network

- **Service Layers**

- MBMS Service Layer
- Group Communication
- 5G Broadcast
- ATSC Service Layer
- DVB-I Service Layer
- DASH

- These technologies are developments based on long and substantial research
- These technologies have undergone a significant diligence by being standardized at the appropriate organizations
 - IETF, 3GPP, ETSI, MPEG, DVB, ATSC, DASH-IF
- These technologies are already implemented
- These technologies are at least partially available on QCT products and on commercial devices
- Open source & proprietary implementations exists
- Interop was done with different partners

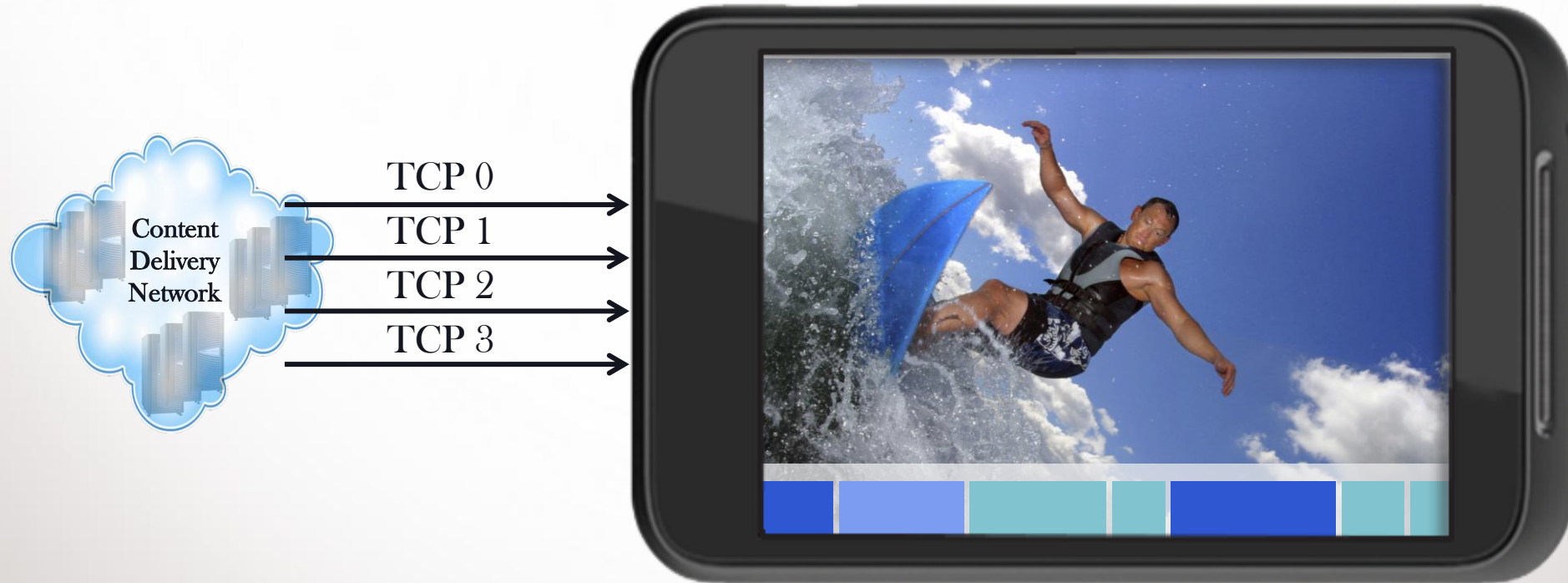
DFSplash

<https://screenplaysmag.com/blog/digital-fountain-challenges-hd-web-video-delivery-cost-structures/>

- [September 2007](#)
- [DF Splash from Digital Fountain](#): I've been following these guys for years having met them at a Multicasting conference years ago. Digital Fountain's technology breaks up a file into various bits and somehow is able to reassemble them on the receiving end in way that helps eliminate latency issues. The company is putting the technology to use in its new DF Splash content delivery network, slated to launch in January 2008. What, a new CDN? Based on the demonstration, this is a CDN that could make some waves, delivering high-res video over a standard Internet connection very efficiently. And, they're sitting on Amazon's Web Services Network, lowering their need for servers and other capital equipment things. Digital Fountain is differentiating itself from Limelight and Akamai by claiming they need far less server deployments to reach the edge. For instance, the demonstration was pulling from 12 servers, though that number can vary. DF Splash will start by serving up the QuickTime H.264 format because it does high-res compression very well and it can extend down to the mobile and set-top box platforms. Viewers just need a browser plug-in to watch the videos. Given that Adobe is an investor and the fact that Adobe is going to be supporting H.264 shortly, my guess is this will be available for Flash sooner than later.
- <https://www.digitalbroadcasting.com/doc/nonstop-riot-delivers-the-hottest-new-music-0001>

2014: Snapdragon Transport Accelerator™

Video delivery over multiple HTTP/TCP connections



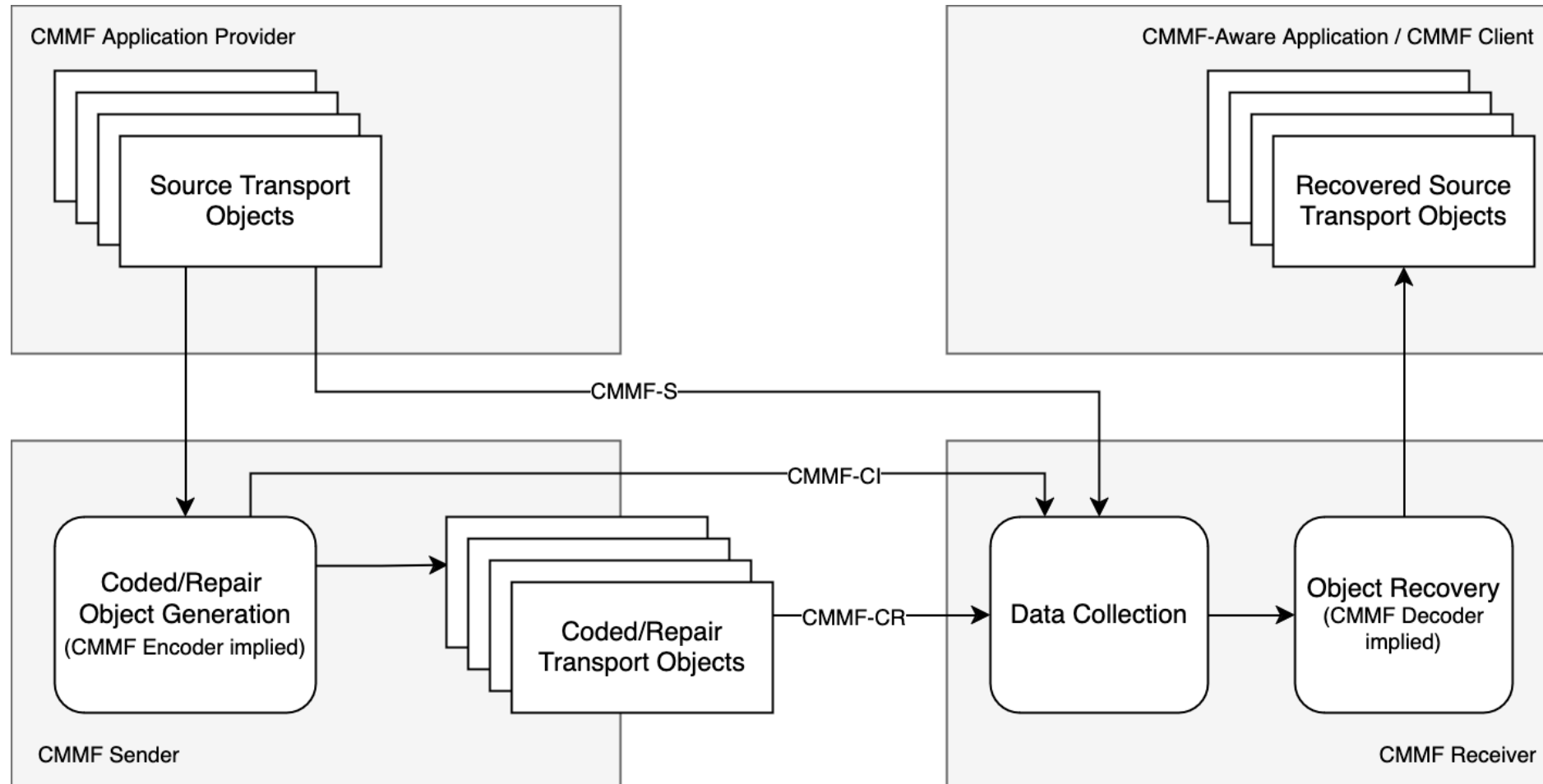
Improved and consistent quality & reduced stalling

Background

- [ETSI TS 103 973](#) defines a Content Delivery based Instantation in Annex D
- CMMF is considered as a Content Delivery Protocol (CDP) as defined in clause of 8 of [RFC5052].
- Annex D provides a mapping of CMMF to the RFC 5052 principles and vice versa
- This instantiation also permits to re-use existing FEC Code including Raptor (the 3GPP MBMS code) as defined in RFC 5053 and RaptorQ as defined in RFC 6330.
- The protocol also makes use of 3GPP familiar concepts such as the File Delivery Table in FLUTE and used in MBMS and MBS

CMMF Architecture

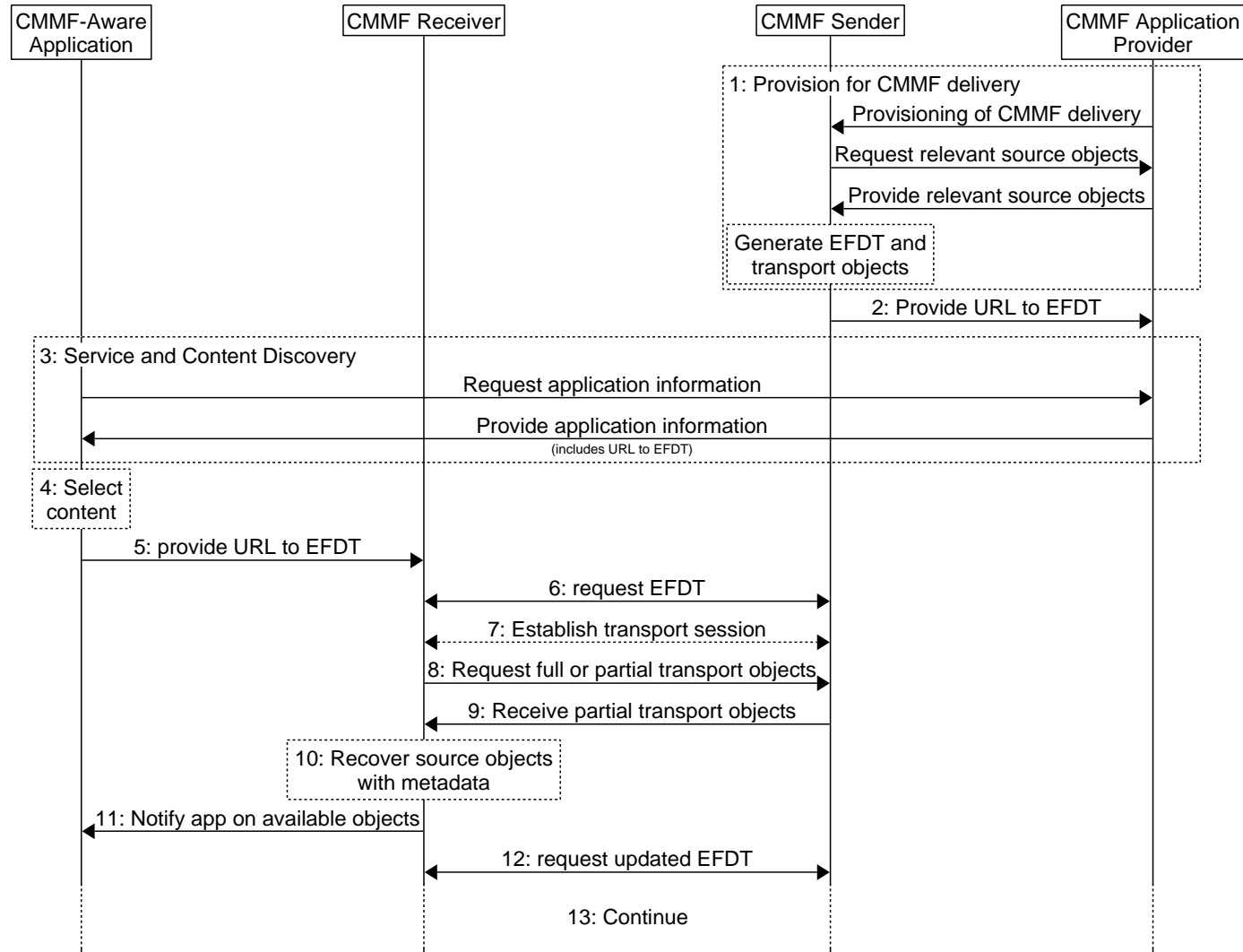
Simplified Architecture as developed by Qualcomm during CMMF specification development



Defined Reference Points

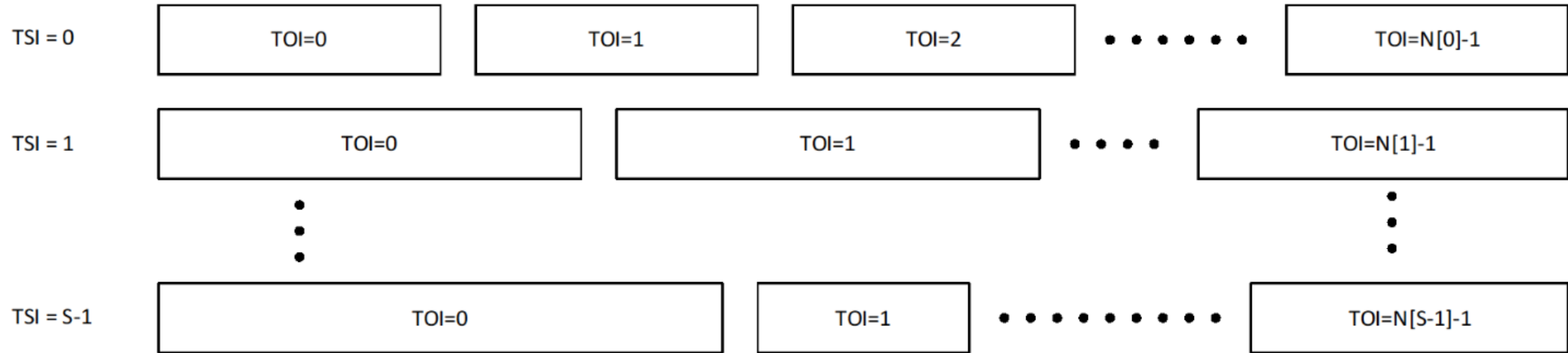
- CMMF-CI: This reference point provides Configuration Information describing the location and relationship of the source and coded/repair objects, may be provided to a CMMF receiver.
- CMMF-S: This reference point provides the source transport objects. For CMMF, these objects are unmodified from original data. Parts of these objects may be used by Object Recovery to recover source objects.
- CMMF-CR: This reference point provides the coded/repair transport objects. Details on the formats on this reference point are provided in clause.
- Not explicitly mentioned in the diagram are the following reference points:
 - The server-side configuration of the CMMF sender
 - The client-side API to have an API between the CMMF receiver and the application
- Note also that the application is completely independent of any manifest and so on

Call flow for FLUTE-based CMMF CDP instantiation (Figure D.1)



CMMF Transport Objects and Transport Sessions

Source and Repair objects are assigned to TSI and TOI



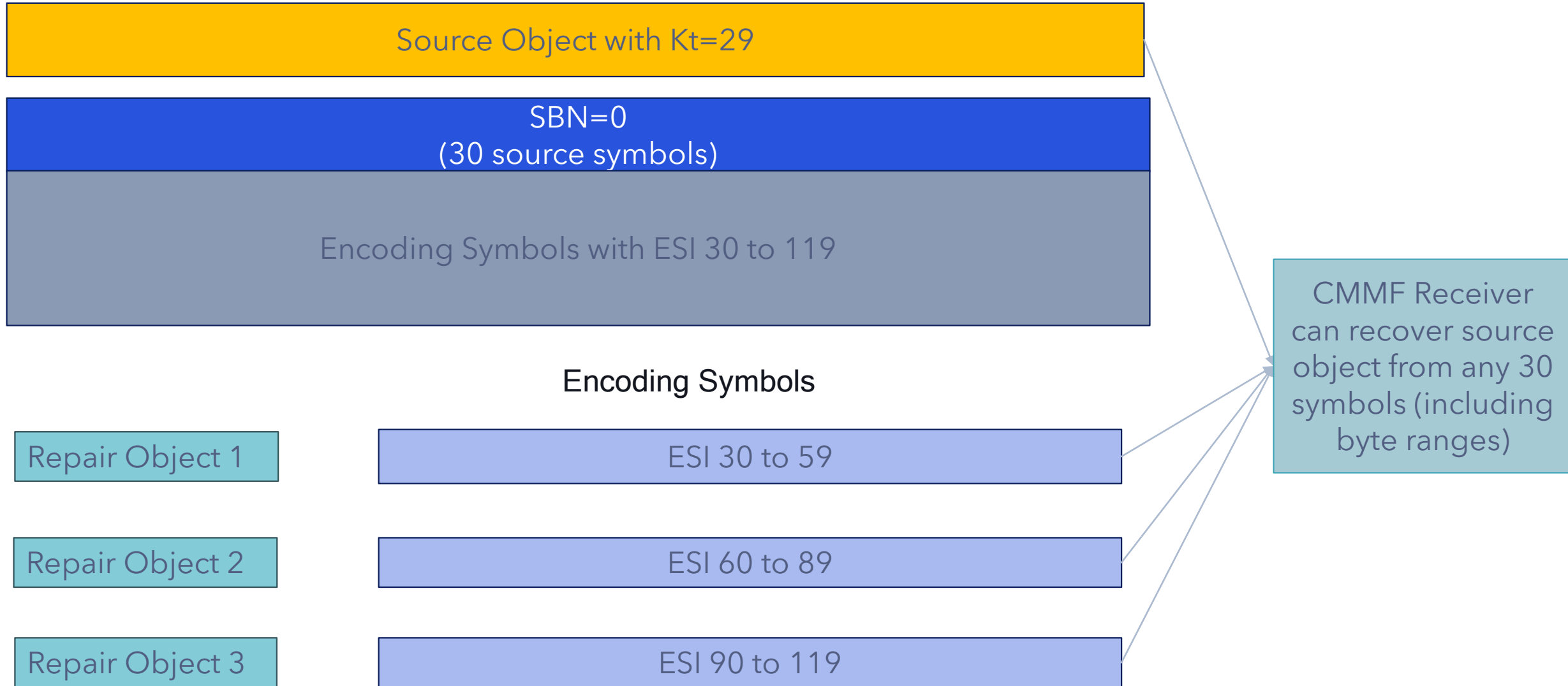
- Information is provided to receiver through configuration information, describing
 - Each logical flow that is operated
 - Each object, metadata of the object and other information
 - The type of the object as well encoding parameters

Configuration Parameters

- The Configuration Information allows the CMMF client to map an application request to CMMF receiver operations.
- Table D.1.5-1 provides examples of possible Configuration Information that can be used where source and coded/repair transport objects are described.
- Some use cases may require additional information or only a subset of this information, and a simpler version of this parameter set may be used.
- Updates of the CI information may happen

Parameter	Usage	Definition
Complete	OD	Indicates whether the Configuration Information is complete.
Location	O	Provides information where the Configuration Information can be accessed in carried externally.
Expires	M	Provides information when this Configuration Information is no longer valid and an update is needed, for example using a reload from Location.
Source Flow	1 ... S	Provides 1 ... S source flows.
TSI	M	Identifier of the source flow.
Object	1 ... N	Provides 1 ... N objects in the source flow.
TOI	M	Transport object identifier (TOI) value that represents the source object.
Size	M	Size of the transmission object in bytes.
Content-Type		Describes media type of file.
Encoding		Describes encoding of file .
Message Digest		Message digest of file.
Associated URI		Name, Identification, and Location of file (specified by the URI).
Access URL		The URL where the source object can be accessed. If the field is not present, then the source flow is not directly accessible.
availabilityStartTime		Provides a wall-clock time when the resource is accessible.
availabilityEndTime		Provides a wall-clock time when the resource ceases to be available.
<Additional metadata>		May include cache or E-Tag metadata.
Representation		Refers to a DASH Representation in an MPD or a Track in an HLS manifest.
Coded/Repair Flow	1 ... R	Provides 1 ... R coded/repair flows.
TSI	M	Identifier of the coded/repair flow.
Object	1 ... N	Provides 1 ... N objects in the coded/repair flow.
TOI	M	Transport object identifier (TOI) value that represents the coded/repair object.
FEC-OTI		If object is coded using a scheme based on [RFC5052], FEC Object transmission information including the FEC Encoding ID and, if relevant, the FEC Instance ID.
includedSourceTOI	M	List of (TSI, TOI pairs) of the included source transport objects forming super objects . Typically, only a single pair is provided.
Content-Type		Media Mime Type of the file.
completeObject	OD FALSE	Indicates whether the transport object includes sufficient information to recover all files included in this coded/repair object.
symbolArrangement		Provide this symbol arrangement in the object according to Table 78. If not present, the symbol Arrangement is unknown and only present in the bitstream.
sAParameters		may be present if the symbolArrangement is present. If present, it provides the parameters assigned to the symbol arrangement as defined in Table 36. For arrangement 2 and 3, this is a comma-separated list of: Index difference, Symbol group, Index in symbol group
Access URLs		The URLs where the coded/repair object can be accessed.
availabilityStartTime		Provides a wall-clock time, when the resource is accessible.
availabilityEndTime		Provides a wall-clock time, when the resource ceases to be available.
<Additional metadata >		

Encoding Options in Annex D



Source Object with Kt=29 symbols

block=0 block_num_symbols = 10	block=1 block_num_symbols = 10	block=2 block_num_symbols = 9
Encoding Symbols with ESI 0 to 29	Encoding Symbols with ESI 0 to 29	Encoding Symbols with ESI 0 to 26

mbpg_difference=3

mbpg_first_symbol_index=0

0 0 0 3 3 3

.....

2 2 2 2 2
4 4 4 7 7

mbpg_num_symbols=25

mbpg_first_symbol_index=1

1 1 1 4 4 4

.....

2 2 2 2 2
5 5 5 8 8

mbpg_num_symbols=25

mbpg_first_symbol_index=2

2 2 2 5 5 5

.....

2 2 2 2 2
6 6 6 9 9

mbpg_num_symbols=25

mbpg_symbol_group_subset_index = 4

Encoded Symbols

Source Object with Kt=29 symbols

block=0 (ESI 0-9) block_num_symbols = 10	block=1 (ESI 0-9) block_num_symbols = 10	block=2 (ESI 0-8) block_num_symbols = 9
FEC with ESI 10 to 30	FEC with ESI 10 to 30	FEC with ESI 9 to 30

mbpg_difference=3

mbpg_first_symbol_index=0

mbpg_first_symbol_index=1

mbpg_first_symbol_index=2

Source Symbols

0 3 6 9 2 5 8 1 4 7

1 4 7 0 3 6 9 2 5 8

2 5 8 1 4 7 0 3 6

mbpg_symbol_group_subset_index=0

1 1 9
0 0

1 1 1
1 1 0

1 1 1
2 2 1

Repair Symbols

.....
2 2 2 2 2 2
5 5 4 8 8 7

.....
2 2 2 2 2 2
6 6 5 9 9 8

.....
2 2 2 3 3 2
7 7 6 0 0 9

mbpg_num_symbols=30

mbpg_num_symbols=30

mbpg_num_symbols=29

One Implementation of CI - Extended FDT

Aligned with File Delivery Table from FLUTE with extensions

Multiple Files -

Self-contained objects including source symbols

Single File - Source and Partial Encoding Object

```
<?xml version="1.0" encoding="UTF-8"?>
<FDTInstance xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:ETSI:CMMF:2023:FDT"
  xsi:schemaLocation="urn:ETSI:CMMF:2023:FDT extendedFDT.xsd"
  Expires="2024-05-30T09:30:10Z"
  Complete="true"
  ContentType="video/mp4 codecs='avc1.42c01e,mp4a.40.29' profiles='iso8'"
  FEC-Encoding-ID="6"
  FEC-Encoding-Symbol-Length="64">
  <File ContentLocation="https://example.com/efd1.mp4"
    TOI="0"
    Content-Length="64000">
    <EncodedObjects type="source"
      independentObject="true">https://example.com/efd1.mp4</EncodedObjects>
    <EncodedObjects type="partial"
      interleavingType="sequential-sbn"
      includedSymbols="500,0,1001">https://example.com/part1.cmf</EncodedObjects>
    <EncodedObjects type="partial"
      interleavingType="sequential-sbn"
      includedSymbols="500,0,1501">https://example.com/part2.cmf</EncodedObjects>
    <EncodedObjects type="partial"
      interleavingType="sequential-sbn"
      includedSymbols="500,0,2001">https://example.com/part3.cmf</EncodedObjects>
  </File>
</FDTInstance>
```

```
<FDTInstance xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns="urn:ETSI:CMMF:2023:FDT"
  xsi:schemaLocation="urn:ETSI:CMMF:2023:FDT extendedFDT.xsd"
  Expires="2024-05-30T09:30:10Z"
  Complete="true"
  FEC-OTI-FEC-Encoding-ID="6"
  FEC-OTI-Encoding-Symbol-Length="64">
  <File ContentLocation="https://example.com/efd1-video.mp4"
    ContentType="video/mp4 codecs='avc1.42c01e' profiles='iso8'"
    TOI="0"
    Content-Length="64000">
    <EncodedObjects type="self-contained"
      interleavingType="spread"
      independentObject="true"
      includedSymbols="1001,3,1,0,0,0">https://example.com/part1-video.cmf</EncodedObjects>
    <EncodedObjects type="self-contained"
      interleavingType="spread"
      independentObject="true"
      includedSymbols="1001,3,1,0,0,1">https://example.com/part2-video.cmf</EncodedObjects>
    <EncodedObjects type="self-contained"
      interleavingType="spread"
      independentObject="true"
      includedSymbols="1001,3,1,0,0,2">https://example.com/part3-video.cmf</EncodedObjects>
  </File>
  <File ContentLocation="https://example.com/efd1-audio.mp4"
    ContentType="audio/mp4 codecs='mp4a.40.29' profiles='iso8'"
    TOI="1"
    Content-Length="4800">
    <EncodedObjects type="self-contained"
      interleavingType="spread"
      independentObject="true"
      includedSymbols="80,3,1,0,0,0">https://example.com/part1-audio.cmf</EncodedObjects>
    <EncodedObjects type="self-contained"
      interleavingType="spread"
      independentObject="true"
      includedSymbols="80,3,1,0,0,1">https://example.com/part2-audio.cmf</EncodedObjects>
    <EncodedObjects type="self-contained"
      interleavingType="spread"
      independentObject="true"
      includedSymbols="80,3,1,0,0,2">https://example.com/part3-audio.cmf</EncodedObjects>
  </File>
</FDTInstance>
```

5G Media Streaming and CMMF

Rel-19 Study on Advanced Media Delivery

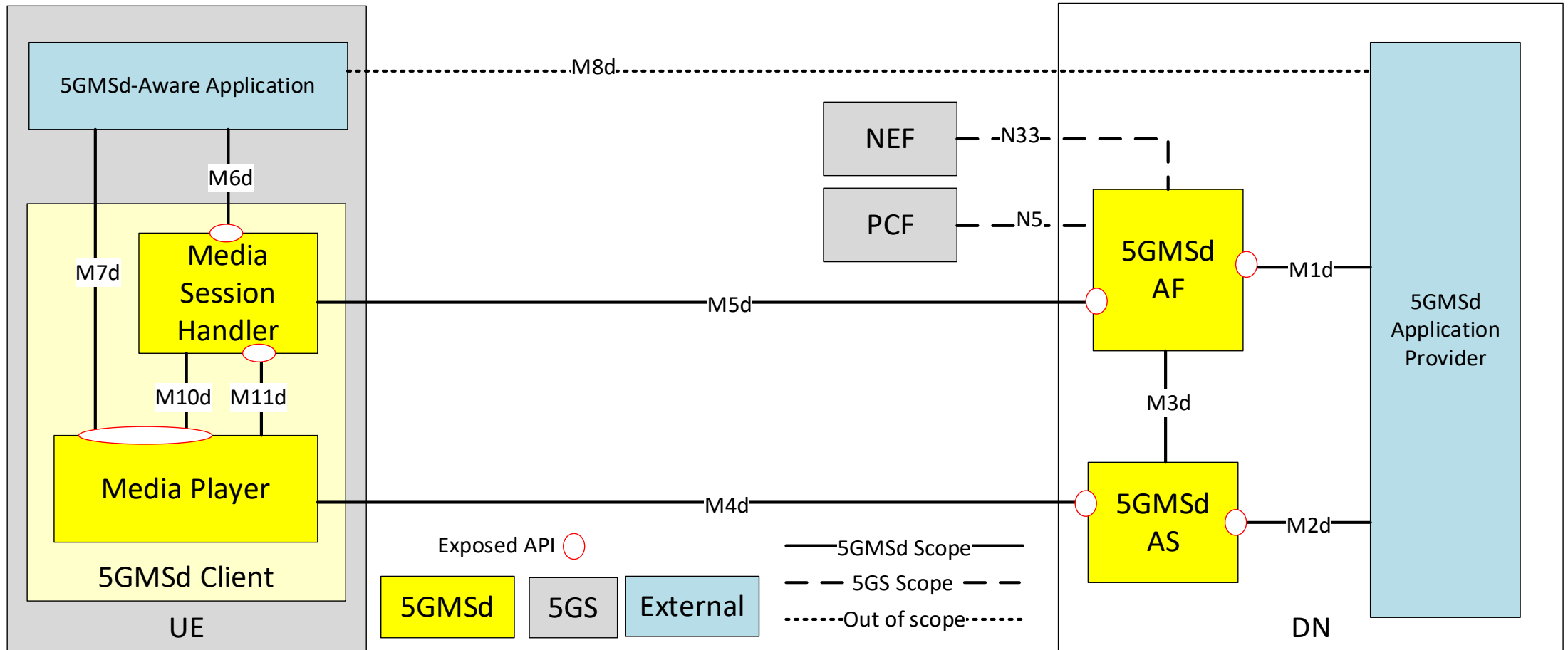
Major work completed and approved just yesterday

- In the extensions to 5G Media Streaming, the focus was on integration of the latest of the greatest streaming management functionalities such as Common Media Client Data (CMCD), Content Steering, Coded Multisource Media Format (CMMF), Low Latency, Low Loss, and Scalable Throughput (L4S), Content Protection Information eXchange (CPIX) and many other technologies.
- Based on the initial study, a stage-2 work item was agreed in ([S4-242270](#), QC Rapporteur, 21 co-signers) and stage-3 should follow in March.
- CMMF integration is an ongoing study item, it requires stage-3 details



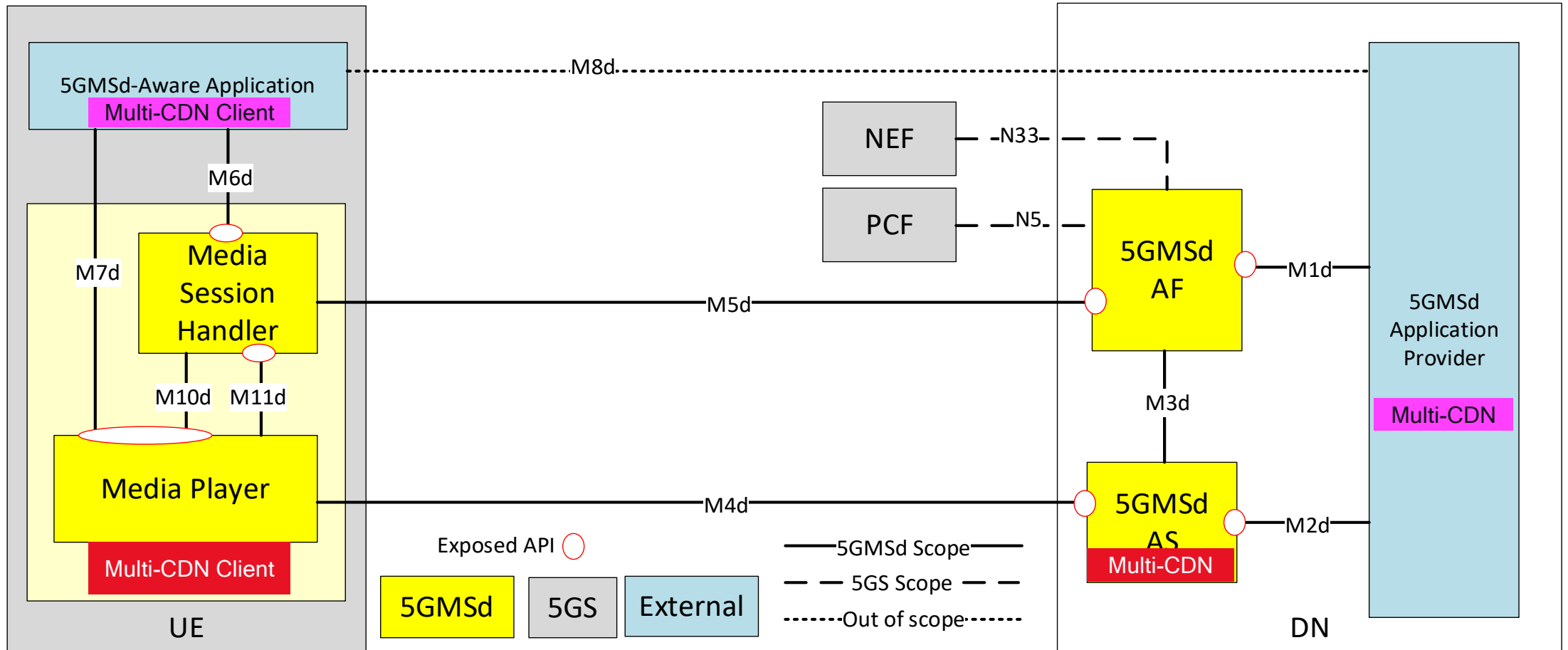
5G Media Streaming architecture

As defined in TS 26.501

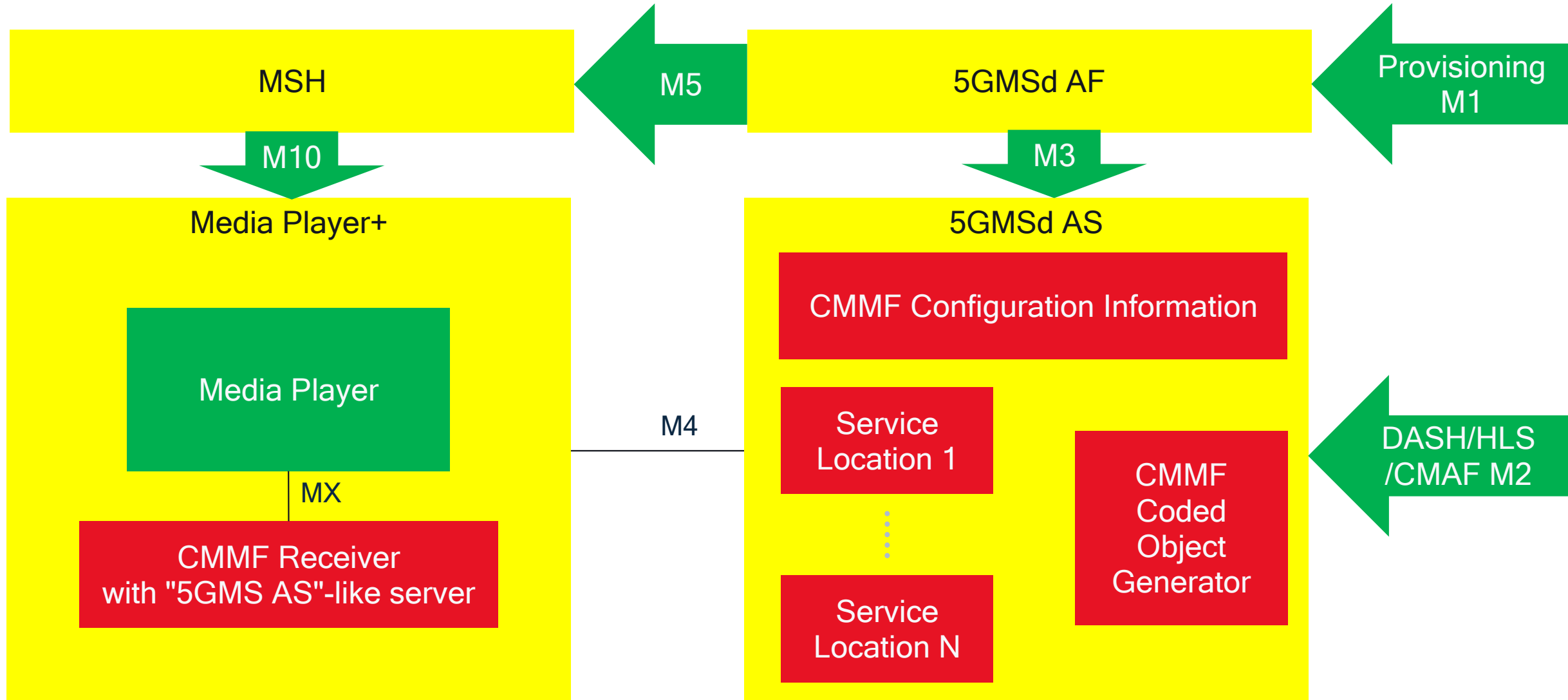


Multi-CDN is either ...

... A feature of the **5GMS AS** or it is a feature of the **Application Provider**



Architecture Extensions for CMMF



Thank you

Qualcomm

Follow us on: [in](#) [twitter](#) [instagram](#) [youtube](#) [facebook](#)

For more information, visit us at:

qualcomm.com & qualcomm.com/blog

Nothing in these materials is an offer to sell any of the components or devices referenced herein.

©2018-2023 Qualcomm Technologies, Inc. and/or its affiliated companies. All Rights Reserved.

Qualcomm is a trademark or registered trademark of Qualcomm Incorporated. Other products and brand names may be trademarks or registered trademarks of their respective owners.

References in this presentation to "Qualcomm" may mean Qualcomm Incorporated, Qualcomm Technologies, Inc., and/or other subsidiaries or business units within the Qualcomm corporate structure, as applicable. Qualcomm Incorporated includes our licensing business, QTL, and the vast majority of our patent portfolio. Qualcomm Technologies, Inc., a subsidiary of Qualcomm Incorporated, operates, along with its subsidiaries, substantially all of our engineering, research and development functions, and substantially all of our products and services businesses, including our QCT semiconductor business.

Snapdragon and Qualcomm branded products are products of Qualcomm Technologies, Inc. and/or its subsidiaries. Qualcomm patented technologies are licensed by Qualcomm Incorporated.