SEGMENTS:2024

DELIVERY

ARCHITECTURES FOR MULTI-CDN SWITCHING

Yuriy Reznik
VP, Research
Brightcove, Inc.

OUTLINE

SEGMENTS:2024

1. Overview of activity in SVTA P&P SG

2. Technologies under consideration / study

   - Client-side switching
   - Server-side content steering
   - Edge-supported steering framework

3. Demo & Preliminary results

4. Conclusions

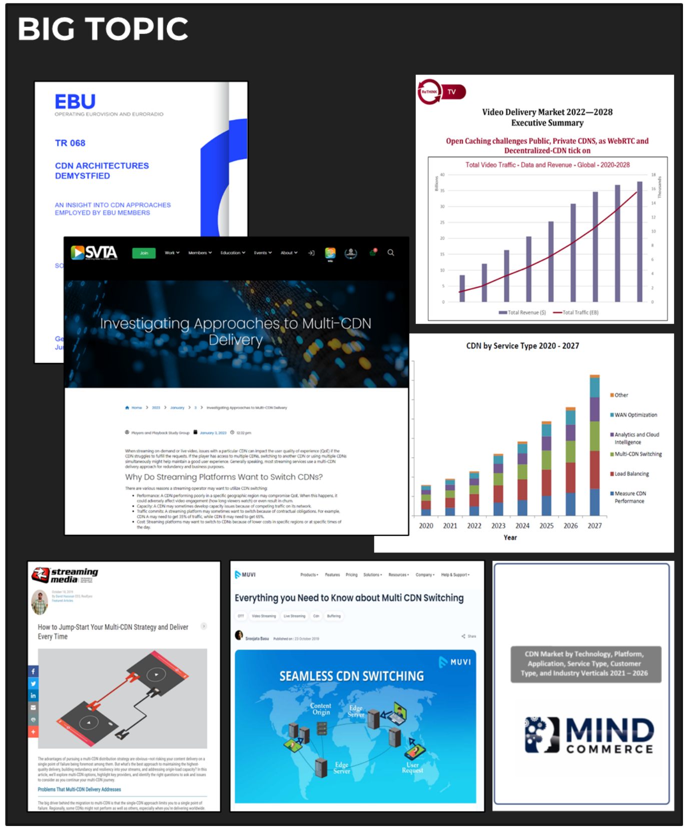
MULTI-CDN DELIVERY

SEGMENTS:2024

Many architectures have been proposed in the past

Architecture	Pros	Cons
DNS-based	This is the simplest of all solutions since the source video URL always remains constant.	Switch delay is more time-consuming, ranging from 300 seconds to even five minutes in case of CDN failures. This can immensely hamper the user QoE.
On-the-fly manifest rewrite	Better user experience due to midstream switching eliminating the need for hard refresh during video playback. No matter the volume of session resets, this method reduces the chances of a cascade effect that may hamper the video workflow.	Rewriting the manifest can sometimes bring about errors. Midstream switching is not completely seamless, and takes time for the server to understand that a particular CDN is unavailable.
Server-side	It is a relatively simple CDN switching method to implement since changes happen in the server itself that is easier for the operator to control.	Page loading may take some time, adding to delays. Since CDN switching is based on the collective data from many clients, it does necessarily consider the unique conditions of the actual clients.
Client-side	QoS data is almost accurate as it is fetched based on individual clients' local and real-time performance metrics. Seamless midstream CDN switching is possible.	It is a complex procedure to implement when built in-house due to the code complexity of the algorithms that requires detailed planning.

https://www.svta.org/2023/01/03/investigating-approaches-to-multi-cdn-delivery/



But none of them is perfect…

- The objective of SVTA activity is to study and produce report guiding the industry on the available choices

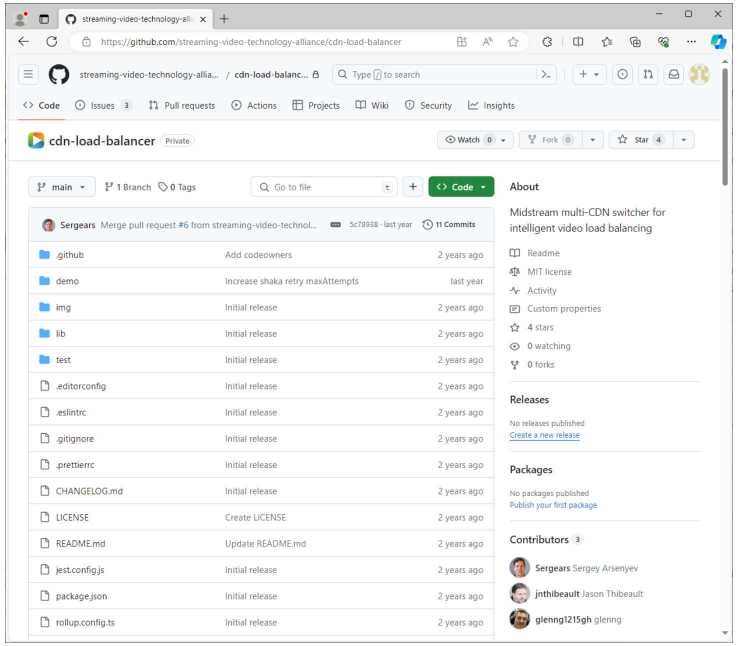
ARCHITECTURES UNDER STUDY

SEGMENTS:2024

SVTA member contributions:

Client-side solution:

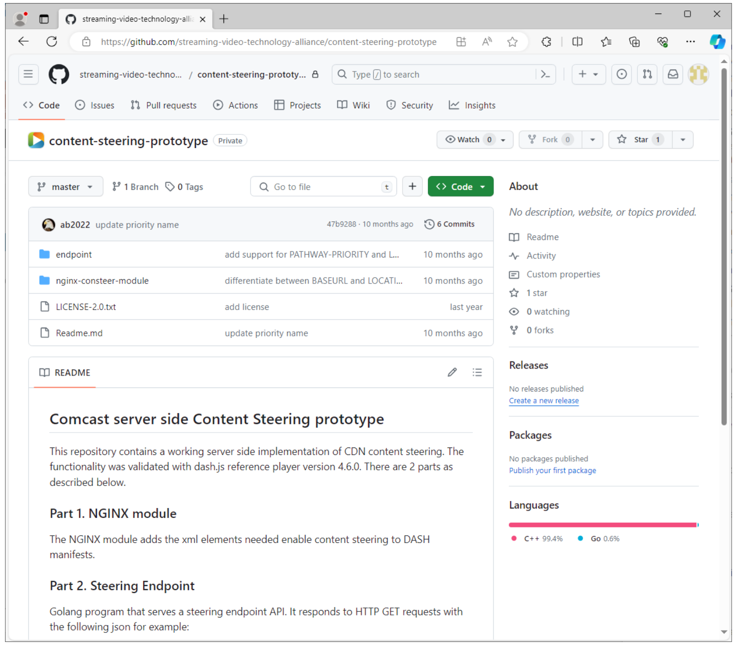
CDN Load Balancer (Lumen)



https://github.com/streaming-video-technology-alliance/cdn-load-balancer

Server-side solution:

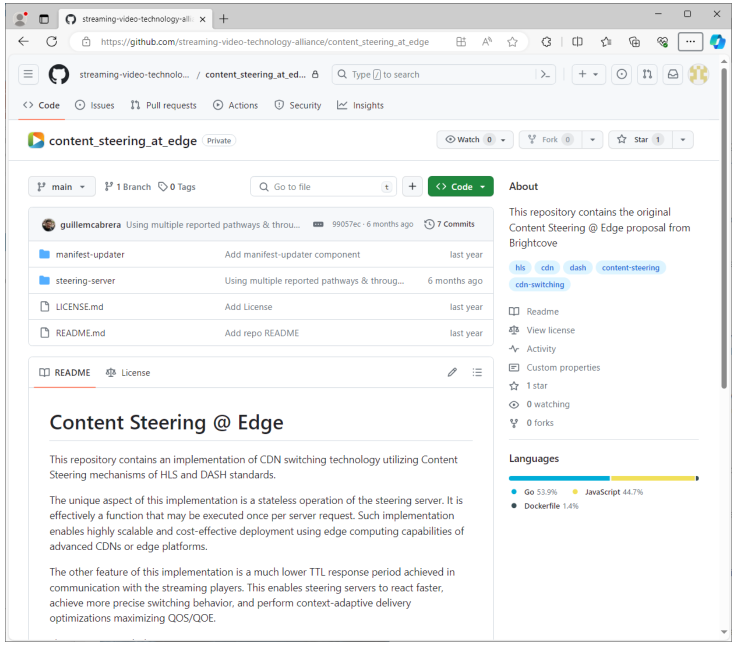
Content steering prototype (Comcast)



https://github.com/streaming-video-technology-alliance/content-steering-prototype

Edge-side solution:

Content steering @ edge (Brightcove)



https://github.com/streaming-video-technology-alliance/content_steering_at_edge

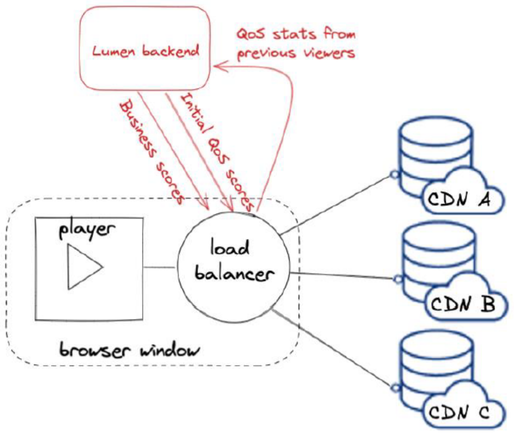
CLIENT-SIDE SWITCHING

SEGMENTS:2024

CDN load balancer by Lumen

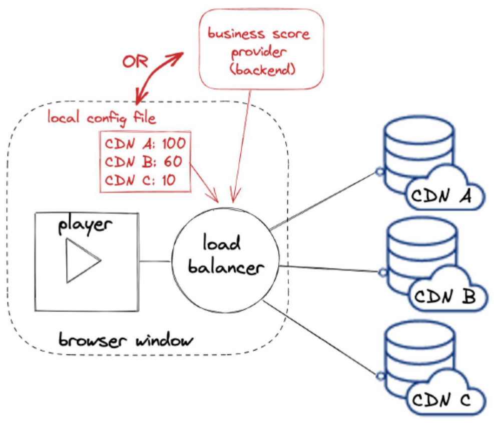
$$\text{CDN global score} = \text{CDN QoS score} \times \text{CDN business score}$$

Commercial version



Initial CDN is chosen based on supplied business scores and QoS scores from previous viewers

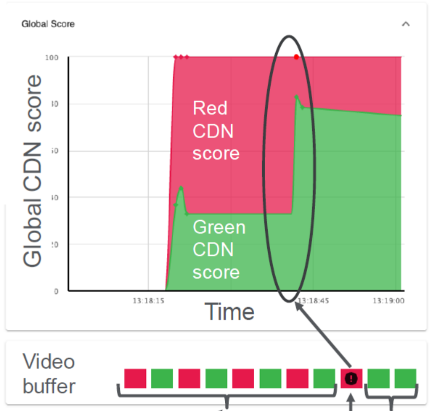
Open-source version



Initial CDN is chosen based on supplied business scores and default QoS scores

Broadcaster can prioritize certain CDNs over the others because of

- Lower cost
- Geographical proximity
- Contractual obligations
- etc



Segments successfully downloaded from red CDN and green CDN

Error happened when requesting from red CDN

Now prefer green CDN

LUMEN

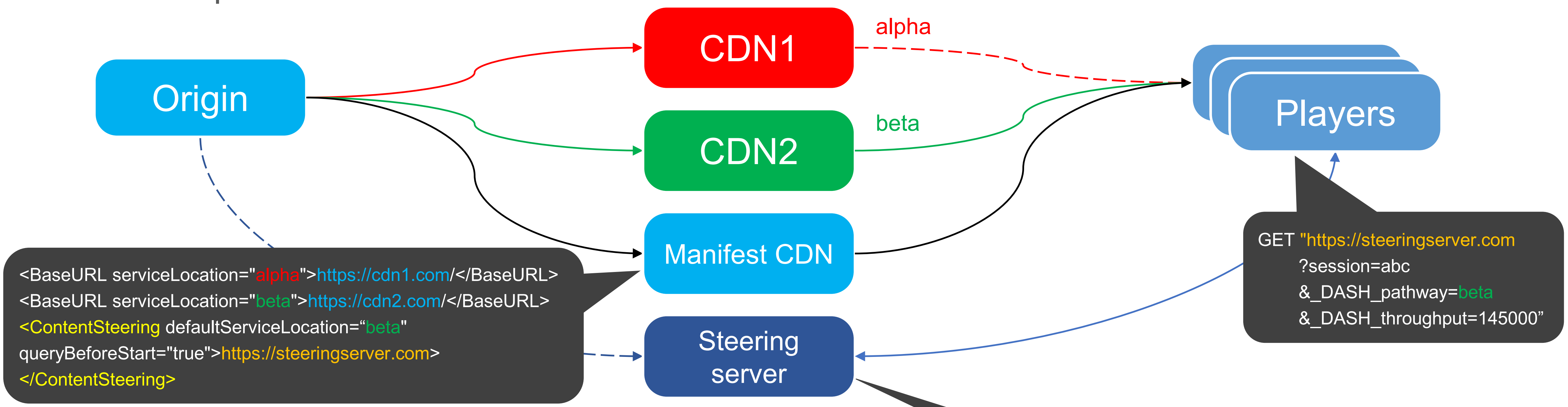
SVTA open source version works with:

- Shaka player (HLS, DASH), and DASH.js (DASH)
- Requires back-end for providing initial QoS and business scores.

HLS / DASH CONTENT STEERING

SEGMENTS:2024

The concept



Pros:

- Standards based
- The same steering protocol & server is used for both HLS and DASH
- Simple integration – no need to patch players!
- Complements the existing BaseURL redundancy / failover behavior mechanisms

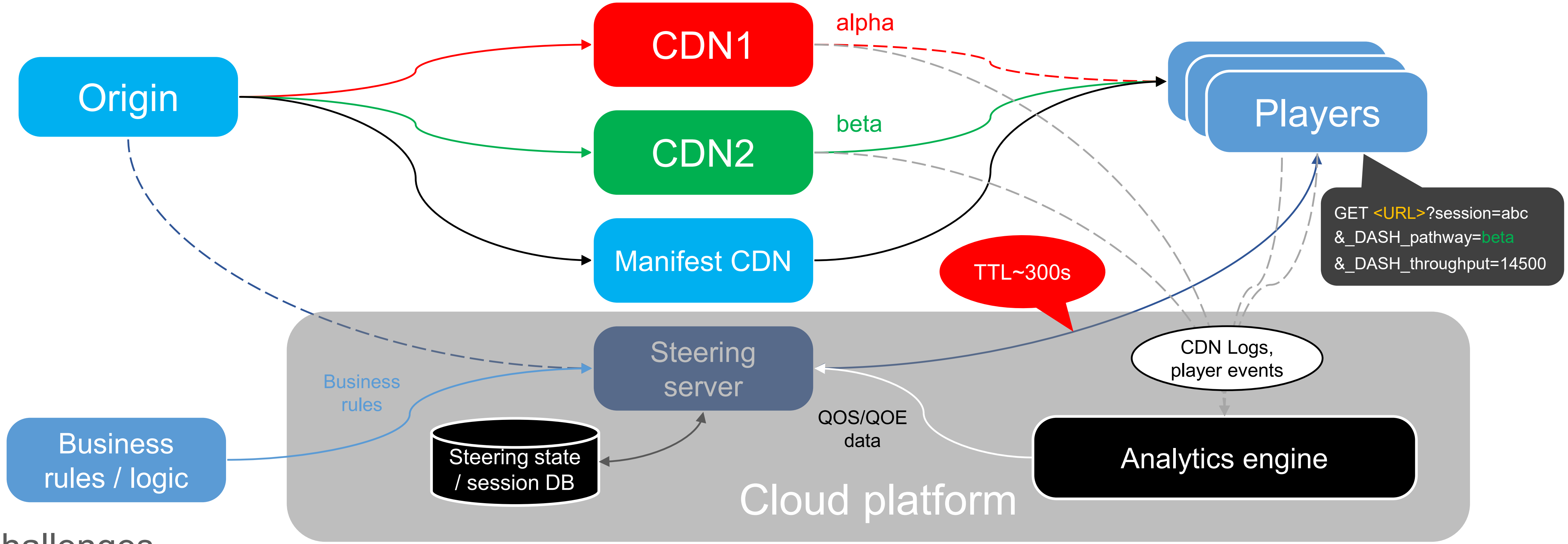
CONTENT STEERING: CLIENTS

Content-steering-capable clients:

Media Player	DASH Content Steering	HLS Content Steering
dash.js	since version 4.5.0	
VIDEO.JS	since version 8.8.0	since version 8.8.0
Shaka Player	since version 4.6.0	since version 4.6.0
ExoPlayer	planned for 2024	planned for 2024
hls.js		since version 1.4.0
AVPlayer		since iOS version 15.0

CONTENT STEERING SERVER

Server-side implementation



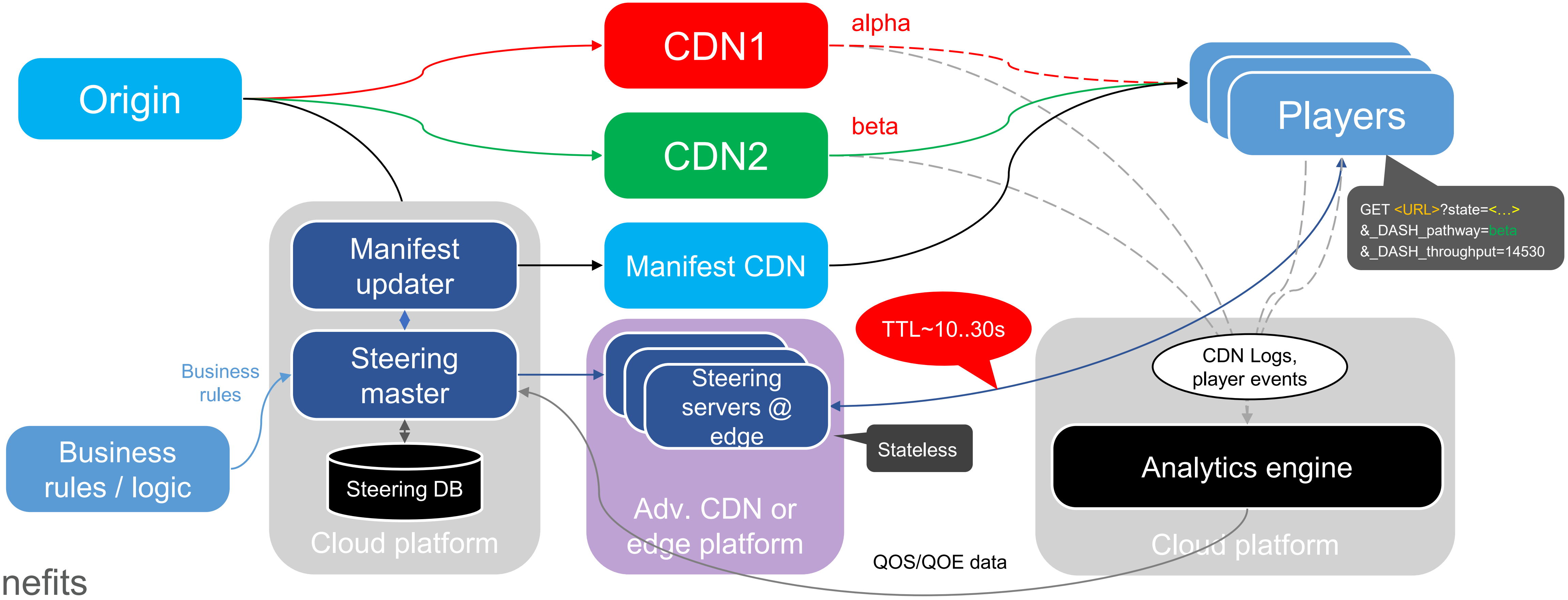
Challenges

- TTL time: 300s default is too long! Suitable for basic CDN load balancing. Not suitable for QOE optimizations!
- Scalability: the steering server should be at least as scalable as manifest CDN!
- Costs: reducing TTL will increase number of requests and traffic to the steering server!

CONTENT STEERING @ EDGE

SEGMENTS:2024

Edge-based architecture



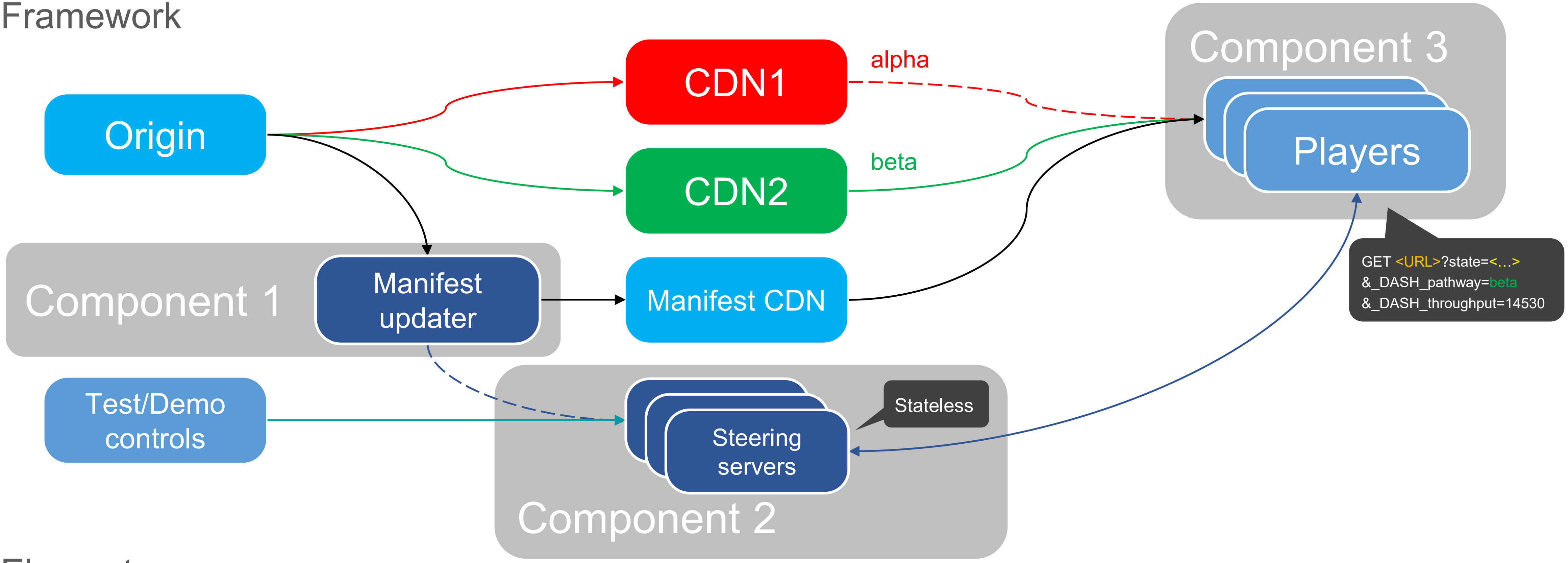
Benefits

- Scales well with CDNs or edge platforms.
- TTL can be smaller; comparable to player buffer delay; Can be used to optimize QOE!

OPEN SOURCE PROJECT IN SVTA

SEGMENTS:2024

Framework

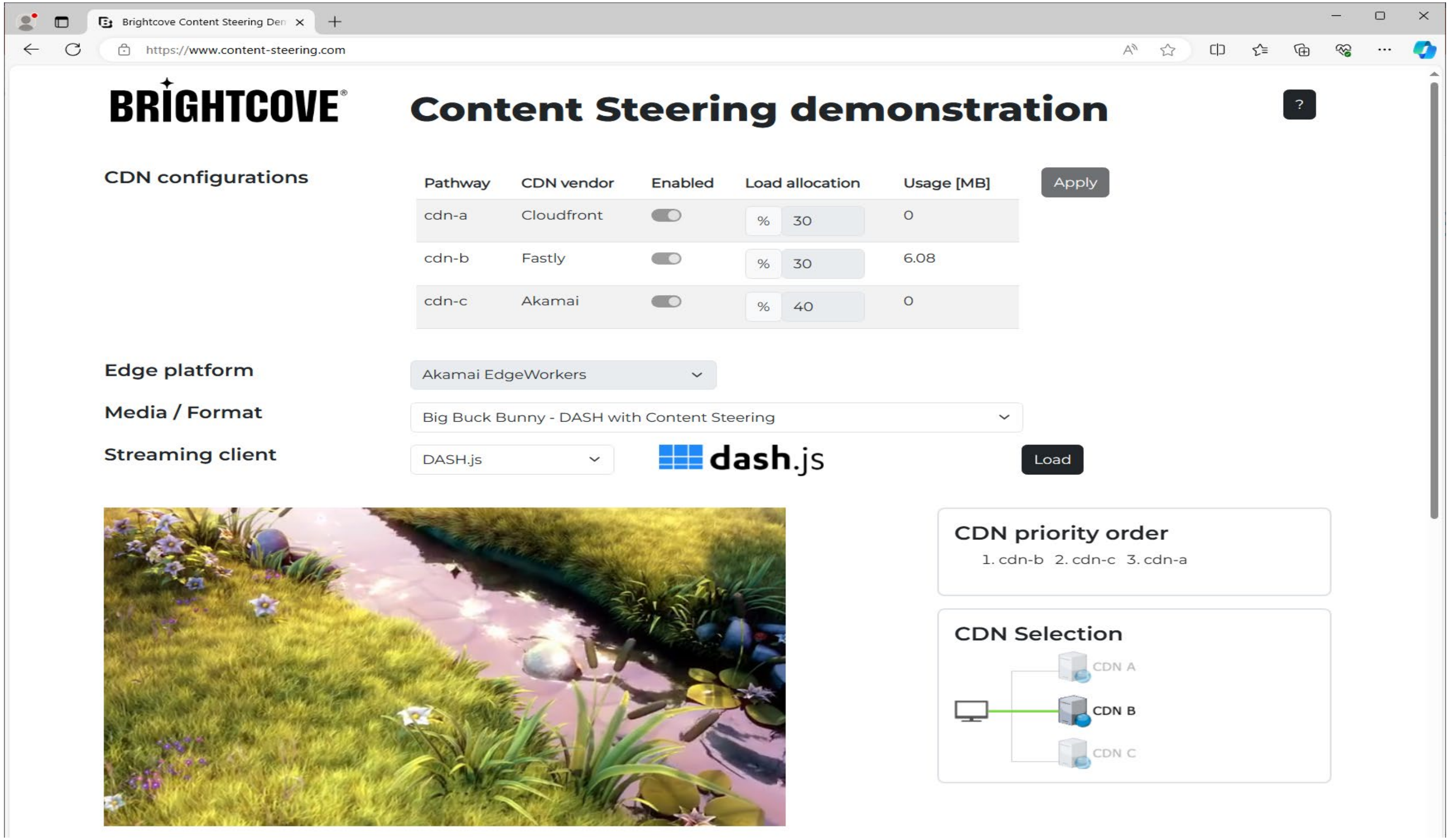


Elements:

- Component 1: manifest updater inserting content steering information in the manifests (Golang)
- Component 2: steering server implementation with several deployment variants (Node.js, Lambda @ Edge, etc.)
- Component 3: DASH and HLS players: DASH.js, HLS.js, video.js, shaka – existing open source projects

DEMO: www.content-steering.com

SEGMENTS:2024



TESTING FRAMEWORK

SEGMENTS:2024

Control panel:

https://v2.content-steering.com

Playback statistics

Volume

System	CDN A + B + C + Content Steering	CDN A	CDN B	CDN C
Video views	615	581	581	587
Seconds played	368432	351974	350868	345320
Traffic [GB]	54.41	53.46	53.42	53.72

QoS

System	CDN A + B + C + Content Steering	CDN A	CDN B	CDN C
Throughput [Mbps]	199.91	217.72	156.79	209.88
Throughput Std Dev [Mbps]	242.06	247.96	238.23	248.50
Latency [ms]	20.08	12.98	39.90	21.84
Latency Std Dev [ms]	95.02	141.01	1483.71	547.87
CDN switches	38	0	0	0

QoE

System	CDN A + B + C + Content Steering	CDN A	CDN B	CDN C
Start time [ms]	797.09	631.74	669.14	909.09
Re-buffering ratio	0.57	0.59	1.33	0.61
Re-buffering events	169	172	172	173
Video bitrate [Mbps]	3.46	3.44	3.42	3.41
Resolution [lines]	1558	1551	1546	1541
Rendition switches	664	617	603	662

Start playback session

System

Media / Format

Streaming client

Load

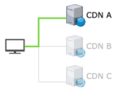
Playback sessions

https://v2.content-steering.com/player.html?session=eyJzZXNzaW9uSWQiOiIwMDAwMDAwMC0wMDAwLTAwMDAtMDAwMC0wMDAwMDAwMDAwMDAiLCJtYW5pZmVzdCI6ImRhc2gubXBkP3NyY:

Playback session using "CDN A"

CDN Selection

1. cdn-a



Session statistics

Volume

Seconds played	68.14
Traffic [GB]	0.01

QoS

Throughput [Mbps]	10.59
Throughput Std Dev [Mbps]	1.22
Latency [ms]	0.22
Latency Std Dev [ms]	0.08
CDN switches	0

QoE

Start time [ms]	744
Re-buffering ratio	0.00
Re-buffering events	0
Video bitrate [Mbps]	4531.00
Resolution [lines]	1920
Rendition switches	0

Fragment Requests

Type	Pathway	Request URL
Audio	cdn-a	https://cdn-a.content-steering.com/bbb/audio_128kbps/seg-69.m4f
Video	cdn-a	https://cdn-a.content-steering.com/bbb/video_1920x1080_4531kbps/seg-67.m4f

Steering Data

Request

Timestamp	2024-02-14T17:19:07.440Z
Steering URL	URL

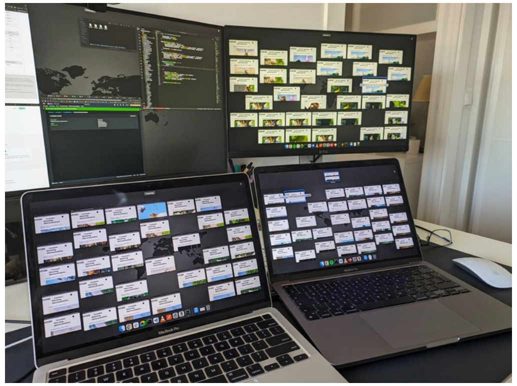
https://cdn-a.content-steering.com/dash.dcsm?steering_params=eyJtaW5CaXRyYXRlIjo5MTQ4NzgsImNkbk9yZGVyIjpbImNkbi1hIl0sInByZXZpb3VzVGhyb3VnaHB1dCI6Nzc4NTAwMCwicGF0aHdheXMiOlt7ImlkIjoiY2RuLWEiLCJ0aHJvdWdocHV0Ijo3OTUxOTQxfV0sInRpbWVzdGFtcCI6MTcwNzkzMTEzNjM0OX0%3D&_DASH_pathway=%22cdn-a%22&_DASH_throughput=4225250

Pathway cdn-a

Throughput 4225250 bps

Steering Params

Statistics collection



100000x

Real-world tests
Multiple sites / labs
No artificial throttling
The same conditions to different technologies

NEXT STEPS

1. Complete study / testing

2. Publish SVTA report on CDN switching

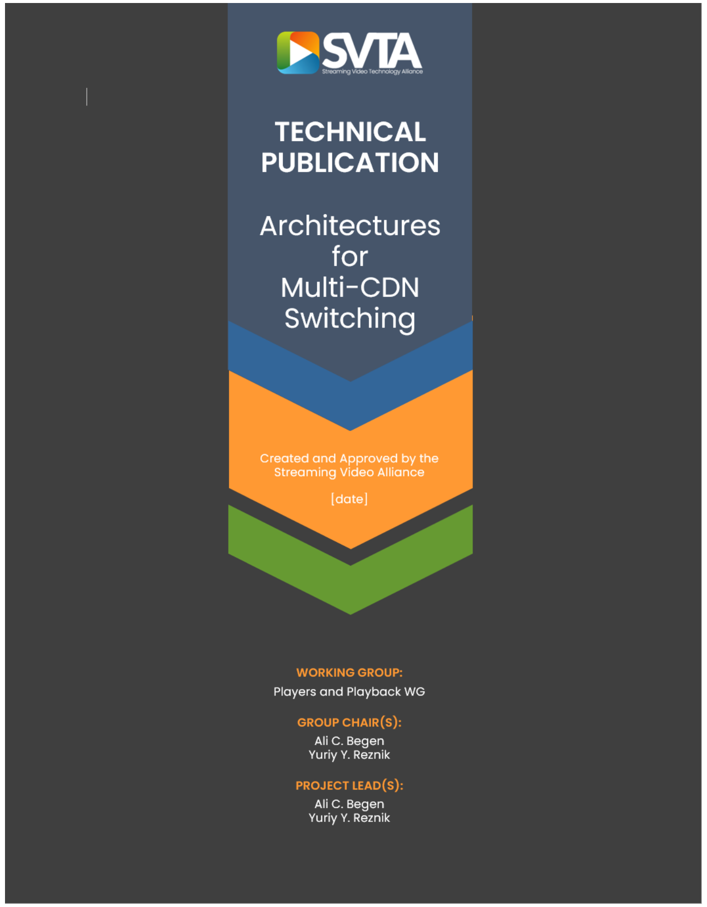
   - Problem explanation
   - Comparison of architectures
   - Summary of findings based on out tests

3. Synergies with other projects:

   - SVTA Open caching
   - SVTA QoE SG

4. Applications in hybrid delivery ecosystems

   - Wireless: 5G-MAG, 5G broadcast, etc.
   - Satellite assisted delivery: 5G-EMERGE, etc.



SVTA

TECHNICAL PUBLICATION

Architectures for Multi-CDN Switching

Created and Approved by the Streaming Video Alliance

[date]

WORKING GROUP:
Players and Playback WG

GROUP CHAIR(S):
Ali C. Begen
Yuriy Y. Reznik

PROJECT LEAD(S):
Ali C. Begen
Yuriy Y. Reznik

SEGMENTS:2024

#SEGMENTS2024

Organized by SVTA

Streaming Video Technology Alliance