TECHNICAL WHITEPAPER

SVTA1108: CMSD to Enhance Media Streaming: A White Paper

Created and Approved by the Streaming Video Technology Alliance

January 5, 2026

WORKING GROUP:

DASH-IF WG

GROUP CHAIR(S):

Thomas Stockhammer (Qualcomm)
Yuriy Reznik (SME)

PROJECT LEAD(S):

Mohammed Raad (Unified Streaming)

1. CONTRIBUTORS

The following people contributed to this document:

- Will Law (Akamai)
- Bertrand Berthelot (Broadpeak)
- Thomas Stockhammer (Qualcomm)
- Mohamed Raad (Unified Streaming)
- Roberto Ramos (Unified Streaming)
- Yuriy Reznik (Streaming Labs)

2. ABSTRACT

This whitepaper presents a collection of results from studies of the usage of Common Media Server Data (CMSD) parameters to enhance media streaming performance. The presented results show that the CMSD keys “availability time”, “startup flag”, and estimated throughput have a positive impact on quality of experience.

2.1. Versioning

Revision	original publication	Date	January 5, 2026
• Published document			

Table 1: Document versioning

3. OBJECTIVES AND SCOPE

The objectives of this document are to:

- Present a study of several potential uses of CMSD parameters in media delivery systems
- Evaluate the impacts of the proposed uses of CMSD parameters on the performance of players and CDNs
- Confirm the positive effects on the performance of CDNs and players, and consequently on the overall quality of experience delivered to end users.

3.1. Scope

Scope of this document ADDRESSES:

- Informative results related to the use of CMSD parameters to enhance streaming services' performance as experienced by the consumer of the streamed content.

Scope of this document DOES NOT ADDRESS:

- Any normative aspects of the use of CMSD

TABLE OF CONTENTS

4. ACRONYMS AND DEFINITIONS

This document uses the following acronyms and other terms. It is a quick reference and a snapshot in time. For the most up to date acronyms and definitions along with other terms and references please refer to the SVTA Wiki at https://wiki.svta.org/.

Abbreviation	Description
at	Availability time
br	Encoded bitrate
d	Object duration
du	Duress
ept	Earliest presentation time
etp	Estimated throughput
ht	Held time
mb	Max suggested bitrate
n	Intermediary identifier
nor	Next Object Response
nrr	Next range response
ot	Object type
rd	Response delay
rtt	Round trip time
sf	Streaming format
st	Stream type
su	Startup

Table 2: Acronyms and Definitions

5. STATIC CMSD PARAMETERS

Static parameters are set by the origin server and remain constant from origin to player. Each of the parameters is expected to add to the quality of experience of the viewer.

5.1. Availability time "at"

Semantics: The time at which the first byte of the object became available at the origin

Unified Streaming[1] established an experimental testbed to study the impact of the CMSD "at" parameter on latency upstream of the Origin server. This testbed focused on live streaming scenarios in which a live encoder sends media to the Origin using an HTTP POST method. The experiments utilized a modified dash.js version to enable parsing and displaying the "at" and the new proposed Earliest Presentation "ept" parameter in the dash.js reference client. The "ept" is the Timestamp of the ProducerReferenceTimeBox [ISOBMFF] of the first media sample in milliseconds since epoch time (1970-01-01-00:00:00 UTC). The evaluation of the testbed included an emulation of latency between the live encoder and the Origin server.

Figure 1 illustrates the CMSD-based deployment scenario. Figure 2 shows the modified dash.js[2] reference player with the "at" and "ept" values after an emulated delay is identified, and after removing the emulated delay.

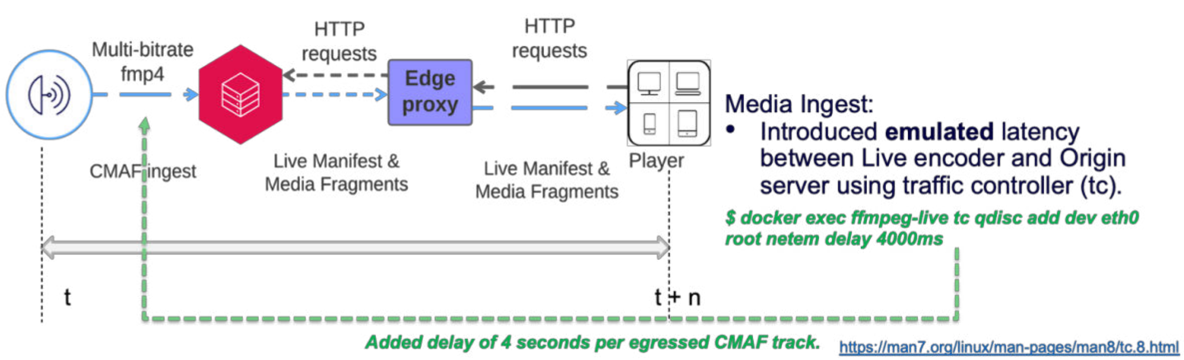

Figure 1: Unified Streaming testbed for the "at" parameter study

[1] https://www.unified-streaming.com/
[2] https://dashjs.org/

Added emulated delay between the live encoder and the Origin server. | **After removing** emulated delay between the live encoder and the Origin server.

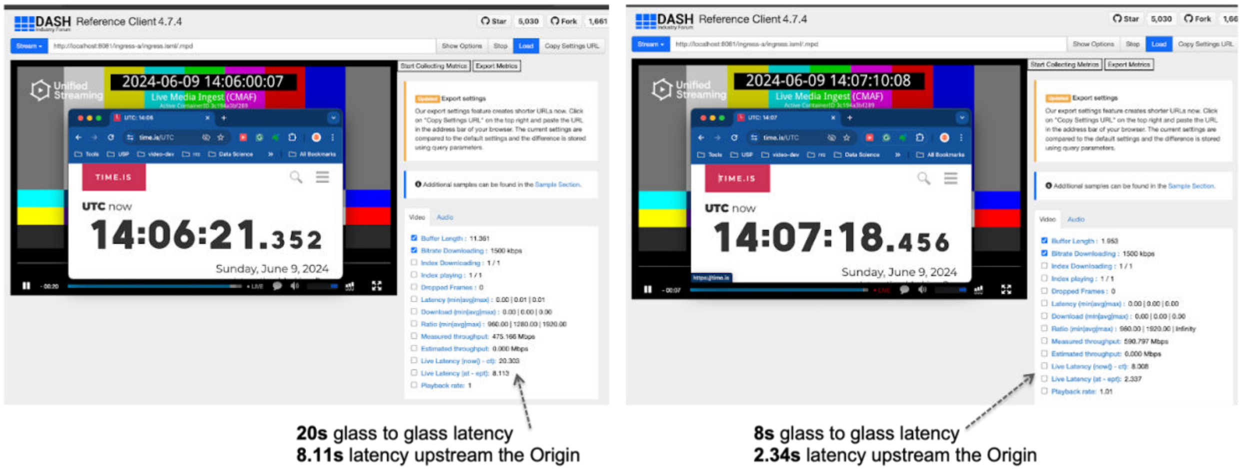

Figure 2: The result of removing the identified delay.

The "at" and the proposed "ept" parameter were used to identify the latency upstream of the Origin server by calculating the difference between “at” and “ept”.

The following advantages of using these two metrics were identified:

- The "at" and "ept" key-value pairs can provide helpful information to the player and intermediate servers about the upstream latency from the Origin server.
- Media players can benefit from this timing information to improve their ABR algorithm.
- Based on this latency information, it is easy to tune the target delay value at the media player.
- Intermediate services (e.g., delivery workflow health checks and control servers) can use this information to identify potential upstream issues and act before they can impact the quality of experience in user devices.

5.2. Startup “su” Flag

Semantics: A flag indicating if the object is needed for the start of the stream.

Unified Streaming and Varnish software[3] established an experimental testbed to study the impact of the use of the CMSD "su" parameter on video start-up time. The conducted experiments used dash.js running in Chrome as the client, Varnish frontend caching, Unified origin, Varnish backend caching, and AWS S3 for content storage. Multiple combinations of these components were experimented with to better understand the impact of the "su" parameter on video start-up time.

Figure 3 illustrates the architecture of the Unified Streaming-Varnish testbed.

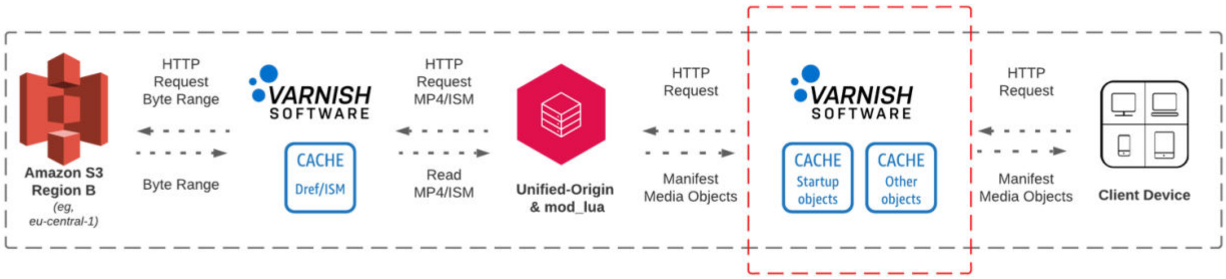

Figure 3: Testbed used to study the impact of the "su" parameter

The "su" parameter in this testbed was used to identify media objects needed for video start-up. In the scenario where a frontend cache was used, these objects were cached separately from other objects. This enabled higher persistence of such objects in the frontend cache and reduced their retrieval time. This related streaming optimization concept is generally known as "prefix caching"[4], implemented in this case by the CMSD "su" parameters and the Varnish cache's policy for caching "su"-marked objects differently than other objects.

The scenarios tested were:

1. Origin only, with no caching.
2. Origin with a backend cache.
3. Origin with a frontend and backend cache, but CMSD "su" is not used.
4. The same as (3) but with the CMSD "su" parameter being used.

[3] https://www.varnish-software.com/

[4] S. Sen, J. Rexford and D. Towsley, "Proxy prefix caching for multimedia streams," *IEEE INFOCOM'99*, New York, NY, USA, 1999, pp. 1310-1319.

It was found that the scenario with the “su” parameter delivered the shortest (and hence, best) video start-up time, followed by the third, second, and first scenarios in that order.

In comparing the third and fourth scenarios, it was observed that using the “su” parameter almost halved the video start-up time. The fourth (“su” used) scenario was found to have a video startup time of approximately 60% less than the second (only backend caching is used). Finally, the CMSD-based scenario (the fourth) delivered a 75% reduction in video start-up time compared to the first scenario. It should be noted that a random network delay with an average of 50ms was included in all the scenarios tested.

6. DYNAMIC CMSD PARAMETERS

Dynamic CMSD parameters remain constant for only a single hop in the chain from the origin to the player. Supporting intermediary clients (servers) are expected to update the values of these parameters.

6.1. Estimated Throughput "etp"

Semantics: "etp" is the throughput, in kbps as an integer value, between the server and the client over the currently negotiated transport as estimated by the server at the start of the response. The throughput may vary during the response, and the client should use this data as an adjunct to its own throughput estimates.

The estimated throughput, as calculated by an edge server, can be used by a player to improve the global playback Quality of Experience (QoE).

At the startup phase, an estimated throughput value can hint at an appropriate starting bitrate level, allowing the player to experience higher quality sooner.

Server-side throughput estimation can also be used for the entire playback session to improve the rate-adaptation algorithms, specifically for low-latency streams based on chunked transfer encoding. The bandwidth estimation at the server side can rely on the transport layer's congestion control layer, leading to more precise estimates.

Broadpeak[5] established an experimental testbed to study the impact of the CMSD etp parameter on the throughput calculation and estimation, and consequently on the ABR heuristic performance and global QoE specifically for live low-latency streaming based on chunked transfer encoding.

The conducted experiments utilized dash.js running in Chrome as the client, along with a Broadpeak origin server and edge cache server, which were able to deliver DASH segments using chunked transfer encoding. The edge server can provide an accurate throughput measurement based on the congestion control module attached to the transport layer, such as BBR (Bottleneck Bandwidth and Round-trip propagation time).

The experimentation consisted of comparing the ABR behavior and overall playback experience of dash.js player when estimating throughput:

[5] https://broadpeak.tv/

- based on “moof” [ISOBMFF] parsing performed on the client side
- or based on the CMSD “etp” value sent by the edge cache server.

During the experiments, traffic shaping was applied to the output network layer of the edge cache server to simulate limited network capacity and available bandwidth between the edge server and the player.

Figures 4 and 5 illustrate the player behavior regarding ABR logic and selected video bitrate (“vb” curve) during a low latency streaming session (of 150s duration) with traffic shaping in cascade mode (“tc” curve) in the two modes, i.e.:

- Baseline mode using default dash.js configuration with throughput calculation based on “moof” parsing (“ctp” curve)
- CMSD etp mode with throughput calculation exclusively based on CMSD “etp” values conveyed by edge cache server to the player (“etp” curve)

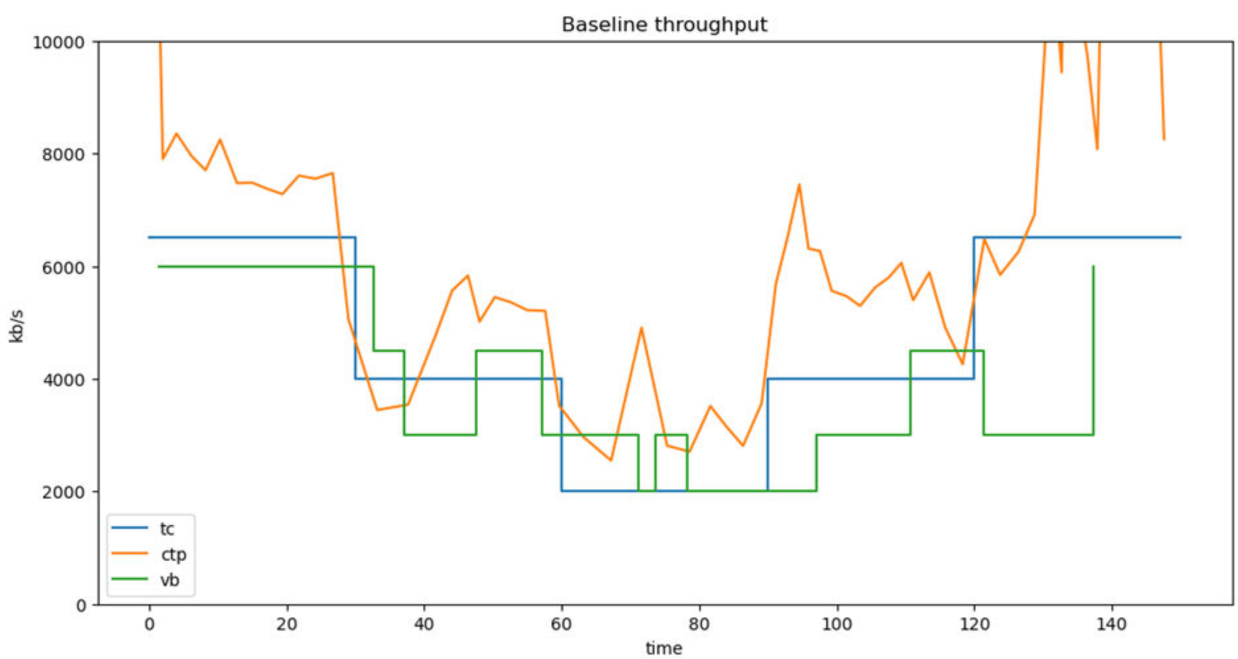

Figure 4: Player video bit rate calculation based on moof parsing

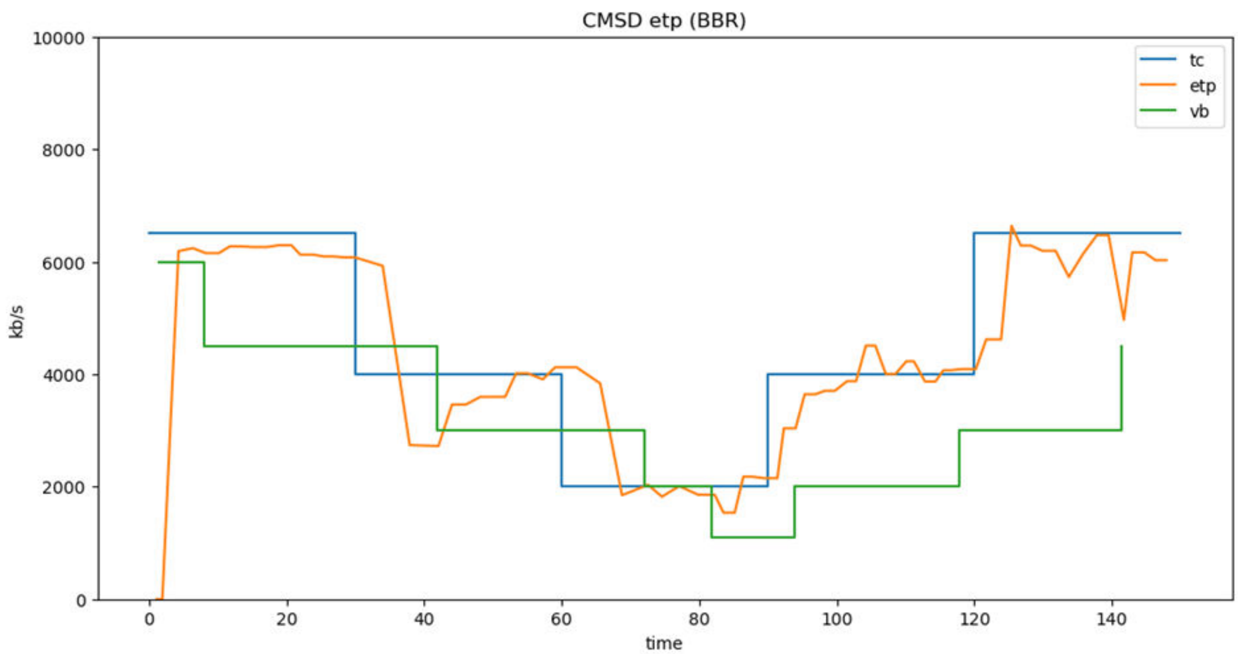

Figure 5: Player video bit rate calculation based on CMSD "etp"

This experiment demonstrated that baseline mode with throughput calculation based on moof box parsing always overestimates the available bandwidth, leading to unsettled ABR decisioning with ping-pong effect and therefore leading to buffer starvation and stalls.

In contrast, when relying on server-side estimated throughput, this throughput sticks to real available bandwidth and, hence, the ABR heuristic is more consistent, thus allowing the player to select the highest possible bitrate without compromising playback QoE.

To measure the impact on the global QoS, the same tests were executed over 100 playback sessions of 150s each for each configuration. The following figures provide a summary of the stall duration amplitudes and mean observed, as well as the mean selected video bitrates when using the "moof" based approach in comparison to using the "etp" approach.

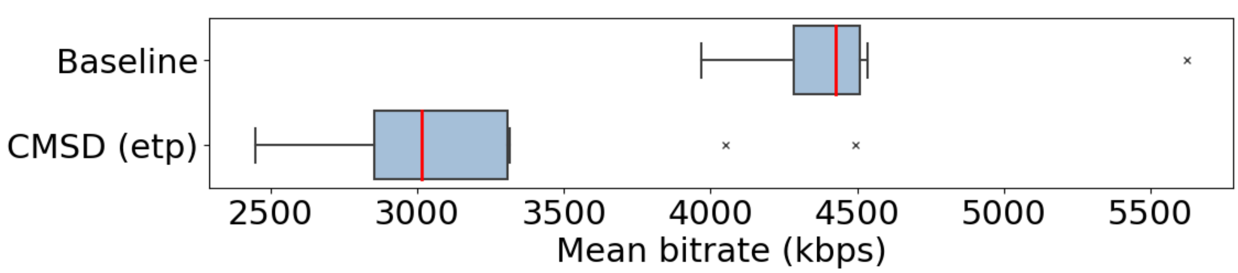

Figure 6: Mean bitrate selected by player based on moof parsing vs CMSD etp use

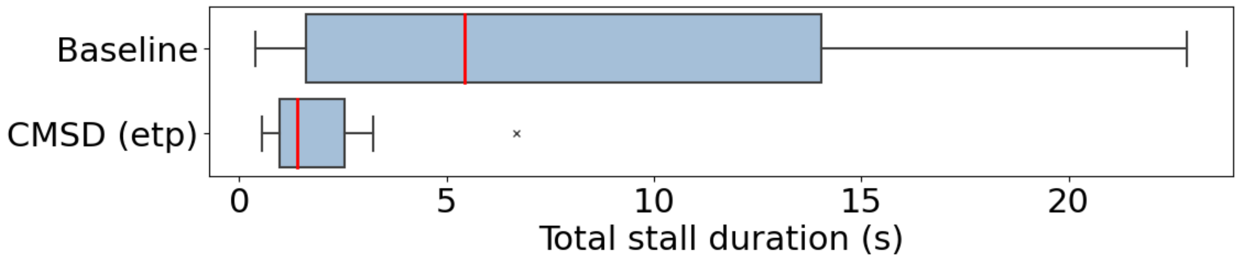

Figure 7: Stall duration by the player based on moof parsing (baseline) vs CMSD etp use

It was observed (as shown in Figures 6 and 7) that client-side estimation (the baseline "moof" approach) results in the player ABR logic selecting higher bitrates, which in turn leads to a high number of stalls, with longer duration, and degrades the global QoE of the playback sessions compared to sessions based on server-side throughput estimation.

7. CONCLUSION

This paper has summarized the results from three studies into the use of CMSD parameters:

- "at" - availability time,
- "su" - startup flag, and
- "etp" - estimated throughput

The "at" parameter was shown to contribute to the reduction of latency. The "su" flag was shown to reduce video startup time, and the "etp" was shown to reduce the average stall duration as well as contribute to a smoother video bitrate.

Although only a few of the defined CMSD parameters have been tested in practical settings, it is evident that the use of these parameters results in an enhanced quality of experience.

Naturally, the complete set of CMSD parameters offers many additional utilities and optimizations that may enable further streaming workflow optimizations and improvements in quality of experience.

8. SPECIFICATIONS AND STANDARDS REFERENCES

The following are the survey questions:

Reference	Specification	More Information
[CMSD]	Common Media Server Data, CTA-5006, Consumer Technology Association	November 2022 version, Published URL: *https://cdn.cta.tech/cta/media/media/resources/standards/pdfs/cta-5006-final.pdf*
[ISOBMFF]	Information technology — Coding of audio-visual objects — Part 12: ISO base media file format	URL: https://www.iso.org/standard/85596.html

Table 3: Specification and standards references

9. TABLES AND FIGURES

9.1. Tables

9.2. Figures

10. ABOUT THE STREAMING VIDEO TECHNOLOGY ALLIANCE

Composed of members from across the video ecosystem, the Streaming Video Technology Alliance is the global association that works to solve critical streaming video challenges to improve end-user experience and adoption. The organization focuses on three main activities: first is to educate the industry on challenges, technologies, and trends through informative, publicly available resources such as whitepapers, articles, and e-books; second is to foster collaboration among different video ecosystem players through working groups, quarterly meetings, and conferences; third is to define solutions for streaming video challenges by producing specifications, best practices, and other technical documentation. For more information, please visit https://www.svta.org.

Streaming Video Technology Alliance
5177 Brandin Court
Fremont, CA 94538 USA
svta.org