SVTA
Streaming Video Technology Alliance

OSMART 2024

December 5, 2024

Agenda

OSMART 2024

01 SVTA and DASH-IF
Presenter: Jason Thibeault

02 SVTA Sandbox
Presenter: Jason Thibeault

03 Common Media Library
Presenter: Casey Occhialini

04 dash.js
Presenter: Daniel Silhavy

05 LiveSim2
Presenter: Torbjörn Einarsson

SVTA and DASH-IF

Presenter: Jason Thibeault

SVTA and DASH-IF

General Overview

- DASH-IF and SVTA Merged in July 2024
- DASH-IF became the DASH-IF Working Group underneath the SVTA umbrella
- The DASH-IF WG continues to work on all existing work items
- For 2025, the SVTA will look for ways to better integrate DASH-IF WG items into other SVTA groups. One example would be to merge the DASH-IF security work into the SVTA Security WG
- The SVTA is committed to supporting DASH-IF open-source efforts like dash.js and LiveSim2

SVTA Sandbox

Presenter: Jason Thibeault

SVTA Sandbox

General Overview

- The objective is to create a cloud-based platform where SVTA working groups can spin-up streaming workflows for testing
- The Low Latency Streaming Working Group was the first to put this into practice but it was done so manually
- Dan Drew of Fortium Partners, an outsourced CTO firm, agreed to take on the automation of the process

SVTA Sandbox

Phase 1

- Terraform recipes for virtualized workflow components (such as Synamedia encoder, FFMPEG, etc.)
- CLI spin-up process using Terraform to instantiate Google Cloud infrastructure to support workflow components
- A library of encoded content and live sources for use with the workflow

SVTA Sandbox

What is next?

- Phase 2 will incorporate a web-based interface to abstract the Terraform CLI process
  - Working Groups with approved projects will be able to use checkboxes and other inputs to specify workflow components
  - A submit button will spin-up the infrastructure
- Phase 3 will include guardrails including automated budget analysis and suspension of project operation when budget has been exceeded.
- We are currently exploring the possibility of open sourcing the sandbox but for the moment, it will be available only to SVTA member companies.

SVTA Sandbox

Synergies to other projects

- SVTA Sandbox will incorporate other OSS projects as Terraform recipes for inclusion in SVTA WG lab/testing projects. This could include:
  - dash.js as the player and include CML components
  - FFMPEG as primary encoder
  - GPAC as stream manager
  - NGINX/Varnish as reverse proxy

SVTA Sandbox

Participation / Where to find us

- The SVTA Sandbox is not currently part of any specific group
- We are looking to build a project around it within the Streaming Video Operations Group. That should happen Q1 2025.
- Until then, if you are interested in participating, please reach out to jt@svta.org to get involved.

Common Media Library

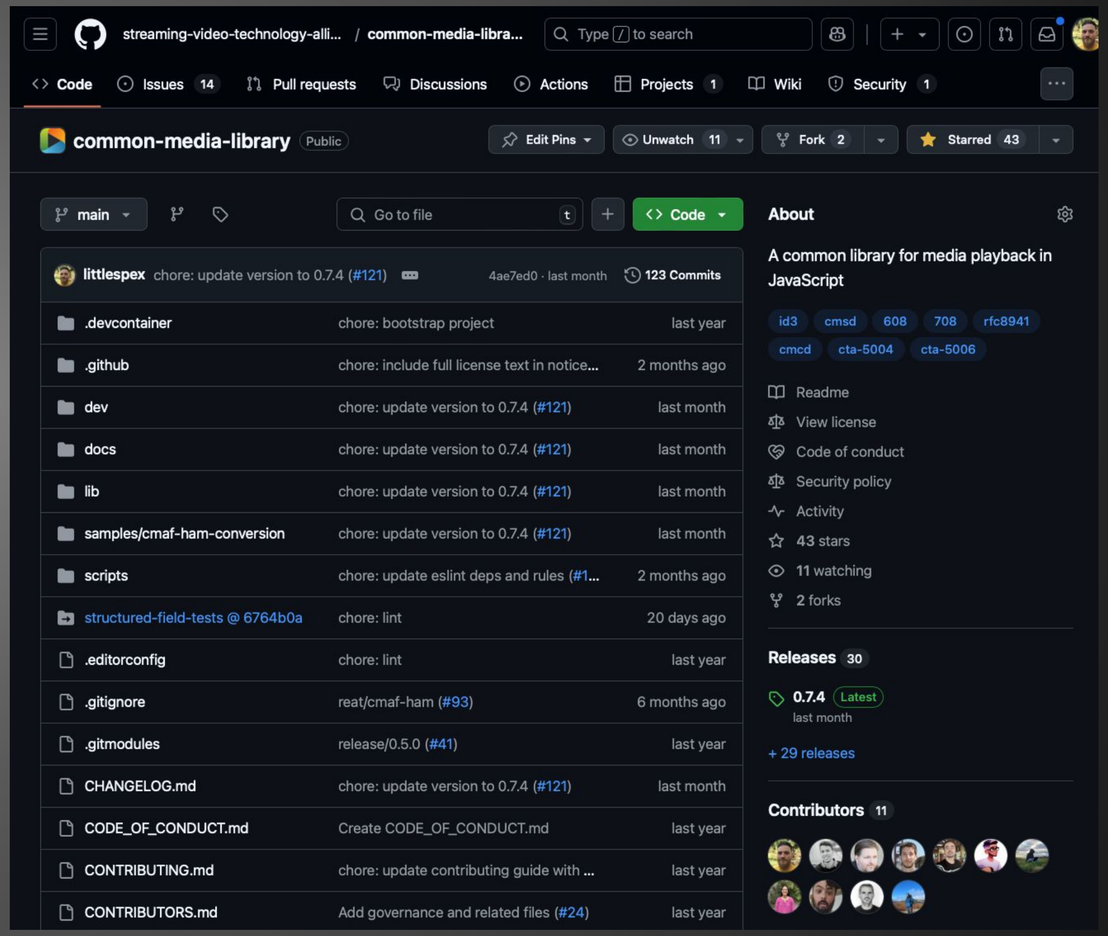
Presenter: Casey Occhialini

Common Media Library

General Overview

The Common Media Library (CML) is a modern Javascript utility library for media processing and playback.

- Maintained by the SVTA, but open to everyone for contributions
- Written in TypeScript and modern build tools
- Compiled to ES6 with Typescript declarations
- No bundling or transpiling to prevent conflicts with app build tool chains.
- Focused on modularity. Only import what is needed.
- Full API documentation
- Both unit and integration test suites



Common Media Library

Features and Current Work Items

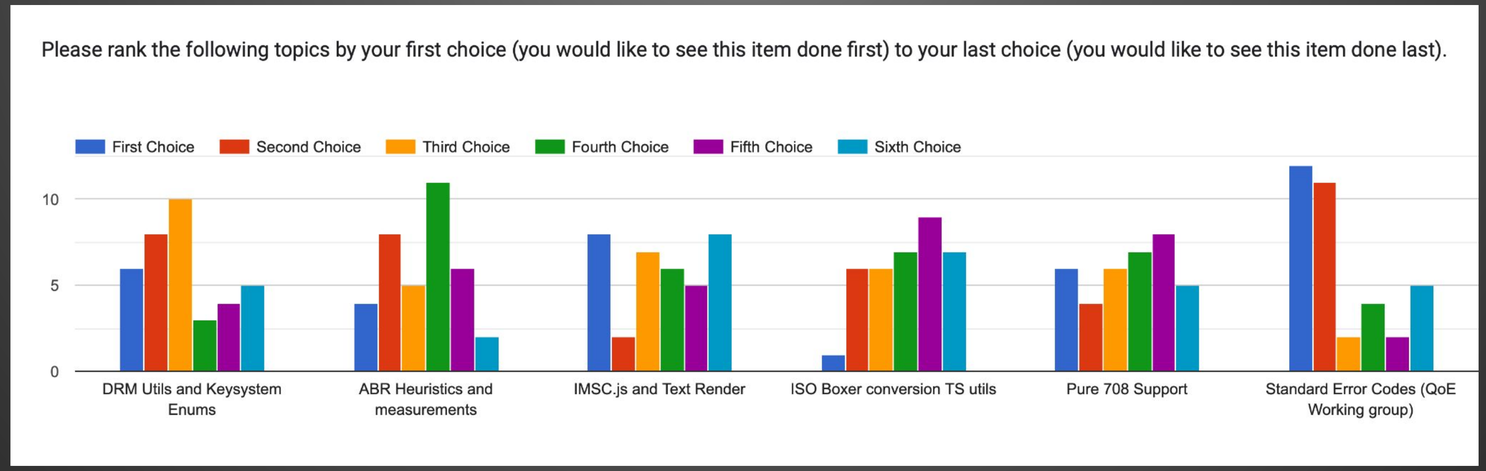
Current libraries include:

- CTA CMCD - Common Media Client Data
  - The addition of Structured Field Values utils sets us up for v2
- CTA CMSD - Common Media Server Data
- ID3 parsing
- Common Media Request/Response interfaces
  - Helps standardize request/response transformers across players
- CTA 608 caption parsing
- CMAF HAM - Hypothetical Application Model
  - Provides a common model to describe both HLS and DASH manifests

Common Media Library

What is next?

- IMSC and Text rendering
- ISO Box Parsing
  - Port of codem-isoboxer with focus on modularity and strict data types
- DRM Utilities
- Common utilities for throughput calculations
- ABR Framework
- CTA 708 parsing and rendering
- SVTA Standard Error Codes



Common Media Library

Synergies to other projects

The library is now integrated in the following projects:

- hls.js - CMCD and ID3
- dash.js - CMCD, CMSD, and ID3
- @montevideo-tech/videojs-cmcd - video.js CMCD plugin
- Working on a shaka-player integration
- Integrated into Paramount's video player

hls.js dash.js qualabs Paramount

Common Media Library

Participation / Where to find us

CML is public project. You don't have to be a SVTA member to contribute or attend the monthly meetings.

- Website:
  https://cml.svta.org
- Repo:
  https://github.com/streaming-video-technology-alliance/common-media-library
- NPM Package:
  @svta/common-media-library
- API Docs:
  https://streaming-video-technology-alliance.github.io/common-media-library/

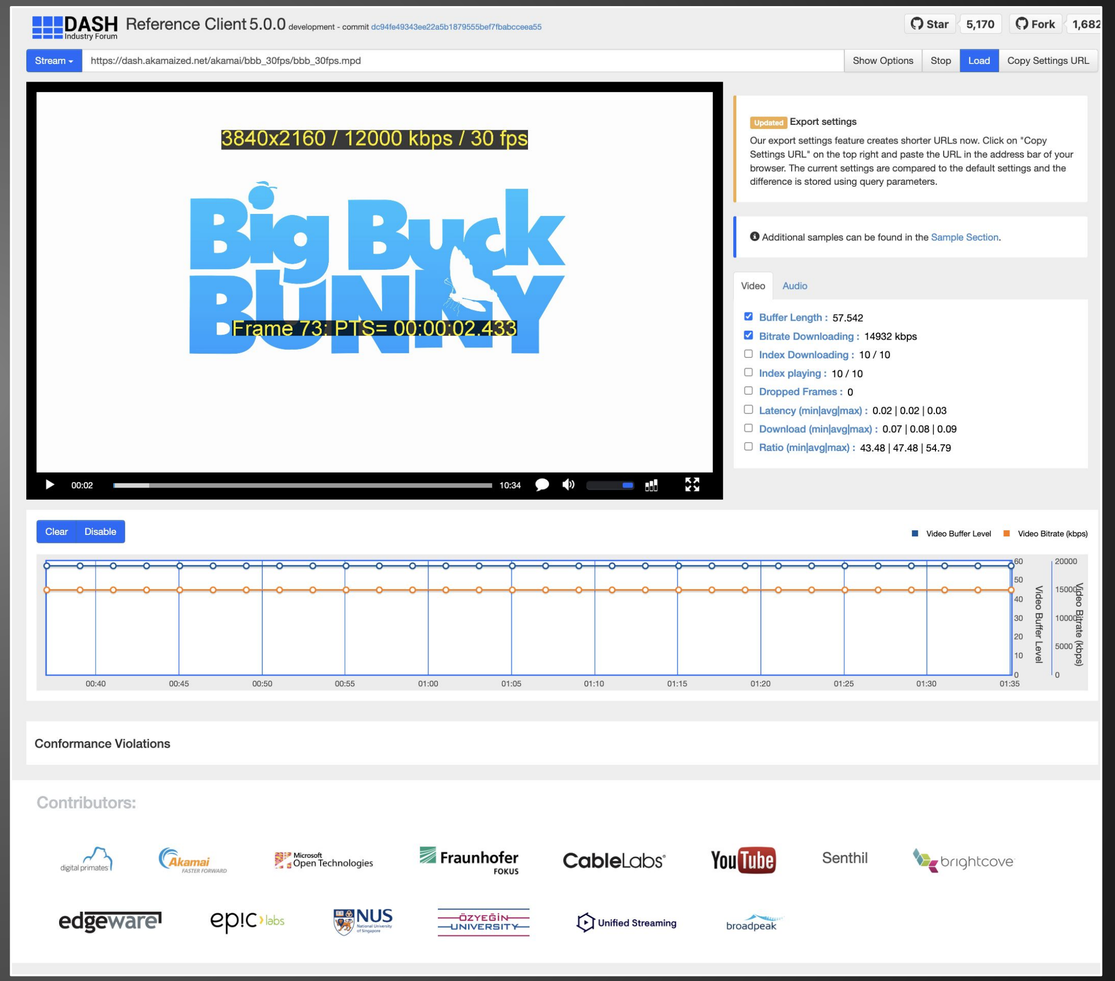
dash.js

Presenter: Daniel Silhavy

dash.js

General Overview

- dash.js is the official reference player by SVTA / DASH Industry Forum for playback of MPEG-DASH content
- Maintained by Fraunhofer FOKUS, community driven development
- Used as a reference client for standardization, foundation for production-grade video applications and for research purposes e.g., implementing new ABR algorithms
- Open-source project on Github - https://github.com/Dash-Industry-Forum/dash.js/ , last released version 4.7.4
- Various features including support for ABR, multiperiod, DRM, MPD patching, Gap handling, CMCD, CMSD, Content Steering, low latency, various subtitle formats (TTML, IMSC1, WebVTT) and many more



dash.js

Features and Current Work Items

- Finalizing work on new major version 5.
  - Project board: https://github.com/orgs/Dash-Industry-Forum/projects/8
- Main features
  - Improved XML parsing times, especially on low-end devices
  - Improved ABR and Throughput logic including support for Resource Timing API, Network Information API and new throughput calculation modes
  - Support for Adaptation-Set Switching
  - Native handling of key status changes
  - Integration of CML for CMCD, CMSD, CEA-608 parsing
  - Support for forced subtitles
  - Initial support for CMCD v2 and DASH Annex I
  - Extended functional testsuite
  - And many more…

dash.js

What is next?

- Finalize last open issues for v5
- Add support for additional CMCD v2 parameters
- Media Presentation Insertion
- SegmentTimeline@Pattern
- L3D-DASH
- Content Steering v1.0.0
- C2PA

dash.js

Synergies to other projects

- DVB-I Reference Client
- HbbTV Reference Application
- 5G-MAG – 5G Media Streaming and 5G Broadcast
- Common Media Library
- Livesim2

dash.js

Participation / Where to find us

- Github project: https://github.com/Dash-Industry-Forum/dash.js
- Reference client: https://reference.dashif.org/dash.js/nightly/samples/dash-if-reference-player/index.html
- Samples: https://reference.dashif.org/dash.js/nightly/samples/index.html
- Documentation: https://dashif.org/dash.js/
- API documentation: http://cdn.dashjs.org/latest/jsdoc/module-MediaPlayer.html
- Slack Channel: https://dashif-slack.azurewebsites.net/
- Google Groups: https://groups.google.com/g/dashjs

Annual face to face meeting in Berlin co-located with Fraunhofer FOKUS Media Web Symposium

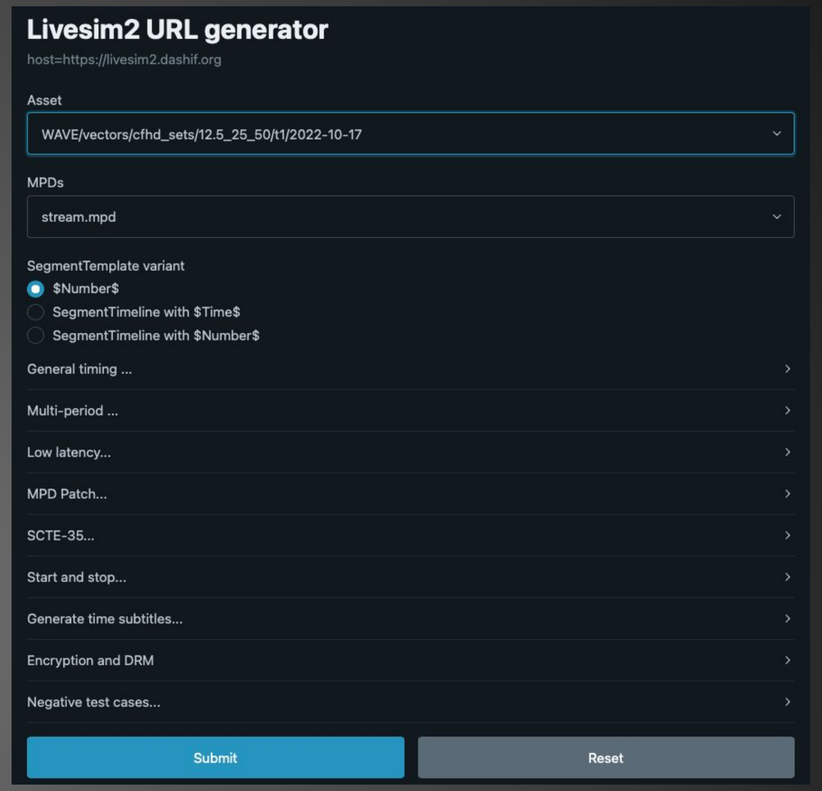
LiveSim2

Presenter: Torbjörn Einarsson

LiveSim2

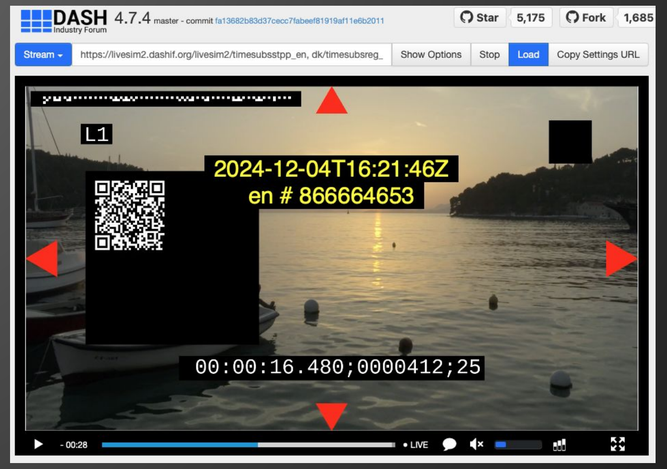
General Overview

- DASH_IF/SVTA tool
- Superconfigurable MPEG DASH live test source
  - Stateless, generating segment/MPds based on URL parameters
- Makes live streams from VoD assets
  - Wall-clock synchronized, e.g. 1h video wraps exactly at full hour
- Written in Go
- Publicly hosted service: https://livesim2.dashif.org
- Easy to set up locally



Some features:

- Low-latency DASH refragmentation on the fly
- Encryption on the fly based on CPIX license data
- Can emulate server issues
- Can add time-synchronized subtitles on the fly
- A URL Generator helps with parameterized URLs
- Integrates with dash.js for immediate playback

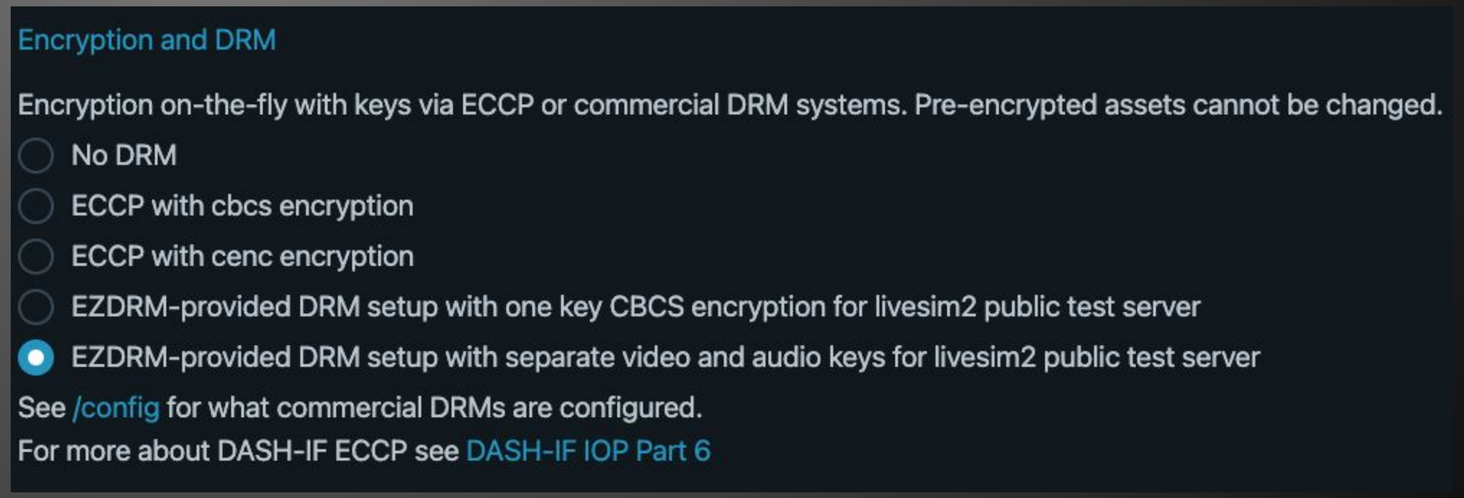


LiveSim2

Features and Current Work Items

Added in the last year:

- MPD Patch support
- DASH-IF CMAF Ingest source
- A CMAF Ingest test receiver
- Support for pre-encrypted content (HbbTV request)
- On-the-fly encryption with Widevine and PlayReady licenses from EZDRM



LiveSim2

What is next?

- Annex I support (segment query URLs)
- CATS test case
- Long wish list of DASH Ed. 6 features
  - Alternative MPDs and SGAI
  - L3D
  - SegmentTimeline Pattern
  - ...

- Currently very limited funding:
  - Maintenance level: 2h/week
- Sponsoring of specific feature implementations is welcome

LiveSim2

Synergies to other projects

- dash.js - live stream test source including new DASH-IF features
- HbbTV test suite - live stream test source
- Shaka-player - live stream test source
- ... lots of sites using the public service

LiveSim2

Participation / Where to find us

- Publicly available source: https://livesim2.dashif.org
- URL generator: https://livesim2.dashif.org/urlgen
- Github project: https://github.com/Dash-Industry-Forum/livesim2
- Github Wiki: https://github.com/Dash-Industry-Forum/livesim2-wiki
- Github planning: https://github.com/orgs/Dash-Industry-Forum/projects/7
- livesim2 1hour-clock content: https://github.com/Dash-Industry-Forum/livesim-content
- DASH-IF Slack Channel: #livesim

Contact: torbjorn.einarsson@eyevinn.se

eyevinn
technology

SVTA
Streaming Video Technology Alliance

THANKS FOR YOUR TIME

Enjoy the meeting!