INTEROPERABILITY FOR INTERNET TV SERVICES

Dr. Thomas Stockhammer
Qualcomm Incorporated

August 1, 2018
Mile High Video, Denver

Qualcomm

OUTLINE

Qualcomm Perspective

Problems and Challenges

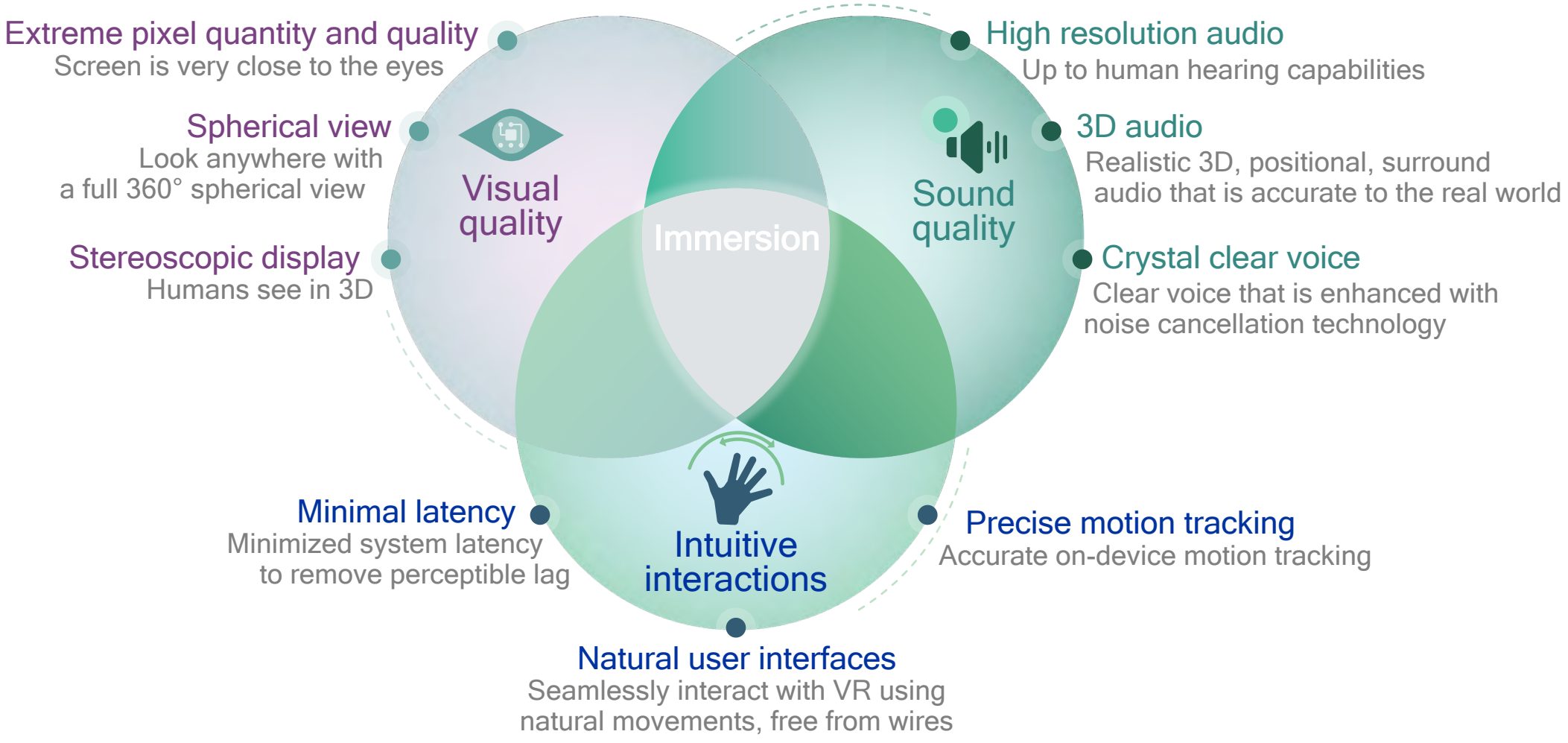
Selected Standardisation efforts

- DASH-IF
- Low-Latency DASH
- CMAF and CTA WAVE
- DVB-I
- 3GPP enTV, 5G and Hybrid Services

Summary

WE WANT TO IMMERSE YOU

Immersion is enabled by different components that work together



Learn more about our vision for the future of VR: www.qualcomm.com/VR

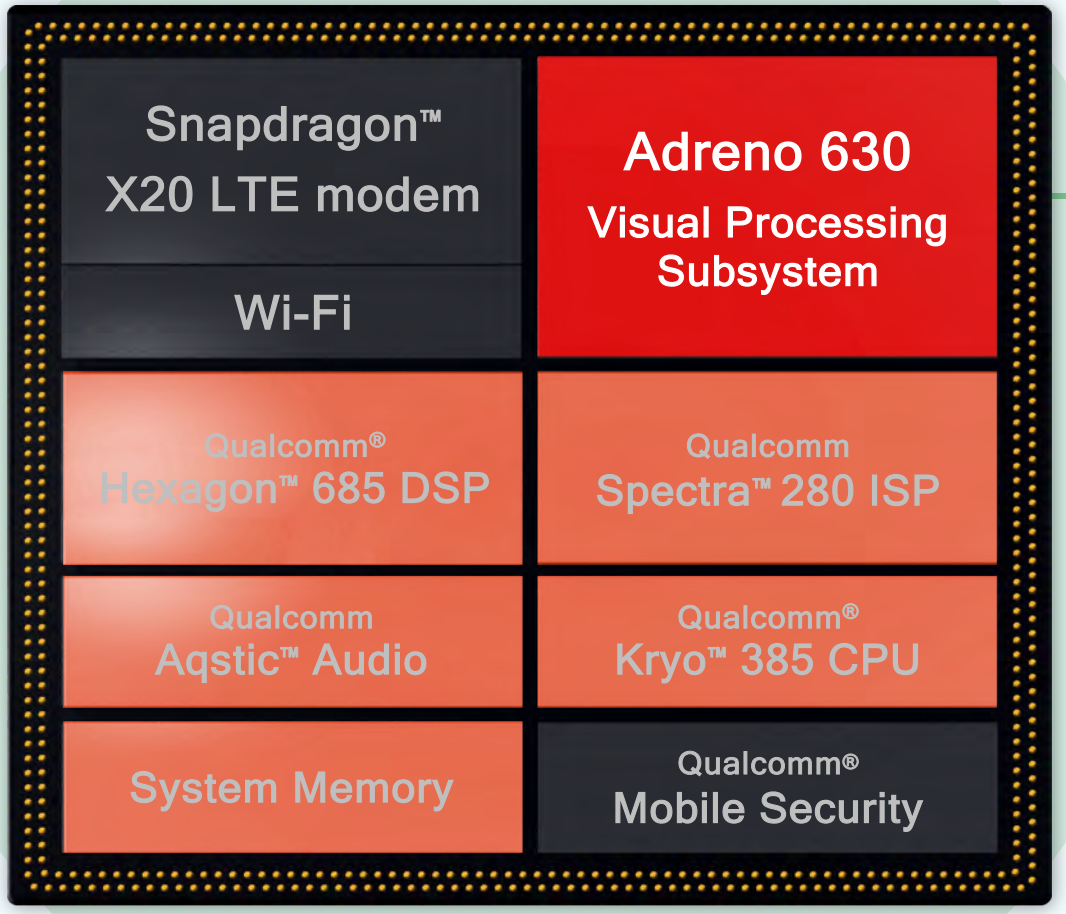
SNAPDRAGON 845

Multimedia/XR/AR

Computer vision, image processing, sensor processing, graphics, video processing, location, and cloud interaction

Benefits

- Integrated and optimized
- Enhanced battery life
- Thermal efficiency
- Standardized implementation
- Mass market cost
- Variety of use cases and industry support



Peak Download Speed: 1.2 Gbps

Peak Upload Speed: 150 Mbps

Ultra HD Premium video playback and encoding @ 4K (3840x2160) 60fps, 10bit HDR, Rec 2020 color gamut

eXtended Reality (XR)

Sensors

Qualcomm® Snapdragon™ Neural Processing Engine (NPE) SDK

Entire SoC is used!

*Compared to Snapdragon 835

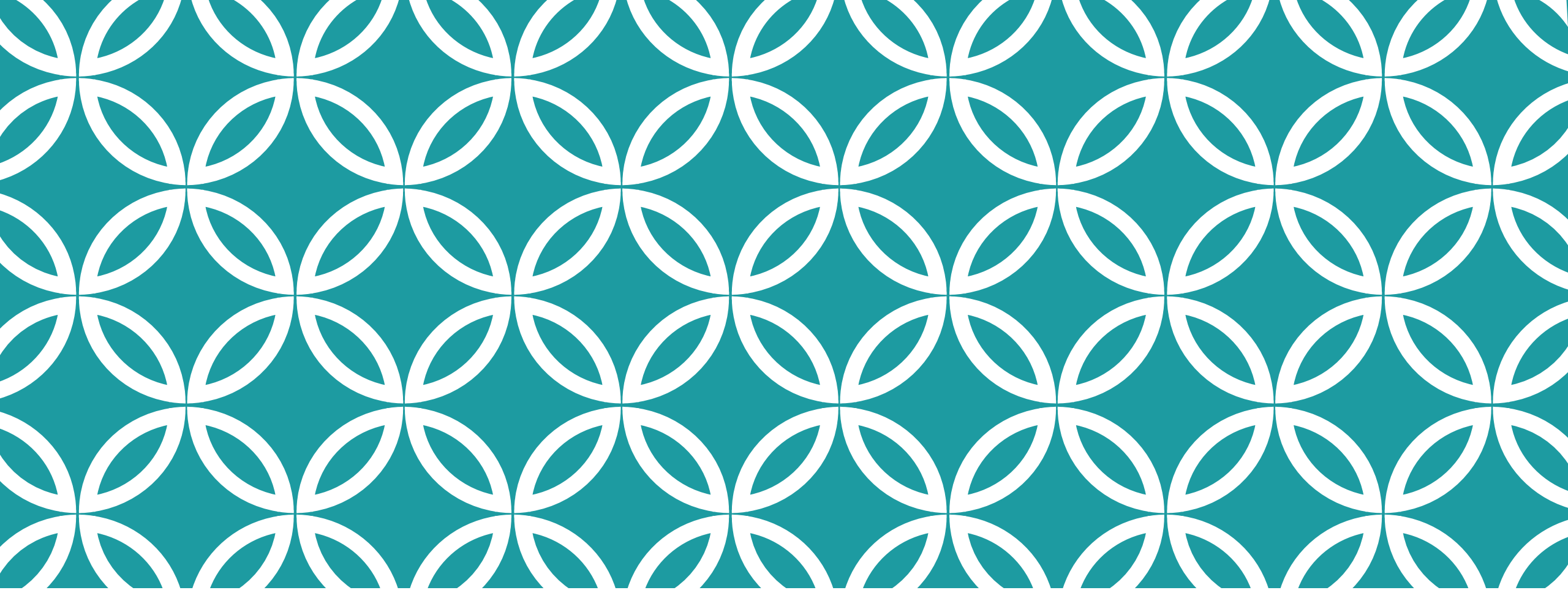
SOME OBJECTIVES AND APPROACHES

Objectives

- Enabling broadcast-grade linear TV service on the Internet
- Making media service more personalized, interactive and immersive
- Enabling monetization of media services
- Making services accessible on many different devices and platforms
- Ensuring an end-to-end work flow with all enablers is in place

Approaches:

- Interoperability programs
- Identifying commercial Demand
- Global standards and ecosystems
- End-to-end workflows and ecosystems
- Supporting implementations by test, open source, conformance and reference tools



DASH-IF

CONTINUING TO SUPPORT DASH ADOPTION . . .

Founded in 2012 after MPEG-DASH completion, DASH-IF addresses

- Interoperability
- Promotion
- Supporting other SDOs and our members

for interoperable deployment of massively scalable Internet Streaming Services

TOOLS CREATED AROUND DASH-IF IOPS

DASH–IF creates guidelines referred to as Interoperability Points (IOPs)

In order to support the deployment of DASH-IF IOPs, DASH-IF Interoperability Group maintains tools:

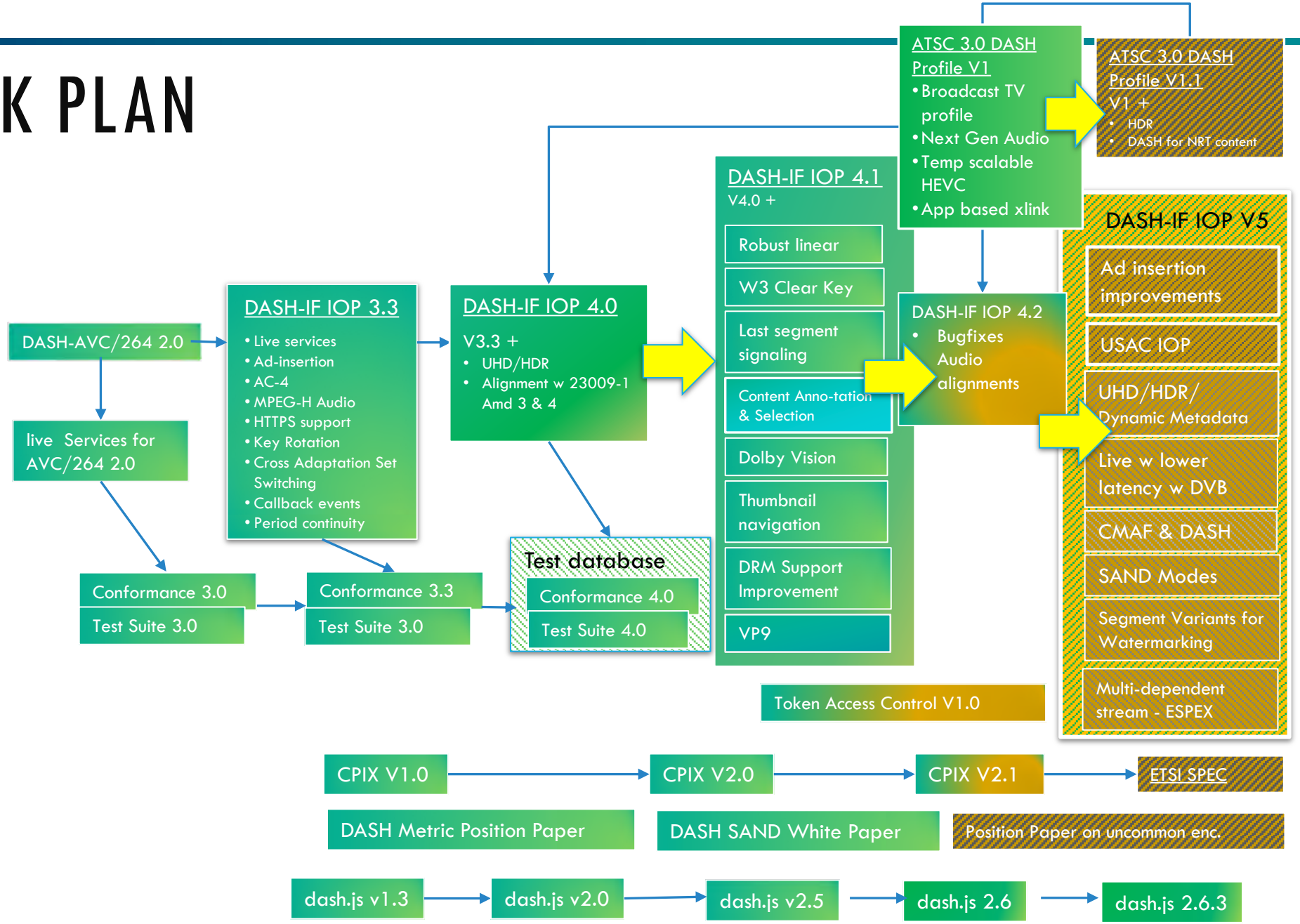
- Documentation of the Interoperability Points
  - V5.0 will be significantly revised both in terms of structuring and publishing
- Test Cases backing the Interoperability Points
- Test Vectors for Test Cases hosted on DASH-IF and Akamai servers
- Conformance Software hosted on DASH-IF web site
- Reference Player delivered as open source player ➔ dash.js
- Open Source Work Flow for Content Generation
- DASH Identifiers Repository

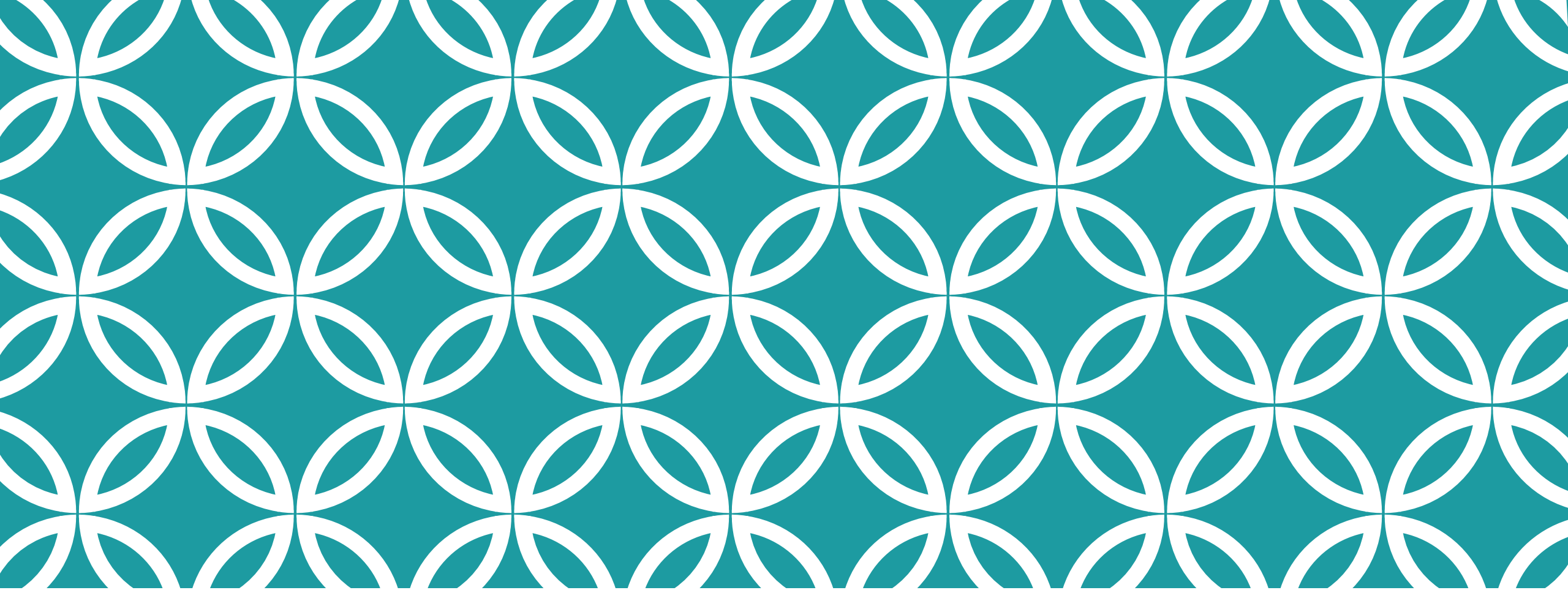
Recent activity: Feature-based development supported by a data base of features, test cases and test vectors

All documentation and tools are publicly available and no fees are attached.

For details refer to http://dashif.org ➔ Interoperability

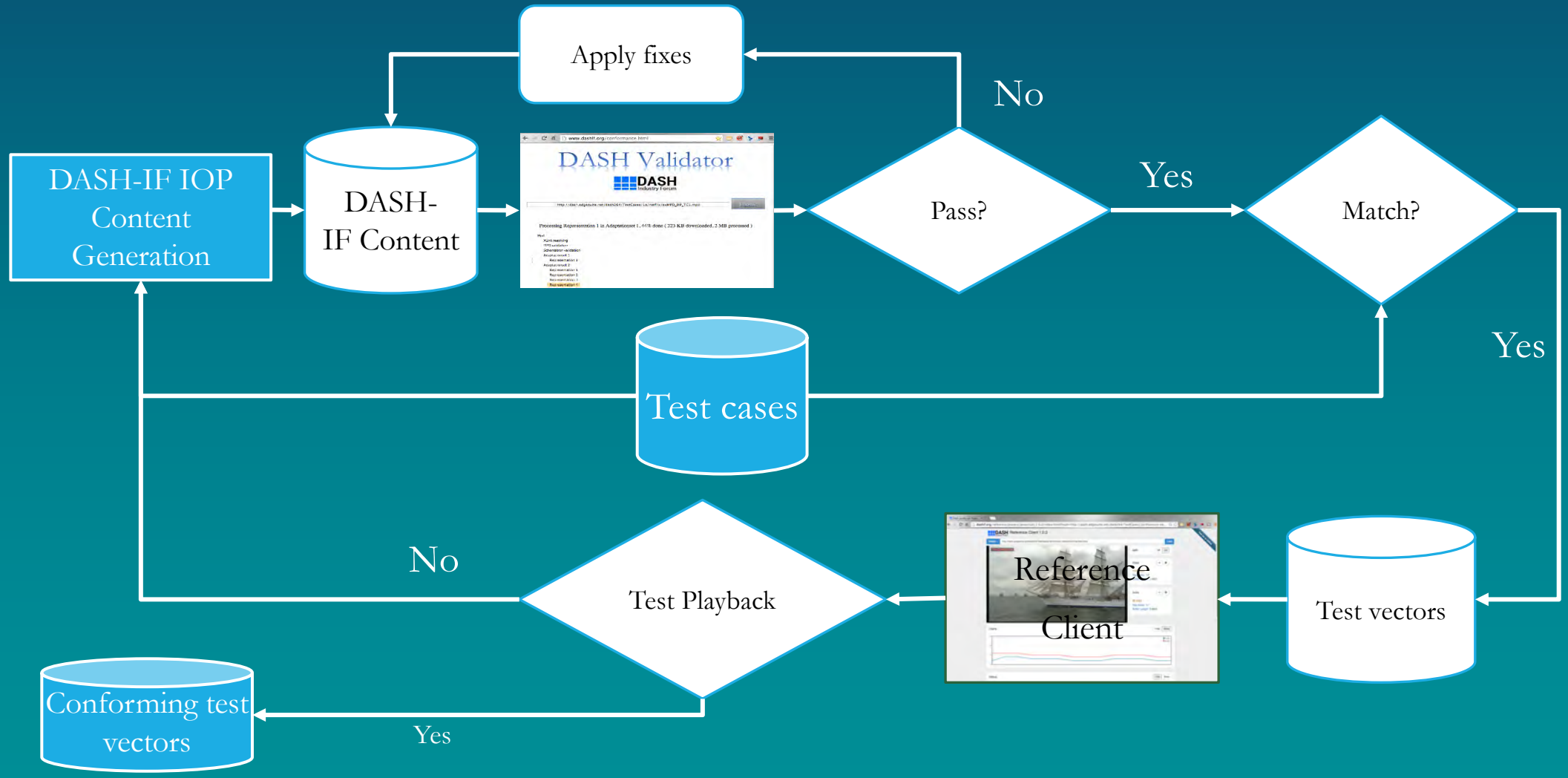
WORK PLAN


DASH-AVC/264 2.0
live Services for AVC/264 2.0
DASH-IF IOP 3.3
• Live services
• Ad-insertion
• AC-4
• MPEG-H Audio
• HTTPS support
• Key Rotation
• Cross Adaptation Set Switching
• Callback events
• Period continuity
DASH-IF IOP 4.0
V3.3 +
• UHD/HDR
• Alignment w 23009-1 Amd 3 & 4
ATSC 3.0 DASH Profile V1
• Broadcast TV profile
• Next Gen Audio
• Temp scalable HEVC
• App based xlink
ATSC 3.0 DASH Profile V1.1
V1 +
• HDR
• DASH for NRT content
DASH-IF IOP 4.1
V4.0 +
Robust linear
W3 Clear Key
Last segment signaling
Content Anno-tation & Selection
Dolby Vision
Thumbnail navigation
DRM Support Improvement
VP9
DASH-IF IOP 4.2
• Bugfixes
Audio alignments
DASH-IF IOP V5
Ad insertion improvements
USAC IOP
UHD/HDR/ Dynamic Metadata
Live w lower latency w DVB
CMAF & DASH
SAND Modes
Segment Variants for Watermarking
Multi-dependent stream - ESPEX
Conformance 3.0
Test Suite 3.0
Conformance 3.3
Test Suite 3.0
Test database
Conformance 4.0
Test Suite 4.0
Token Access Control V1.0
CPIX V1.0
CPIX V2.0
CPIX V2.1
ETSI SPEC
DASH Metric Position Paper
DASH SAND White Paper
Position Paper on uncommon enc.
dash.js v1.3
dash.js v2.0
dash.js v2.5
dash.js 2.6
dash.js 2.6.3



DASH-IF TOOLS AND PROCESSES

TEST VECTOR GENERATION HIGH-LEVEL FRAMEWORK AND DASH-IF ASSETS

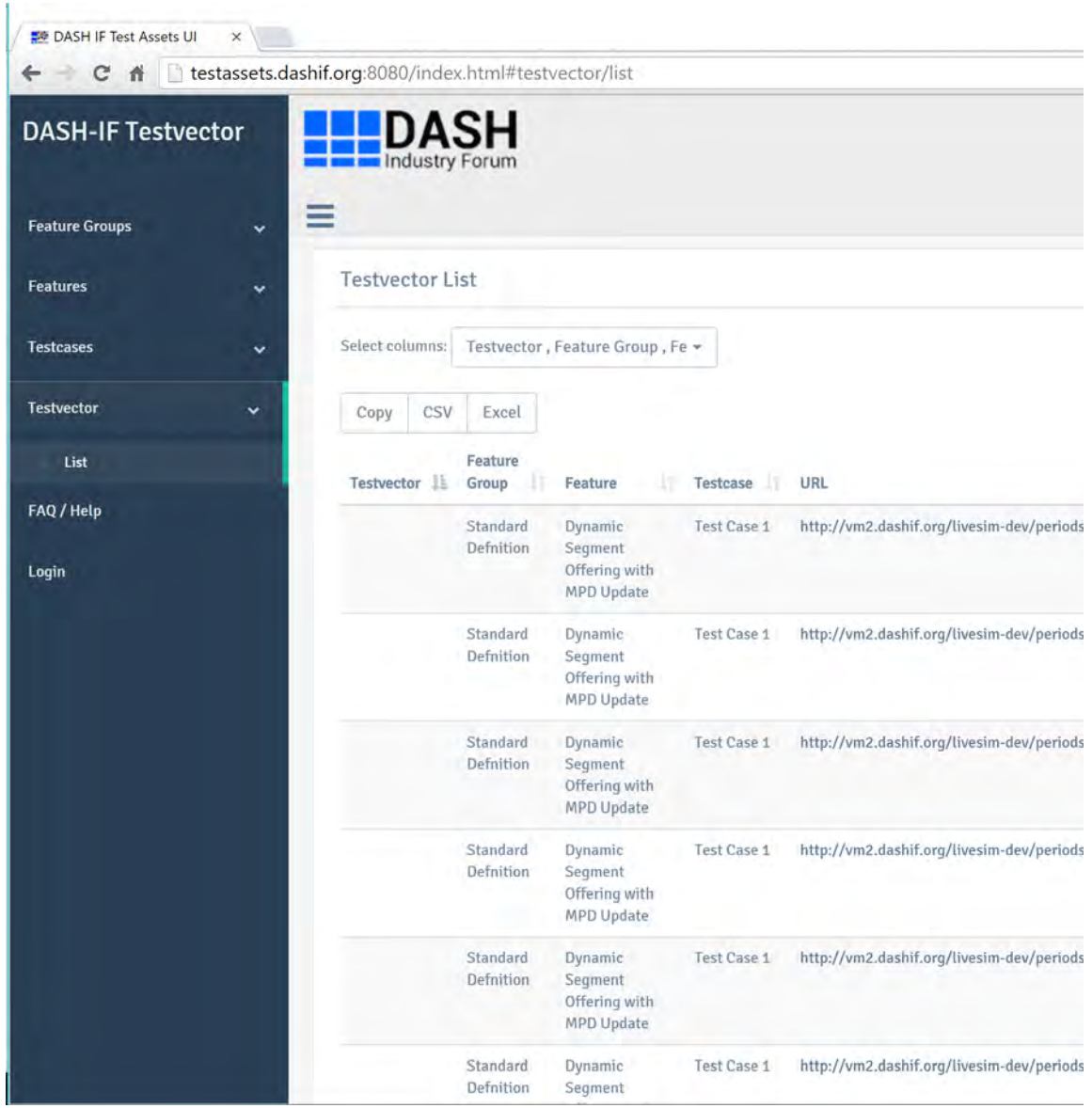


TO ADDRESS SCALABILITY: DASH-IF TEST ASSET DATA BASE

A database and UI hosted on an Amazon EC2 instance

Multiple levels of user authorization

- Public viewing of assets
- DASH-IF users for updating dataset
- Master-level user



CONFORMANCE

Validation of

- MPD
- Representations
- Interactions MPD/Representations

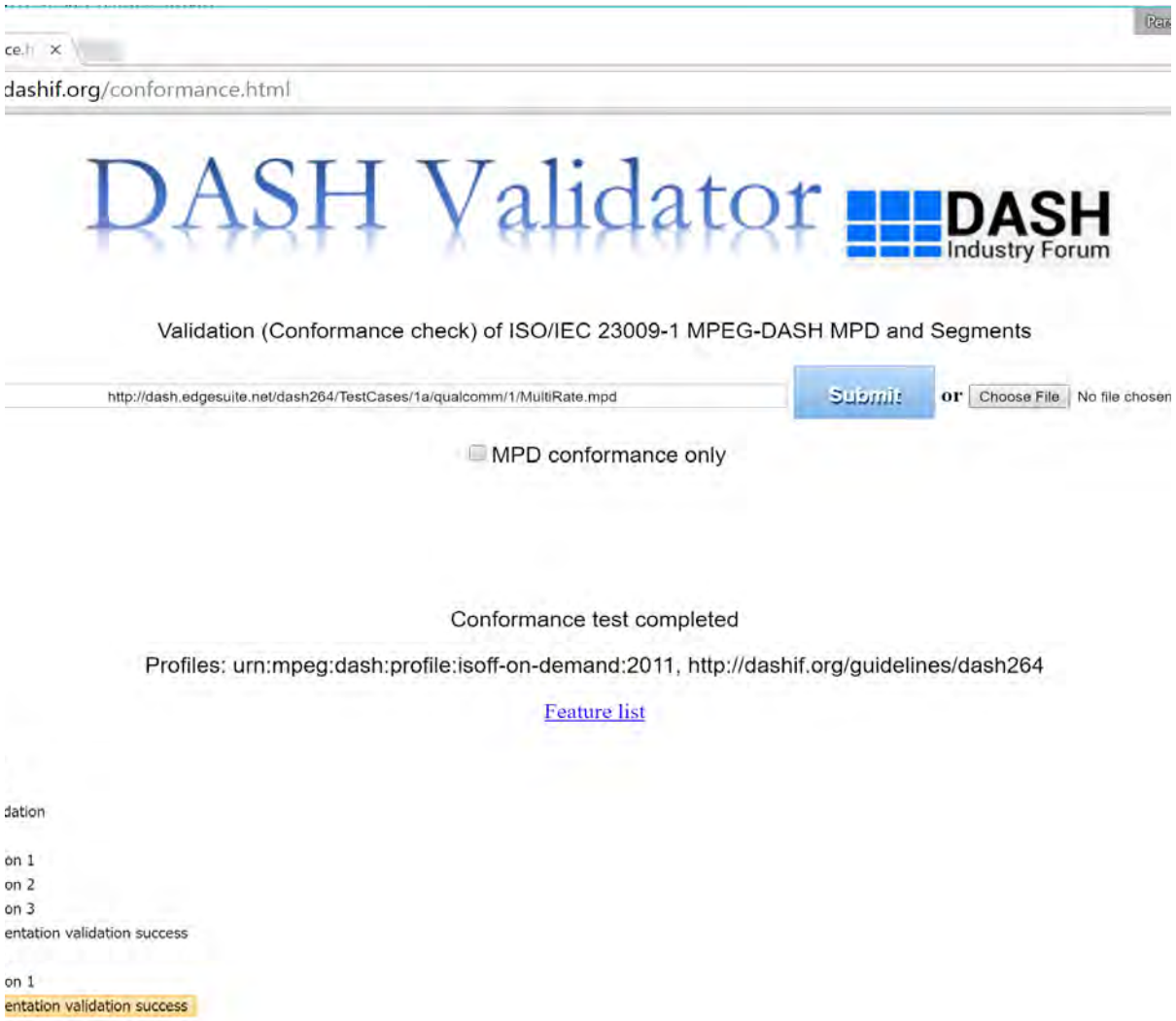
  According to ISO/IEC 23009-1 and DASH-IF IOP

  No elementary stream

Source Git

- Frontend: https://github.com/Dash-Industry-Forum/Conformance-Software
- Backend: https://github.com/Dash-Industry-Forum/Conformance-and-reference-source

https://dashif.org/conformance.html



WHAT IS THE REFERENCE CLIENT?

A player which can play back the DASH-IF IOP test vectors and which illustrates best practices in doing so.

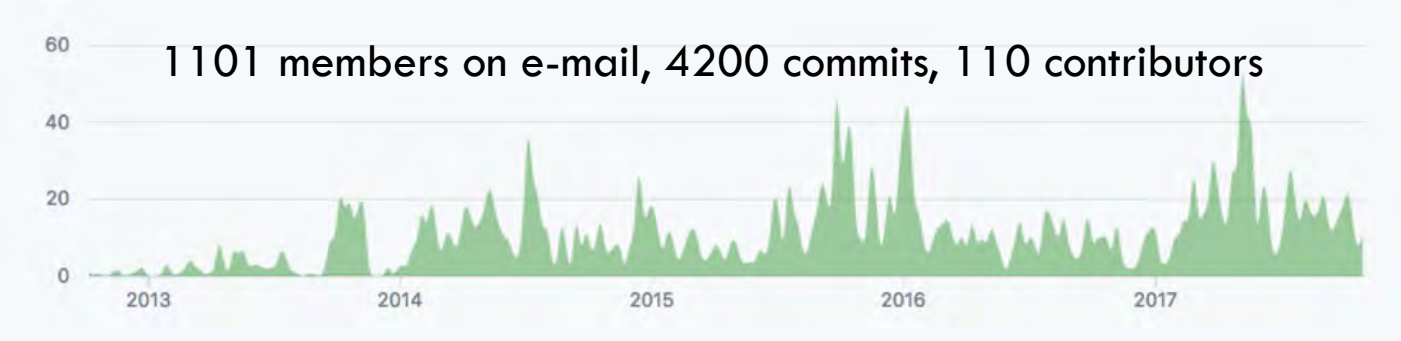
The client is written in JavaScript and makes use of the W3C Media Source Extensions and Encrypted Media Extensions to the VideoElement.

Current client works on Chrome v30+, IE11 under Win8.1, Edge under Win10, Safari under Yosemite, Firefox 39+.

It is an open-source project on Github - https://github.com/Dash-Industry-Forum/dash.js/

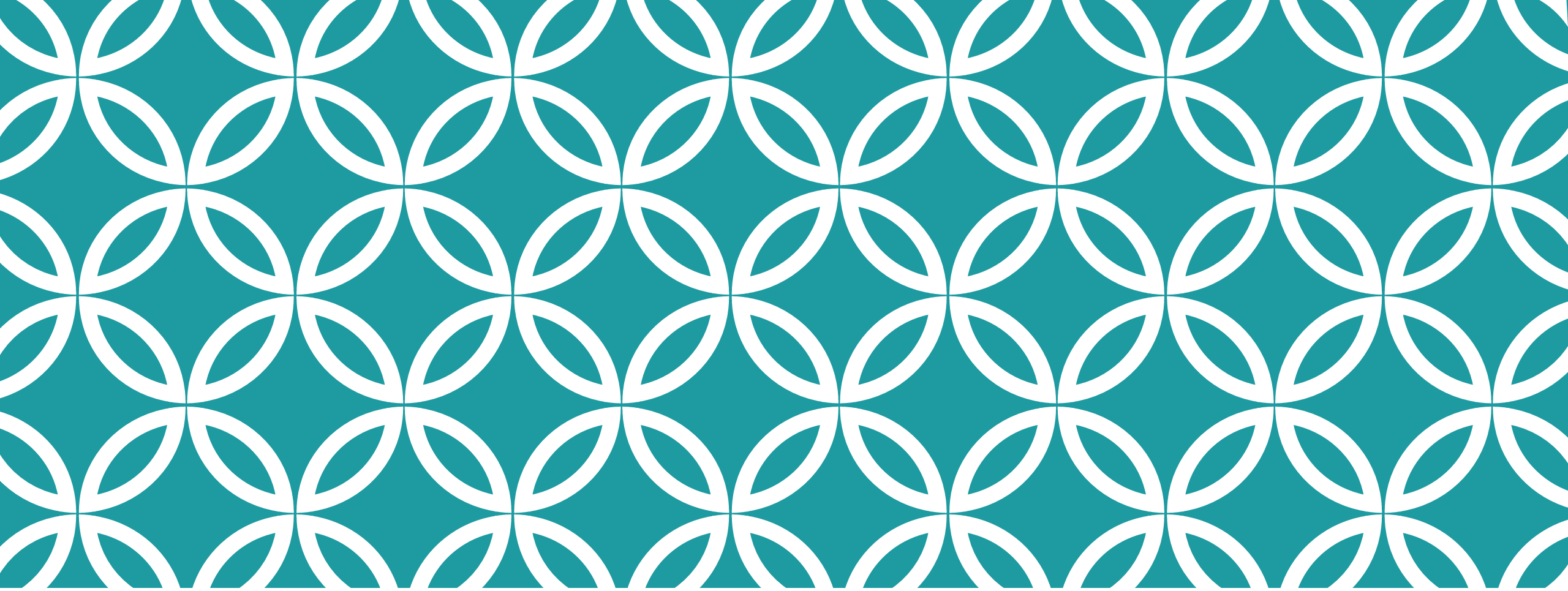
Project has been underway approximately four years.

There have been 80 different collaborators to the code to date.



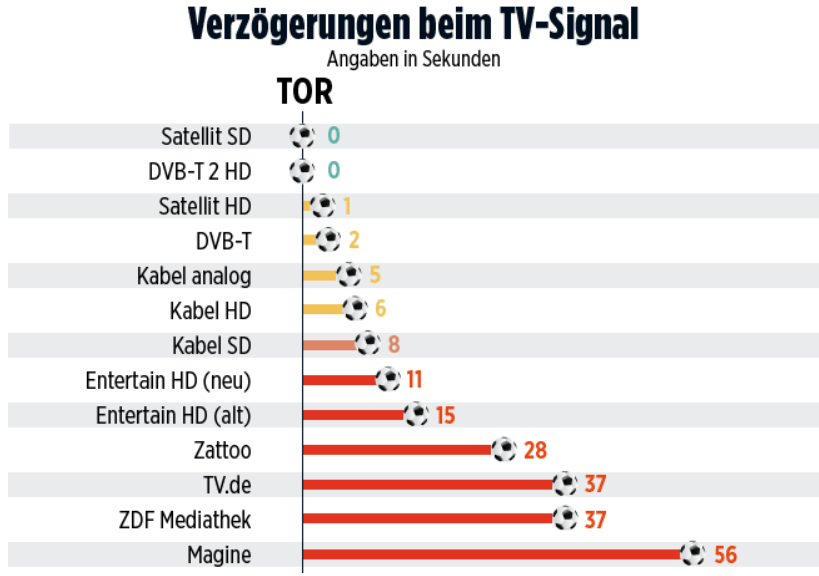
- Availability of open source tools to generate the test material
  - DASH-IF supports the generation of open source tools
- Availability of test content with proper licensing
  - We decided to exclusively work with that Creative Common v4.,
- Keeping the development of all test tools in sync
  - decided to work based on features that are gradually added
  - Commitment by proponents
- Hosting and size of data, reliability
  - Rely on commercial hosting services: Akamai, Amazon EC2
  - Specific software developed to minimize impact
- Handling of scalability/consistency of test material
  - Database was generated that can be used to host
- Coordination
  - Commitment of supporting company
  - Divide and conquer such that individuals can contribute
  - Issues are tracked on github throughout
- Test success
  - We only define qualitative criteria at this point in time
  - We only test for interoperability for now, not for performance

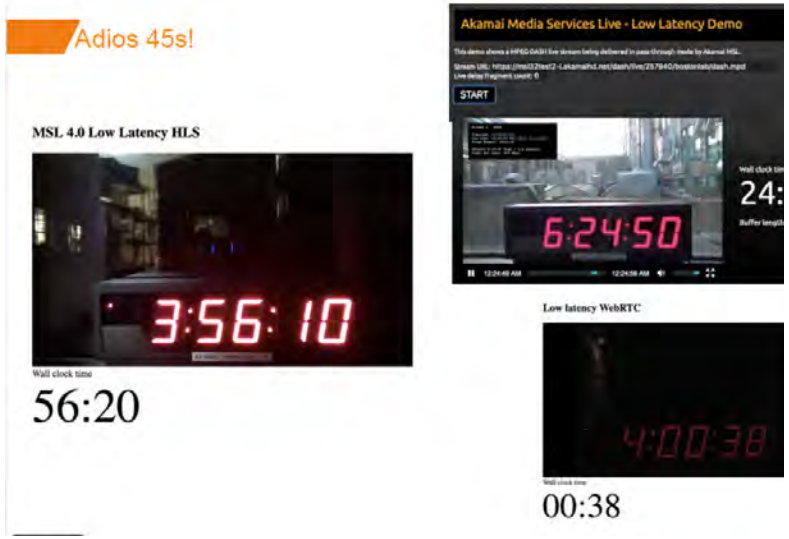
ADDRESSING COMPLEXITY



LOW-LATENCY DASH

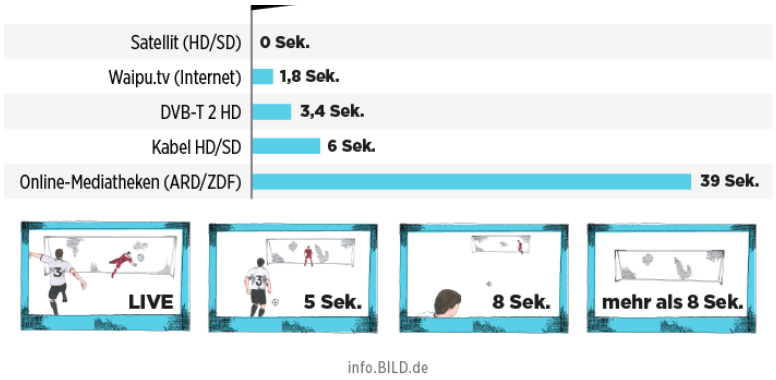
REMEMBER 2017 WORKSHOP





info.BILD.de | Quelle: heise online

AND THERE WAS THE WORLD CUP 2018



KEY PERFORMANCE INDICATORS

Latency:

- End-to-End Latency (EEL): The latency for an action that is captured by the camera until its visibility on the remote screen.
- Encoding+Distribution Latency (EDL): The latency of the linear playout output (which typically serves as input to distribution encoder(s)) to the screen

Startup Delay

- Live Edge Start-up Delay (LSD): The time between a user action (service access or service join) and the time until the first media sample of the service is perceived by the user when joining at the live edge
- Seek Start-up Delay (SSD): The time between a user action (service access or service join) and the time until the first media sample of the service is perceived by the user when seeking to a time shift buffer.

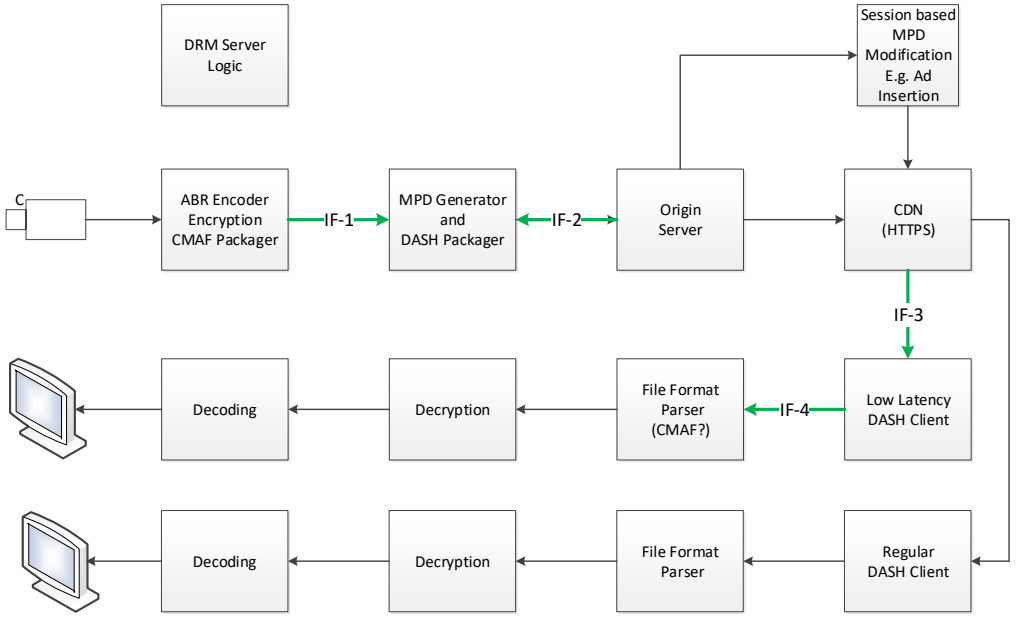
Adjusting the above parameters may impact certain other performance aspects as those document below.

- Compression Efficiency
- Network Efficiency and Scalability
- Robustness to Bandwidth Variations and Errors
- Lower layer protocol changes needed

STATUS OF THE WORK IN DASH-IF AND DVB

DVB

- Completion of use cases (together with DASH-IF) and Commercial Requirements for Low-Latency DASH
  - Encoder to Screen Latency of 3.5 seconds
  - Live Edge Start-up Delay in the order of 1 second or less
  - presentation of a media time at a specific wall-clock time within 500ms tolerance
  - updated DVB-DASH specification shall be completed by Q4/2018
- Technical work started in collaboration with DASH-IF



DASH-IF

- In the progress of drafting guidelines for Low-Latency DASH
- Context of real service operation issues: Program changes, ad insertion, operational problems, scalability
- Guidelines include
  - Interface between Encoder and DASH Packager assuming CMAF packaging
  - DASH Packager Operation including MPD generation and MPD updates, as well as segment generation
  - Client Implementation Guidelines and requirements: buffers, ABR logic, etc.
- Development of test, reference and conformance tools

SELECTED TECHNOLOGY OVERVIEW AND SOME ISSUES

HTTP Chunked Transfer Encoding of partially available files

- possible with CDNs today as long as they are configured for low latency streaming

Multiple Movie Fragments per Segment ➔ CMAF Chunking

- Supported in MSE
- However,
  - has not been tested on legacy clients in HbbTV
  - significant percentage of clients fail to play segments with multiple movie fragments
- What should the chunk duration be: 1 frame, 320ms? Experiments necessary
- What about sparse tracks such as subtitle? Clarifications/bug fixes needed in MPEG file format

Signaling early availability in MPD

- Supported by DASH MPD, but required some clarification ➔ Cor.3 in MPEG

Using @duration and $Number$ for now

- We identified an issue with Segment Timeline which requires the duration of the Segment to be known to announce

Accelerated playback in the device to address both low latency and fast startup

- Generally supported in MSE, but is it tested? Not supported in common TV Set decoder APIs ➔ CTA WAVE
- What are the implications of applying this to the user perception

Discussion on new HTTP variants, for example to support queued requests

LOW-LATENCY STREAMING

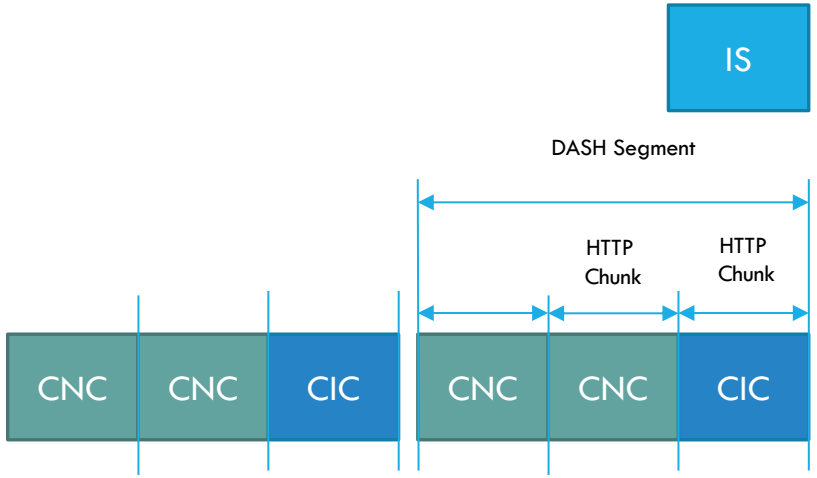
Encoder

CH = CMAF Header

CNC = CMAF non-initial chunk

CIC = CMAF initial chunk

DASH Packager



10s

Regular DASH Client

Segments

CDN caches Segments

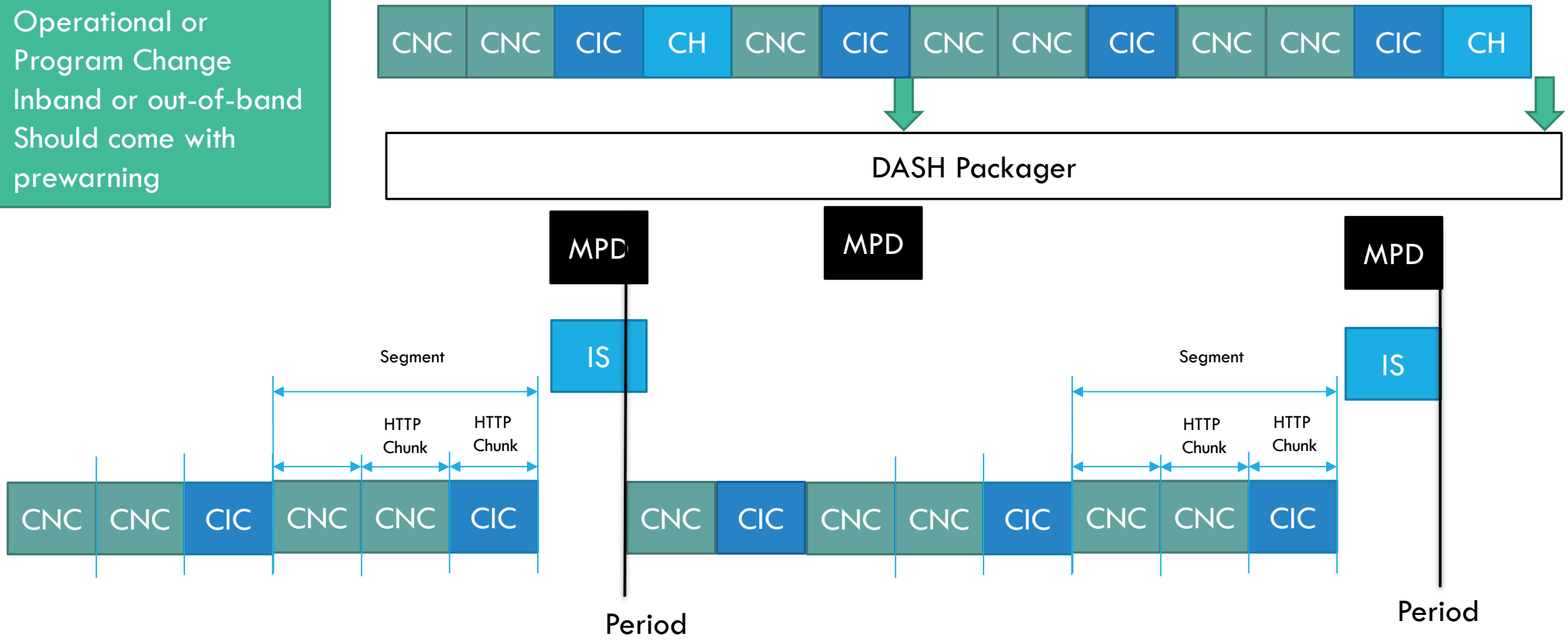
3s

Low-Latency DASH Client

Chunks

LOW-LATENCY STREAMING AND TRIGGERS

Triggers:
- Operational or Program Change
- Inband or out-of-band
- Should come with prewarning

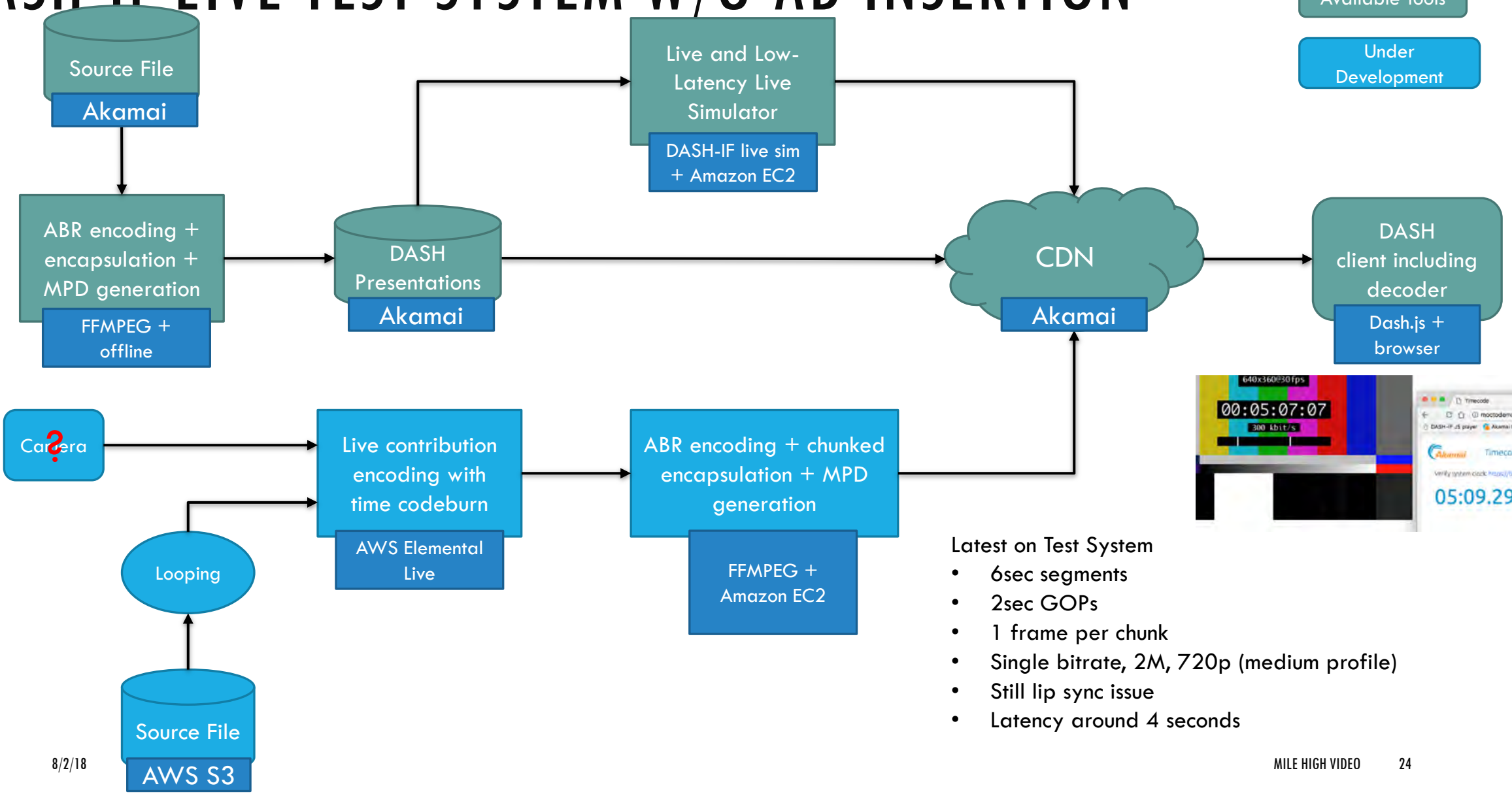


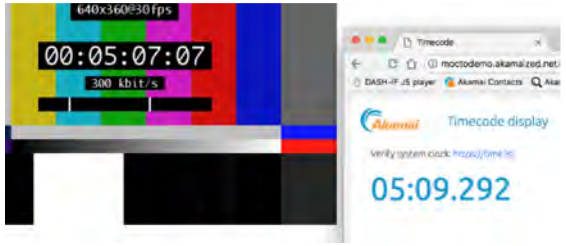
DASH-IF LIVE TEST SYSTEM W/O AD INSERTION

Tools and Hosting

Available Tools

Under Development

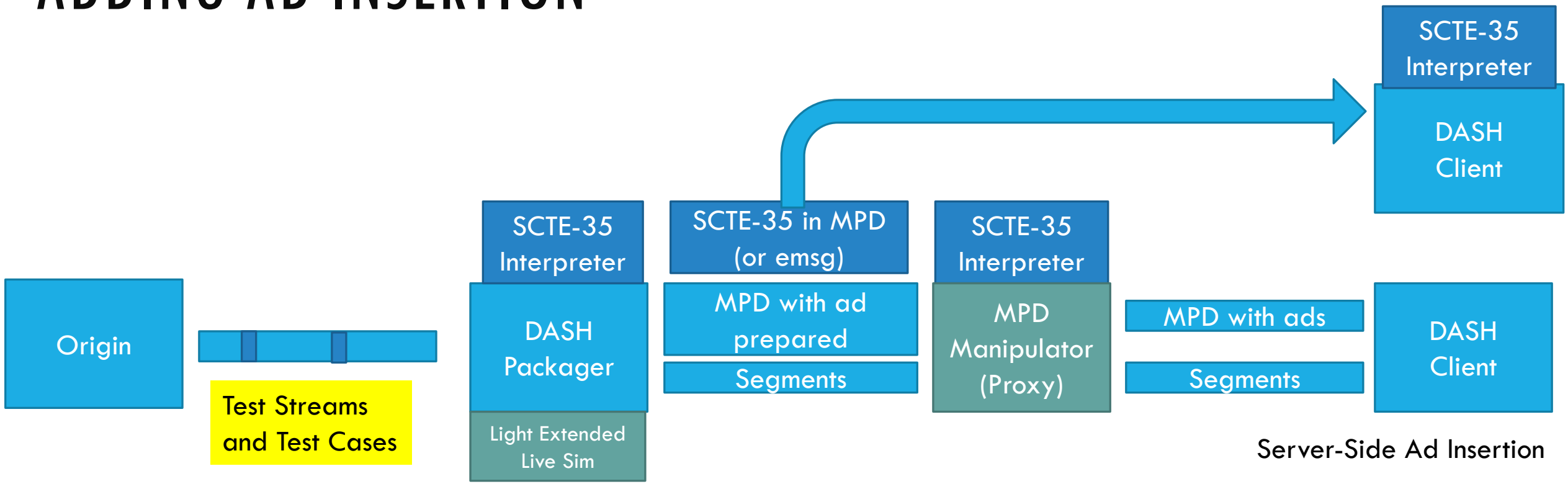




Latest on Test System

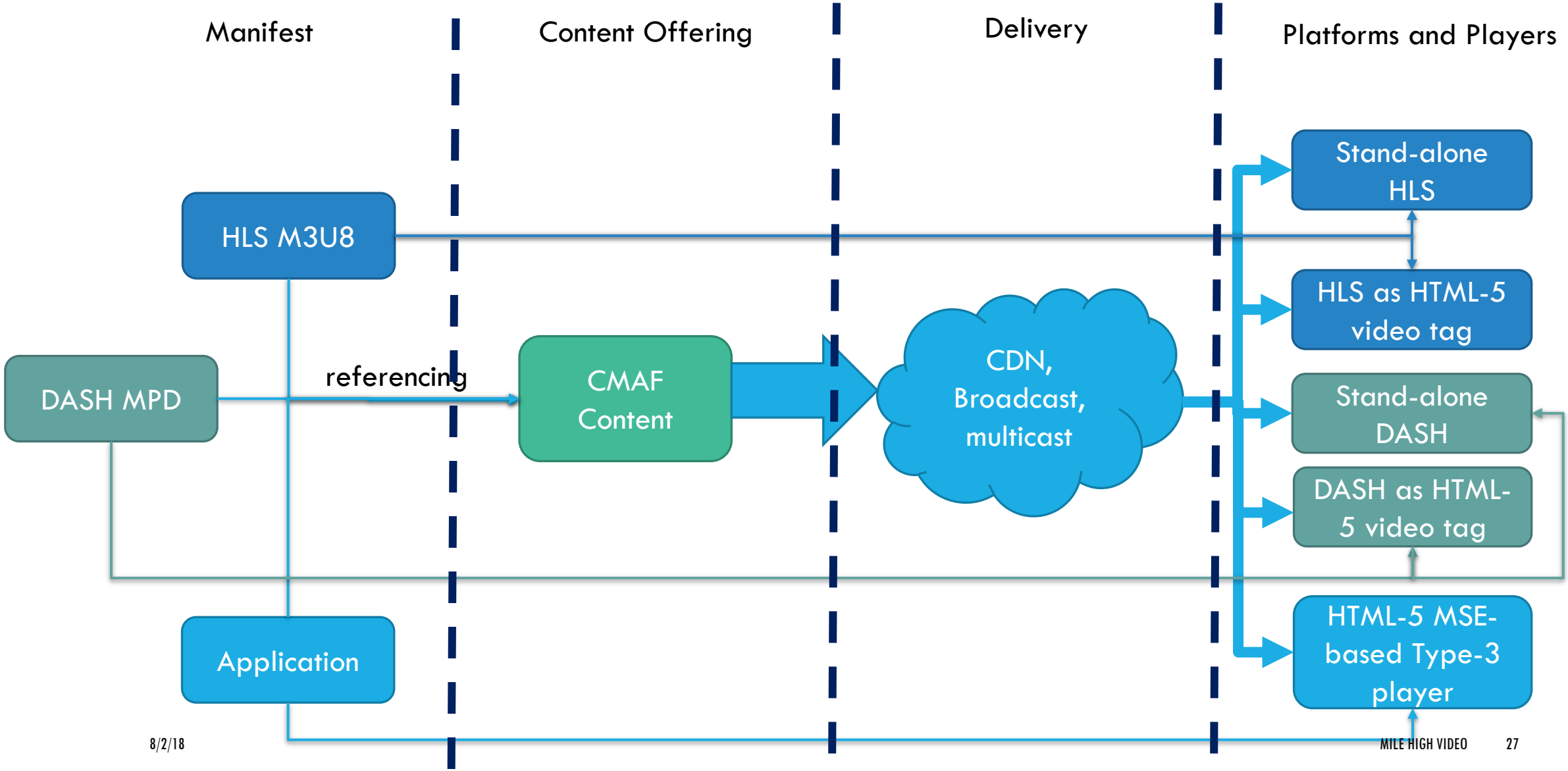
- 6sec segments
- 2sec GOPs
- 1 frame per chunk
- Single bitrate, 2M, 720p (medium profile)
- Still lip sync issue
- Latency around 4 seconds

ADDING AD INSERTION

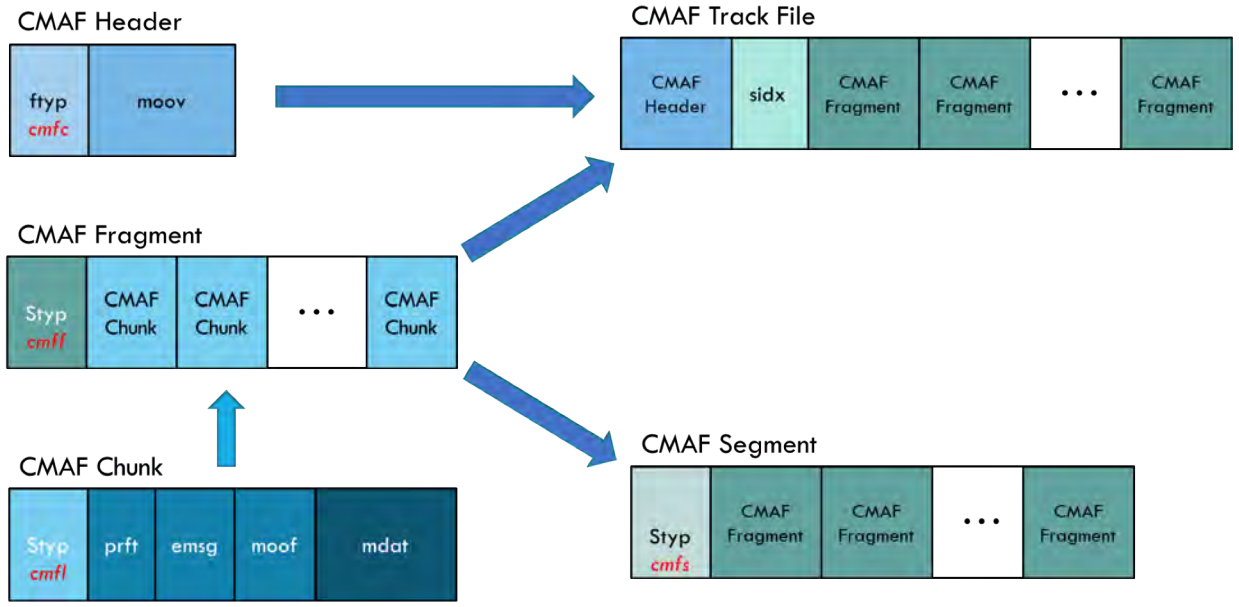


CMAF AND CTA WAVE

DIFFERENT PLAYERS – SINGLE ENCODING AND COMMON DELIVERY



DASH-IF AND CMAF



Gradually move segment formats to be CMAF conforming

However, we need to check very carefully on conformance between DASH and CMAF

Common conformance suite for content to identify issues

Work with the industry to fully align the two technologies

Update CMAF to add relevant DASH functionalities

- Multistream
- Period concept and splicing
- Events
- And and and

COMMERCIAL OTT VIDEO ISSUES: WAVE SOLUTION

Content Specification

Content Specification based upcoming ISO MPEG Common Media Application Format (CMAF), compatible with DASH and HLS.

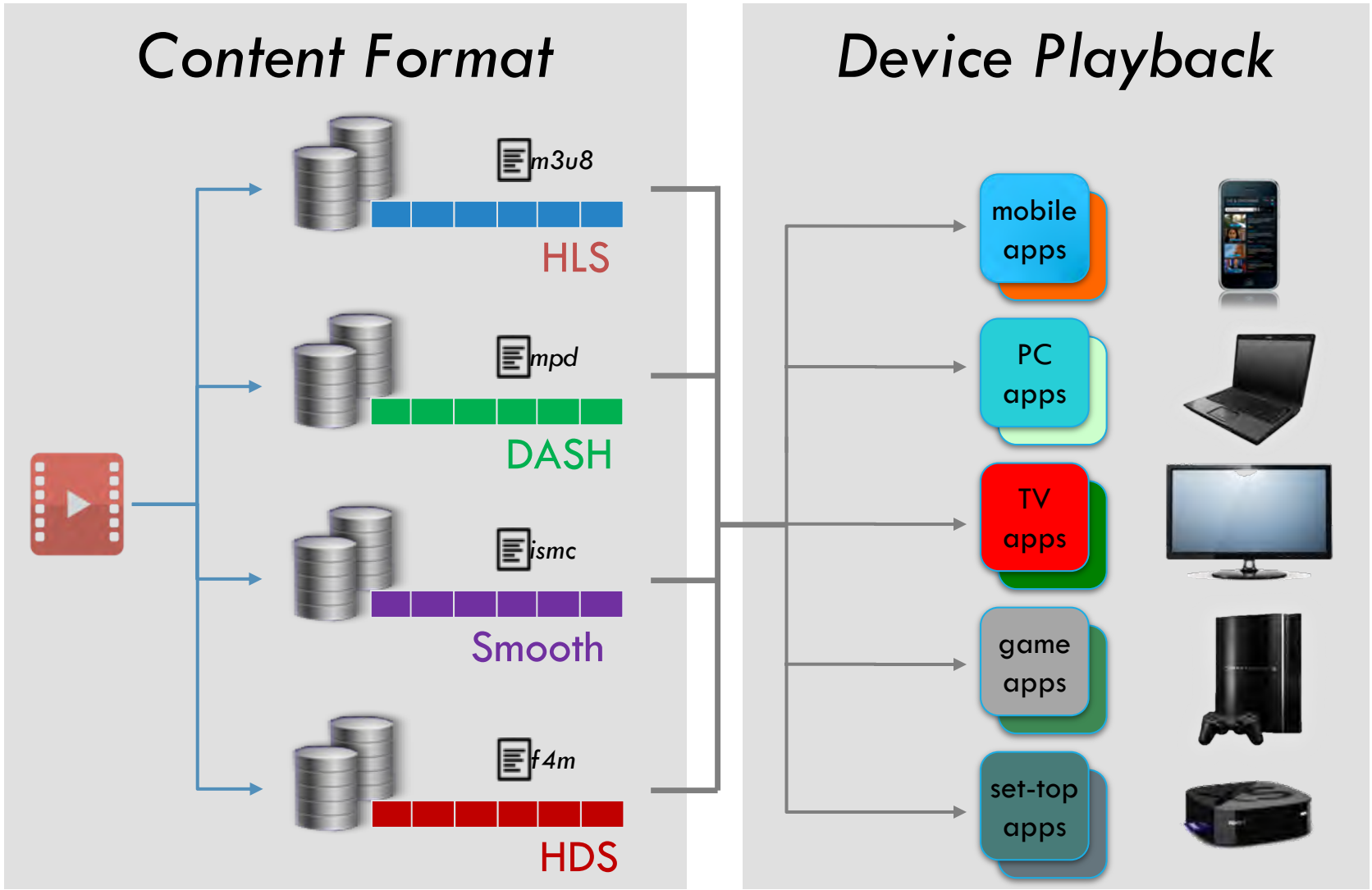
Device Playback Requirements

Testable requirements covering the most common device playback interoperability issues.

HTML5 Reference Platform

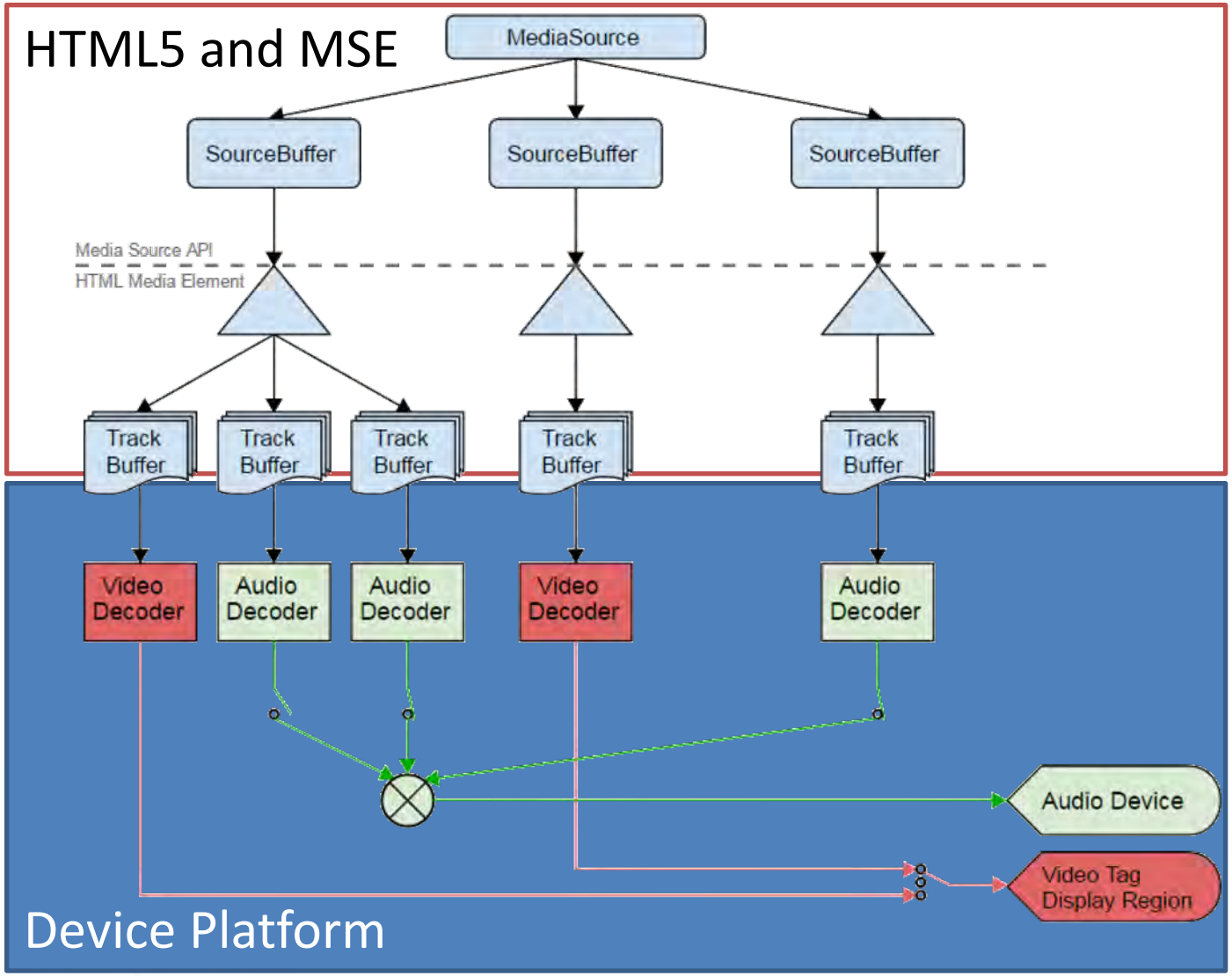
Reference application framework based on HTML5 providing functional guidelines for playback interoperability.

COMMERCIAL OTT VIDEO ISSUES: DEVICE PLAYBACK ISSUES



- Switching bitrate glitches
- Codec incompatibility
- Scaling display issues
- Partial profile support
- Long-term playback instability
- Audio discontinuities
- Request protocol deficiencies
- Memory problems
- CPU weakness
- Variable HDR support
- Unknown capabilities
- Ad splicing problems

Connection to HTML5 & MSE



HTML5 and MSE

- Provide APIs for applications to playback WAVE content
- Extend APIs to ensure more consistent and richer user experience

Device Playback Platform:

- Ensuring that WAVE content can be “played” consistently when using “MSE-like” APIs for different use cases and applications.
- Use HTML5 as reference and test platform, not excluding other platforms

DEVICE PLAYBACK

One of the key missing pieces for consistent Internet TV Services

Media Source Extension

- This specification extends HTMLMediaElement [HTML51] to allow JavaScript to generate media streams for playback.
- Allowing JavaScript to generate streams facilitates a variety of use cases like adaptive streaming and time shifting live streams.

ByteStream Format for ISO BMFF

- https://www.w3.org/TR/mse-byte-stream-format-isobmff/
- This specification defines a Media Source Extensions™ [MEDIA-SOURCE] byte stream format specification based on the ISO Base Media File Format.

For MSE being used, the results of using the API need to fulfill highest-quality requirements

```
var ms = new MediaSource();
video.src = window.URL.createObjectURL(ms);
ms.addEventListener('sourceopen', onMediaSourceOpen);

function onMediaSourceOpen() {
  sourceBuffer = ms.addSourceBuffer('video/mp4; codecs="avc1.4d401f"');
  sourceBuffer.addEventListener('updateend', nextSegment);

  GET(initUrl, appendToBuffer);

  video.play();
}

function nextSegment() {
  var url = templateUrl.replace('$Number$', index);
  GET(url, appendToBuffer);
  index++;
  if (index > numberOfChunks) {
    sourceBuffer.removeEventListener('updateend', nextSegment);
  }
}

function appendToBuffer(videoChunk) {
  if (videoChunk) {
    sourceBuffer.appendBuffer(new Uint8Array(videoChunk));
  }
}
```

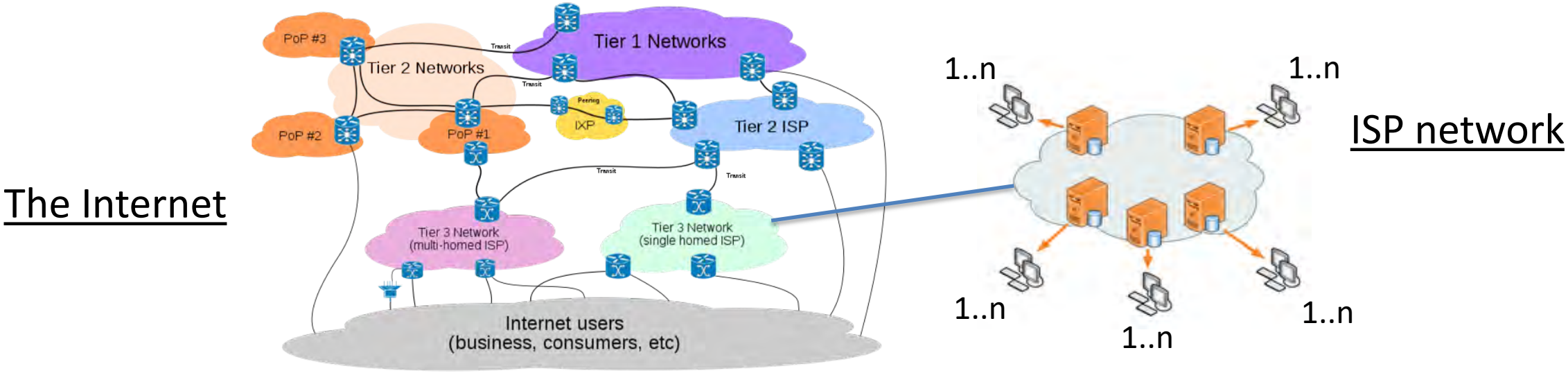
Expected WAVE Specification

Task Force	Title of Output	Expected date of Publication
HATF	Web Media API Snapshot 2017	NAB 2018
CSTF	Content Specification	NAB 2018
DPCTF	Device Playback Specification	Q2 2018
TCTF	WAVE Overall Test Approach	Q2 2018
TCTF	WAVE Test List	Q2 2018
TCTF	WAVE Test Specification	Q2 2018
TCTF	WAVE Test Suite	End 2018
TCTF	WAVE Test Tools	End 2018
HATF	Web Media Application Developer Guidelines 2017	Q2 2018
HATF	Web Media User Agent Integration Specification 2017	TBD

DVB INTERNET SERVICES (DVB-I)

DVB-I, the mission…

- DVB-I, where the "I" stands for "Internet"
  - In the context of audio-visual services, "The Internet" is used for "Over-The-Top" (OTT) delivery
  - Well, "The Internet", as in "CDN overlaid, edge assisted, adaptive delivery, media cloud"



- …To enable DVB services to be discovered and consumed by devices with basic Internet connectivity, principally a non-managed broadband connection and HTTP access, providing a similar user proposition to that of a DVB broadcast service

DVB-I – why do we need it all?

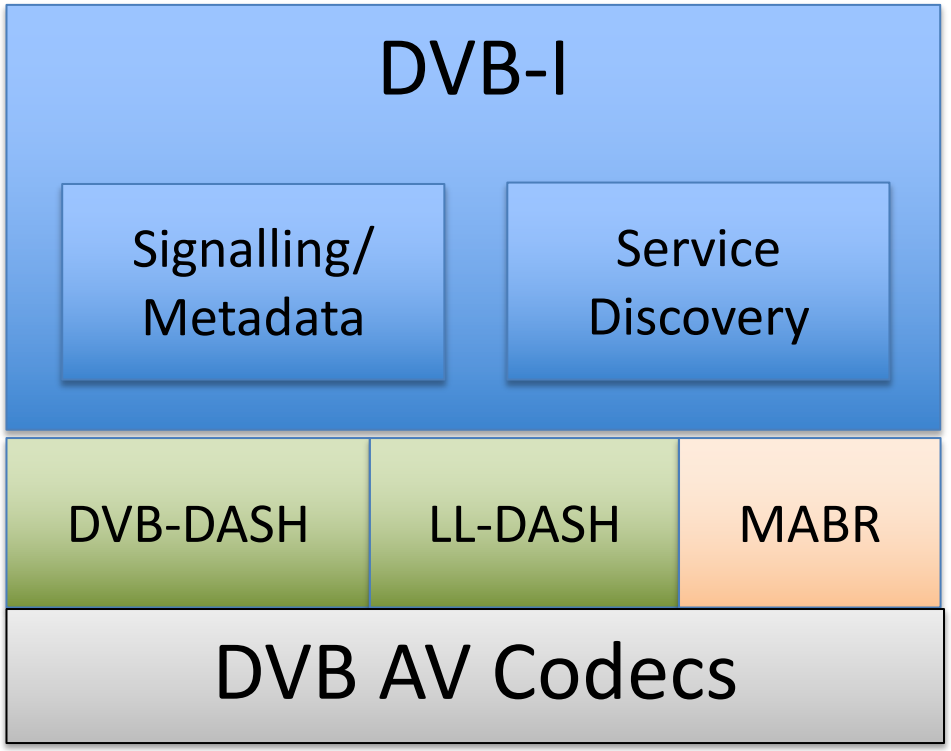
A complex ecosystem of platforms, protocol stacks (including ABR), codecs, DRMs, apps, browsers and the devices able to receive and consume video content has taken shape…

- This is probably a bad thing (or at least the down-side of the "good thing")
  - → *Unless You have managed to cut Yourself a nice piece of the cake!*

- → Or is this "bad" at all? Maybe it's just the modern-day media paradigm, the way it was meant to be?
- The "app" ecosystem - a vertical service offered on receiver device platforms where it's worth providing support
  - → Problem for the consumer – inconvenience of switching between apps; obsolescence of app on otherwise perfectly functional receivers
  - → Problem for the CE manufacturer – support of multiple solutions for the same function – codecs, ABR solutions, DRM systems, user interface constructs
  - → Problem for the broadcaster – how to reach the audience efficiently and reliably?

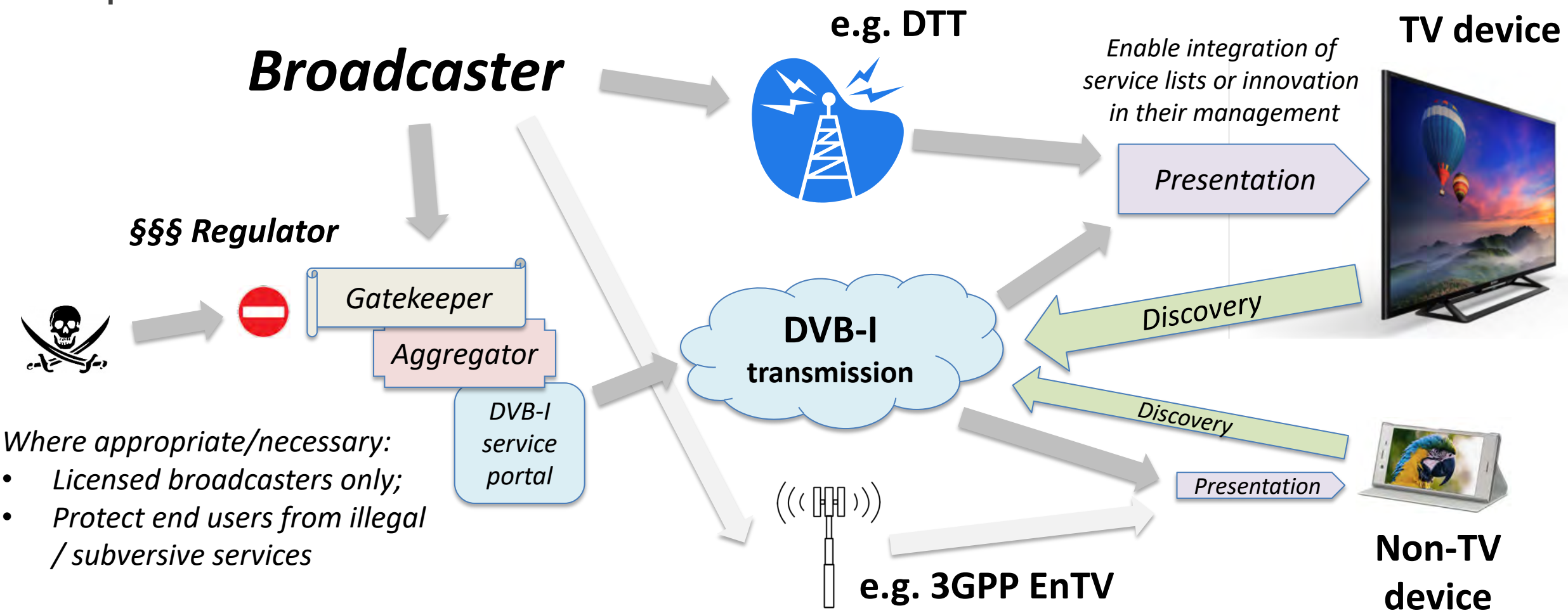
DVB-I, the vision

- ***Harnessing foundation technologies to provide a complete DVB solution for live OTT delivery:***
  - DVB-DASH (ABR – adaptive bit-rate)
    - ETSI TS 103 285
  - Low-latency DASH (LL-DASH)
    - Technical work started
  - Multicast ABR (MABR) - within suitably capable operator networks
    - Technical work ongoing
    - Reference Architecture published
      - DVB blue book A176
- ***Potential synergies with other ongoing DVB work items:***
  - Targeted Advertising
  - Home Broadcast
- ***Potential liaison activities:***



DVB-I, the vision

- Functional overview; likely roles and elements of the DVB-I specification



Status of DVB-I

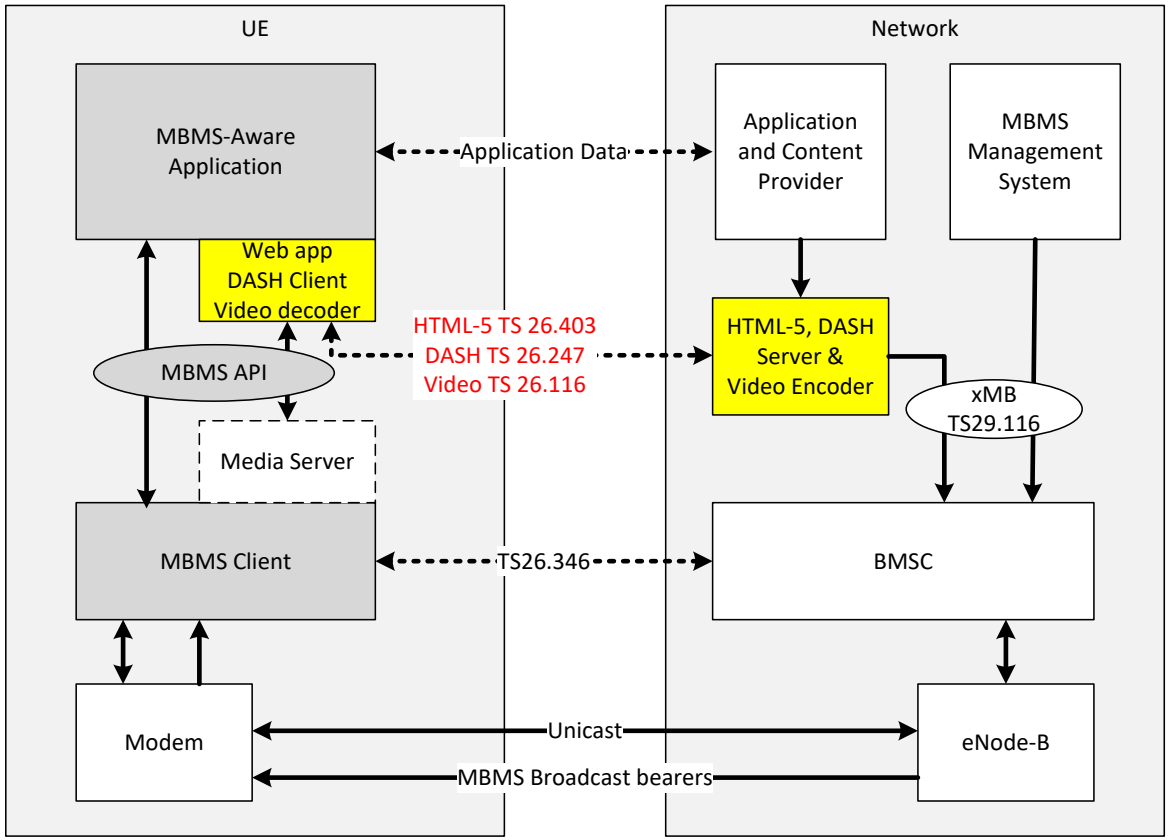
- Commercial Requirements completed, approval by mid August
- Main themes of the into more than 50 CRs
  - Applicability to TVs (w/ & w/o app) and non-TV devices incl. mobile & browser
  - Over-the-top possible, also optimization/management
  - Relying on DVB-DASH for delivery, likely LL-DASH once ready
  - Key concept are Service Lists including Service information, which are semi-static and provide some equivalence to DVB-SI
  - User experience equivalent to DVB-S/T/C/IPTV
  - Services can be 24x7x365 and can be part-time
  - Services can be a mix of live events and VoD Assets, and personalized
  - Hybrid services and devices are considered
  - Trust, security and privacy aspects are considered
  - Expected to have a receiver profile for a minimum-to-implement features for FTA services
  - And many more …
- Considered a starting point to replicate broadcast experience
- Technical work expected to start in fall 2018

3GPP ENTV, 5G AND HYBRID SERVICES

STATUS

Rel-14 enTV is completed with key features:

- Transport-only service
- Receive only services
- Standardized xMB interface
- MBMS-APIs
- Radio enhancements

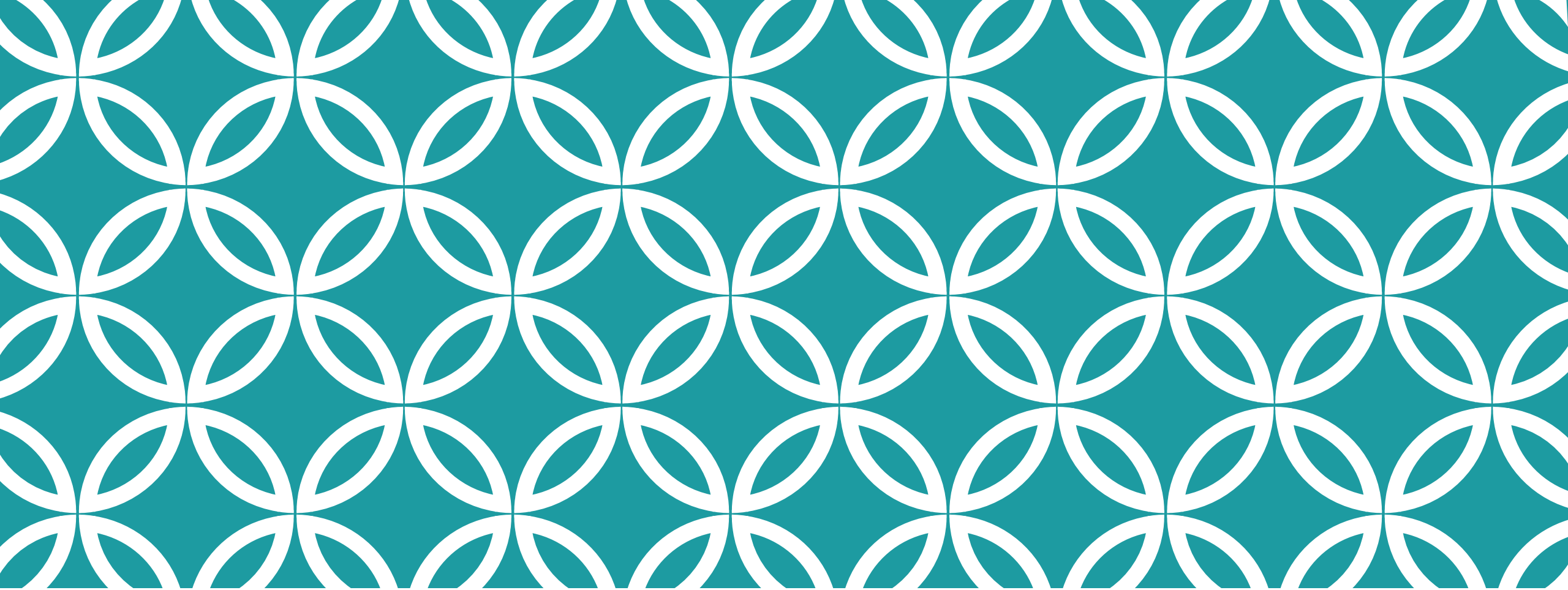


5G

- Rel-15: Increase throughput and reduce latency
- Broadcast is expected to be addressed only at later stage
- enTV is the baseline for standalone system
- Radio optimization combining broadcast and unicast

Hybrid Services

- Enhancing vertical broadcast services by interactive unicast
- Broadcast is a transport optimization by offloading popular content



SUMMARY

SUMMARY

Moving TV Services to the Internet is a major opportunity, but we consistency for reach and quality

Standards remain relevant for the Internet world, but different approaches necessary

No longer vertical services, but individual enablers that are well tested

APIs, testing, reference implementations, modular designs

JOIN THE EFFORTS