DASH-IF/DVB Report on Low-Latency Live Service with DASH

1 Scope

This report provides an overview when using DASH for Low-Latency Streaming. The document is primarily commercial and serves as a guidelines for future technical work in the context of DASH.

2 History

Version	Additions	Date
0.0.1	Initial Version after first Call on April 13	April 14th, 2017
0.0.2	Version before Call on May 19th	May 18th, 2017
0.0.3	Version before Call on May 26th	May 26th, 2017
0.1.0	Version after Call on May 26th	May 29th, 2017
0.2.0	Version after Call on June 2nd	June 2nd, 2017
0.3.0	Version for CM-AVC (including comments from Ali)	June 21st, 2017
0.3.1	Version before Call June 23rd	June 23rd, 2017
0.4.0	Version after Call June 23rd	June 23rd, 2017
0.4.1	Version before Call July 28th	July 28th, 2017
0.5.0	Version after Call July 28th (separate DVB and DASH-IF)	July 28th, 2017

Legend:

- Agreed (applies to full sections)
- Under discussion
- New
- Open issues/Actions

3 Table of Contents

4 Introduction

According to the FierceCable's paper titled "The Future of Internet Video" published in December 2016, consumers are watching more linear TV online than ever, and they increasingly expect a TV-like experience, even off the TV set. However, live programs such as sports require lower latency to deliver a service that's on par with broadcast or pay TV, while still providing all the benefits of an OTT distribution and not jeopardizing the viewing experience. Latencies known from proprietary system such as HLS up to 30-seconds or more can make or break a viewing experience. But with DASH there is light in the tunnel. According James Field from Cisco (cited in the above paper), "DASH is certainly starting to see adoption and that will certainly help reduce some of the latency problems. It's not going to eliminate that, but it's going to make it better."
This report is in nature commercially driven and provides some use cases for live distribution of TV services addressing relevant performance indicators for a successful service. It also attempts to include deployment experience, ongoing standardization efforts and industry practices. The document proposes a way forward for DVB on addressing the issue of Live OTT distribution.

5 Use Cases

5.1 Use Case 1: On Par with Other Distribution Means

A live event is distributed over DASH as well over regular TV distribution. The event should play-out approximately at the same time on both devices in order to avoid different perceptions of the same service when received over different distribution means. The objective should be to get to a range of delay for the DASH based service that is equivalent to cable and IPTV services.

5.2 Use Case 2: ABR Competing with Social Feeds

A live event is distributed over DASH. At the same time, the event is commented on Twitter or other social media that distribute information about the live event. In yet another feed, users have installed a push notification system for important events such as goals in a soccer game. The delay of the TV signal should not be such that notifications are received before the event is observed in the live media over ABR. For example, by personal observation the notification from OneFootball needs to be switched off while watching SkyGo as the goal notifications are received prior to the event seen on screen.

There are three variants:

A. The social feed origins from the venue by viewers
B. The social feed origins from watching other distribution means by viewers
C. The social feed origins from the broadcaster/content provider

In some studies (https://arxiv.org/ftp/arxiv/papers/1106/1106.4300.pdf, clause 5), the delay for Superbowl event on twitter is a minimum 13 seconds, but the analysis assumes that the notification is from users watching on TV (and we can assume a TV delay of 7-12 seconds in the US). This means that the minimum Twitter delay can be assumed in a ballpark range of 5 seconds between the event and the notification being available.

5.3 Use Case 3: Companion Streams and Screens

To deliver content over the Internet with a comparable overall system delay to that of a DVB broadcast system (inc. encoding, multiplexing, distribution, modulation, transmission, reception, demodulation, demultiplexing, decoding and display) for the purposes of

1. near-seamless switching between broadcast and IP-delivered content for the insertion of regionalised or personalised live content into a live broadcast
2. delivering alternative live streams over the Internet for simultaneous presentation with broadcast TV content:
    a. on the same device
    b. on a companion device

For these use cases, it is recognised that the low latency requirement implied by this is likely to require a high performance internet connection and that not all viewers may have suitable connections to achieve this today.

5.4 Use Case 4: Sports Bar

Sports bars are commonly in close proximity to each other and may all show the same live sporting event. Some bars may be using a provider which distributes the content using DVB-T or DVB-S services whilst others may be using DASH ABR. Viewers in a bar with a high latency will have their viewing spoiled as they will hear cheers for the goal before it occurs on their local screen. This creates a commercial incentive for the bar operator to switch to the provider with the lowest latency. The objective should be to get the latency range to not be perceptibly different to that of a DVB broadcast solution for those users who have a sufficient quality (high and consistent speed) connection.

5.5 Use Case 5: Variably Configured Latency across Channels

Service provider with varied services such true-live channels (e.g. Sports or News) and pre-recorded (e.g, Drama, movie) channels.
A service provider operating a number of different channels may have different priorities for each of the channels. For a sports channel their number one priority for users on a high quality connection may be to minimize latency, at the expense of channel start-up time or an increased risk of rebuffering. For a news channel fast start up and reasonably low latency may be prioritized equally. For a drama or movie channel guaranteeing a high QOE and therefore minimizing the risk of rebuffering (whilst still maintaining reasonable start-up time) may be the priority. It should therefore be possible to signal advice on how the service should be managed by the client.
As an extension, even within one channel, certain events may have different latency requirements. The service provider may offer the channel with the lowest required latency, but optimizations per event may be considered interesting for improved stability or other reasons.

5.6 Use Case 6: DASH in ABR Multicast

In an extension to the above use cases, a service provider wants to use an existing multicast infrastructure to distribute DASH content to edge devices and gateways. At the gateway, the service may be translated into a unicast services for distribution over the home network.

5.7 Use Case 7: DASH as broadcast format

In an extension to the above use cases 1-5, a service provider wants to use an existing broadcast infrastructure (e.g. ATSC3.0 or LTE Broadcast) to distribute DASH content to edge devices and gateways. At the gateway, the service may be translated into a unicast services for distribution over the home network.

5.8 Use Case 8: Enterprise Broadcast

An enterprise wants to distribute a live event world wide. No other distribution channel is available. The service provider wants to achieve an end-to-end latency with the service provider targeting the end-to-end latency value of 5 seconds at the minimum as a target, but possibly more (e.g. Apple developer conference).

In a variant, a feedback channel is available and the latency for the response should be perceptually not annoying, possibly lower than 5 seconds.

5.9 Use Case 9: Hybrid Broadcast

In a similar manner as documented in use case 3, a service provider may want to offer a primary program over broadcast, but augment the service with personalized content that is provided over unicast. The content may for example be personalized or targeted ads, secondary languages, alternative views or even fallback in case broadcast gets unavailable. Any transition between broadcast and unicast needs to be seamless and any joint presentation needs to be synchronized.

5.10 Use Case 10: Personal Broadcast

Non-professional users would like to broadcast themselves to their social networks. A current commercial example would be Facebook Live. This use-case has the following characteristics:

1. End-to-end latency are typically expected to be low
2. There is typically one or a few viewers per stream, although occasionally for viral broadcasts the viewership can spike to tens or hundreds of thousands of viewers.
3. Broadcasts are typically generated using a mobile device as the encoder.
4. The number of ingest streams can vary dramatically with flash ingest crowds being quite common.
5. Publishing and subscribing apps are typically coupled to one another.
6. Stream is typically single bitrate and typically no transcoding is applied, but repackaging may be done.

5.11 Use Case 11: Channel Bouquet

In this case, one service provider offers multiple channels that can be switched across, using typically channel switching user interfaces such as a remote control. The channel switch is expected to be comparable to existing TV services without impacting delay and latency requirements.

5.12 Use Case 12: Channel Bouquet from one Event

In an extension to use case 11, the channels may origin from the same event, for example different views of the same event, different camera perspectives, on-board cameras. The orchestration of the event may be done by the user. The event streams should be synchronized.

6 Service Scenarios

6.1 Benchmark DASH Service Configuration

As a typical DASH service configuration is reported:

- Segment duration: 2 - 10 seconds
- Each Segment starts with a SAP=1 or 2, i.e. with an IDR frame for video
- Audio and Video in separate Adaptation Sets
- Player buffer: 3 times maximum/target segment duration
- Players ignore suggested presentation delay or minimum buffer time

6.2 KPIs for live Services

In order to properly define service requirements Key Performance Indicators (KPIs) for Live Services are collected and defined. Note that not all KPIs are relevant for all services.

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
- Distribution-only Latency (DOL): The latency after the output of the distribution encoder to the screen
- CDN latency: The delay caused by the CDN delivery from CDN input to CDN output.

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.

Those two categories are subject to be controllable by the service provider for a consistent service offering. In the remainder, primarily the Live Edge Start-up Delay and the Encoding+Distribution Latency are considered, but for some use cases also the end-to-end latency may be relevant.

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

Compression Efficiency:

- The efficiency of the service with respect to quality and bitrate expressed as the bitrate increase for the same quality due to reducing the latency compared to regular DASH streaming.

Network Efficiency and Scalability:

- Reduction of the cache efficiency, for example the increase of number of objects may result in a drop of cache efficiency
- Increase of the Total Number of Requests (both for segments and MPDs)
- Increase of the Number of Invalid Requests
- Can HTTP/2 be used for the solution? Are there impacts on HTTPS operation
- Number of requests for same/different/non-contiguous sub-ranges

Robustness to Bandwidth Variations and Errors

- Bandwidth: regular ABR logic and the impact
- Errors: CDN failures/delays, etc. and other issues
- Increase of rebuffering events and rebuffering time
- Rebuffering schemes: stay at live edge or build up buffer

Lower layer protocol changes needed

- Other protocols than TCP/IP (used for DASH/CDN distribution)
- TCP/IP modifications
- TCP/IP configuration options
- Addition of Forward Error correction

Other Factors

- SCTE-35 messages from contribution links to DASH packagers may be sloppy in the scheduling (still accurate on the content) and not properly used (for example ad avails are signaled without pre-warning, program changes happen w/o announcements, etc.), so decisions in the packager need to be delayed to overcome issues of inaccuracy.
- Also no look-ahead available for applying segment generation.
- Overall processing delay needs to be added to overcome the sloppiness.

6.3 External Benchmark Numbers

6.3.1 Live Edge Start-up Delay and channel change times for other services

- 0.6 seconds to 2 seconds are reported here http://informitv.com/2008/10/13/channel-change-times-for-iptv-are-faster-than-satellite/
- 1.9 seconds to 4 seconds are reported here https://en.wikipedia.org/wiki/Zap_time
- Typically services target a channel change of 1 second which is good enough for most people.

6.3.2 Encoding and distribution latency

- BBC reported the following numbers:
    - We checked the encoding and distribution latency we have for our broadcast services (across DTT, DSAT, cable). This varies by platform as you might expect. The minimum is approximately 3 seconds and the maximum is a little under 6 seconds, when measured from encoder input to display on a TV.
    - So I'd like to note a range of 3-6 seconds for this as it will be important to some use cases to be aware that broadcast delay may be as little as 3 seconds
    - Of course, others may have additional input on this topic, with smaller or larger latencies.

6.3.3 Social network feed latencies

In some studies (https://arxiv.org/ftp/arxiv/papers/1106/1106.4300.pdf, clause 5), the delay for Superbowl event on twitter is a minimum 13 seconds, but the analysis assumes that the notification is from users watching on TV and assume a TV delay of 7-12 seconds. This means that the minimum Twitter delay can be assumed in a range of 5 seconds between the event and the notification being available. The study also mentions that specific hash tags may be much faster than personal feeds.

7 Deployment Experience and Issues

7.1 Introduction

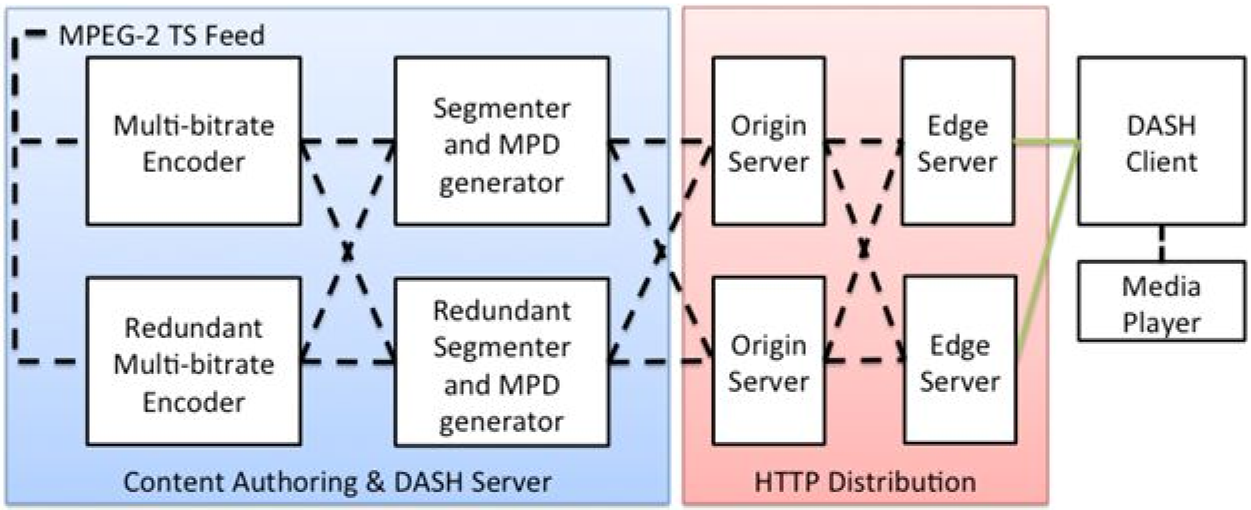
This section collects some deployment experiences on existing service and background material.

7.2 DASH-IF on Live Services

7.2.1 Background

This section summarizes the DASH-IF background on live services as documented in the IOP Guidelines, Annex B.

7.2.2 Live Service Architecture



The figure depicts a redundant set-up for Live DASH with unicast. Function redundancy is added to mitigate the impact of function failures. The redundant functions are typically connected to multiple downstream functions to mitigate link failure impacts.

An MPEG2-TS stream is used often as feed into the encoder chain. The multi-bitrate encoder produces the required number of Representations for each media component and offers those in one Adaptation Set. The content is offered in the ISO BMFF live profile with the constraints according to v2 of this document. The encoder typically locks to the system clock from the MPEG2-TS stream. The encoder forwards the content to the segmenter, which produces the actual DASH segments and handles MPD generation and updates. Content Delivery Network (CDN) technologies are typically used to replicate the content to multiple edge servers. Note: the CDN may include additional caching hierarchy layers, which are not depicted here.

Clients fetch the content from edge servers using HTTP (green connection) according to the MPEG-DASH and DASH-IF IOP specification. Different protocols and delivery formats may be used within the CDN to carry the DASH segments from the segmenter to the Edge Server. For instance, the edge server may use HTTP to check with its parent server when a segment is not (yet) in the local cache. Or, segments may be pushed using IP Multicast from the origin server to relevant edge servers. In some deployments, the live service is augmented with ad insertion. In this case, content may not be generated continuously, but may be interrupted by ads. Ads itself may be personalized, targeted or regionalized.

7.2.3 Typical Problems in Live Distribution

A few typical problems and challenges for live distribution are summarized in Annex B of the DASH-IF IOP:

- Client Server Synchronization Issues: In order to access the DASH segments at the proper time as announced by the segment availability times in the MPD, client and server need to operate in the same time source, in general a globally accurate wall-clock, for example provided by NTP or GPS. There are different reasons why the DASH client and the media generation source may not have identical time source.
- Synchronization Loss of Segmenter: The segmenter as depicted in Figure 25 may lose synchronization against the input timeline for reasons such as power-outage, cord cuts, CRC losses in the incoming signals, etc.
- Encoder Clock Drift: In certain cases, the MBR encoder is slaved to the incoming MPEG-2 TS, i.e. it reuses the media time stamps also for the ISO BMFF.
- Segment Unavailability: When a server cannot serve a requested segment it gives an HTTP 404 response. If the segment URL is calculated according to the information given in the MPD, the client can often interpret the 404 response as a possible synchronization issue, i.e. its time is not synchronized to the time offered in the MPD.
- Swapping across Redundant Tools: In case of failures, redundant tools kick in. If the state is not fully maintained across redundant tools, the service may not be perceived continuous by DASH client. Problems that may happen at the encoder, that redundant encoders do not share the same timeline or the timeline is interrupted. Depending on the swap strategy ("hot" or "warm"), the interruptions are more or less obvious to the client. Similar issues may happen if segmenters fail, for example the state for segment numbering is lost.
- CDN Issues: Typical CDN operational issues are Cache Poisoning and Cache Inconsistency.
- High End-to-end Latency: End-to-end latency (also known as hand-waving latency) is defined as the accumulated delay between an action occurring in front of the camera and that action being visible in a buffered player. It is the sum of
    a. Encoder delay in generating a segment.
    b. Segment upload time to origin server from the encoder.
    c. Edge server segment retrieval time from origin
    d. Segment retrieval time by the player from the edge server
    e. The distance back from the live point at which the player chooses to start playback.
    f. Buffering time on the player before playback commences.

  In steps a through d, assuming non-chunked HTTP transfer, the delay is a linear function of the segment duration. Overly conservative player buffering can also introduce unnecessary delay, as can choosing a starting point behind the live point. Generally the further behind live the player chooses to play, the more stable the delivery system is, which leads to antagonistic demands on any production system of low latency and stability.
- Buffer Management & Bandwidth Estimation: The main user experience degradations in video streaming are rebuffering events. At the same time, user experience is influenced by the quality of the video (typically determined by the bitrate) as well as at least for certain cases on the

end-to-end latency. In order to request the access bitrate, the client does a bandwidth estimation typically based on the history and based on this and the buffer level in the client it decides to maintain or switch Representations. In order to compensate bandwidth variations, the client buffers some media data prior to play-out. More time buffer results less buffer under runs and less rebuffering, but increases end-to-end latency. In order to maximize the buffer in the client and minimize the end-to-end latency the DASH client would like to request the media segment as close as possible to its actual segment availability start time. However, this may cause issues in the playout as the in case of bitrate variations, the buffer may drain quickly and result in playout starvation and rebuffering.

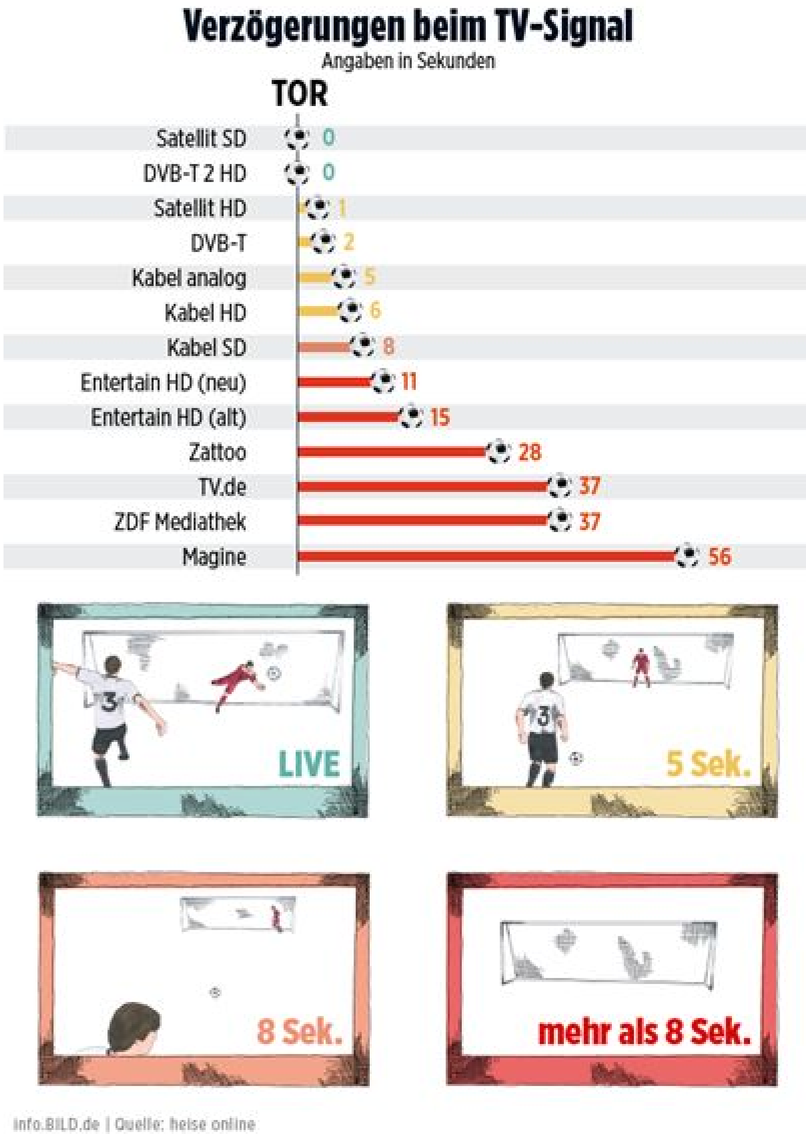
- Start-up Delay and Synchronization Audio/Video: At the start-up and joining, it is relevant that the media playout is initiated, but that the delay at start is reasonable and that the presentation is enabled such that audio and video are presented synchronously. As audio and video Representations typically are offered in different sampling rates, and segments of audio and video are not aligned at segment boundaries. Hence, for proper presentation at startup, it is necessary that the DASH client schedules the presentation at the presentation time aligned to the over media presentation timeline.

7.3 Prevailing Ancient HLS Recommendations

Very often HTTP Streaming is claimed to have unacceptable latency, but to be accurate, it is HLS and the recommended setting that create the latency https://developer.apple.com/library/ios/technotes/tn2224/_index.html: §The value you specify in the EXT-X-TARGETDURATION tag for the maximum media segment duration will have an effect on startup. *We strongly recommend a 10 second target duration*. If you use a smaller target duration, you increase the likelihood of a stall. Here's why: if you've got live content being delivered through a CDN, there will be propagation delays, and for this content to make it all the way out to the edge nodes on the CDN it will be variable. In addition, if the client is fetching the data over the cellular network there will be higher latencies. Both of these factors make it much more likely you'll encounter a stall if you use a small target duration.” If you use HLS with 10 seconds segments, buffer 2-3 segments at the client, this immediately results beyond 30 seconds end-to-end latency. According to Apple, initial HLS was never meant to be a low-latency live distribution system. Apple has revised this recommendation, but it is still prevalent in many deployments. In many cases the legacy HLS latency is considered as the ABR Latency.

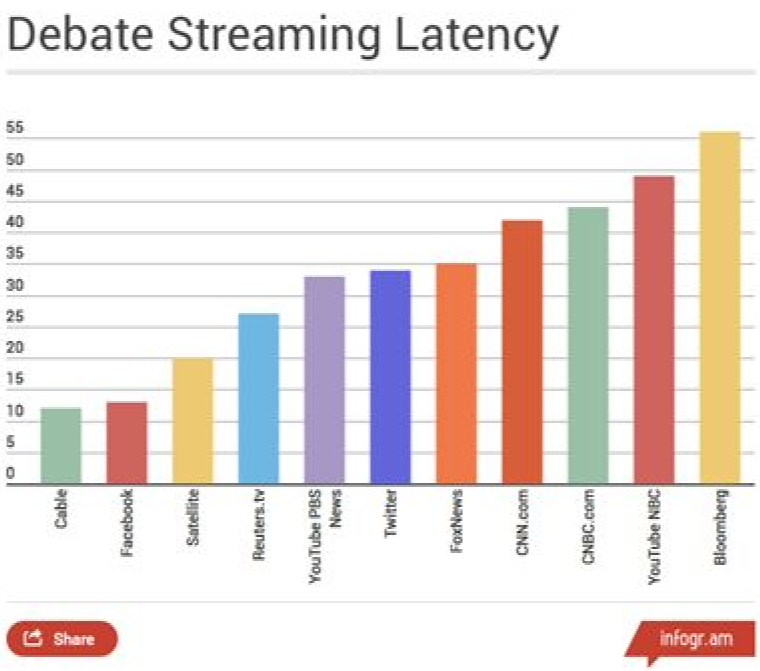
7.4 Some Reported Numbers

The below figure shows the delays of different distribution means for TV Signals in Germany during the European Championship in 2016. This was reported by Bild. OTT latency such as Zattoo or Mediathek loose 30 second and more for compared to existing TV services.



Similar data was shown by Red Bull during the DVB World, the slide is copied below:

The numbers from the presidential debate in the US are provided in the following diagram:



OTT streaming services again have significantly longer delay than regular TV systems.

7.5 Summary of Wowza Report

In July 2017, Wowza published a report on "Low Latency Streaming Media Impacts UX[1]". Wowza summarizes that the user experience of streaming media content is heavily impacted by three metrics:

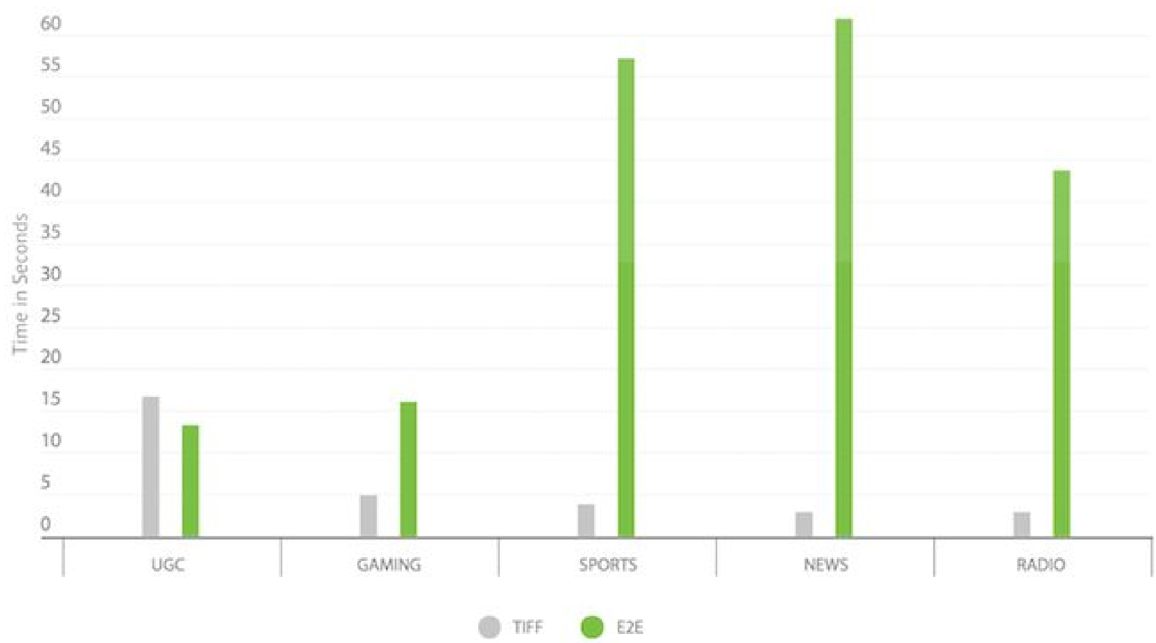
---

[1] https://www.wowza.com/blog/report-low-latency-streaming-media-impacts-ux

- End-to-end latency: The time between video or audio being captured at the source and when it plays back on an end-user's device.
- Time to first frame (TTFF): The time it takes for content to initially load on an end-user's device.
- Perceived quality: The quality of streaming content, as perceived by an end user—including the video resolution; audio clarity; and performance and stability of the platform.

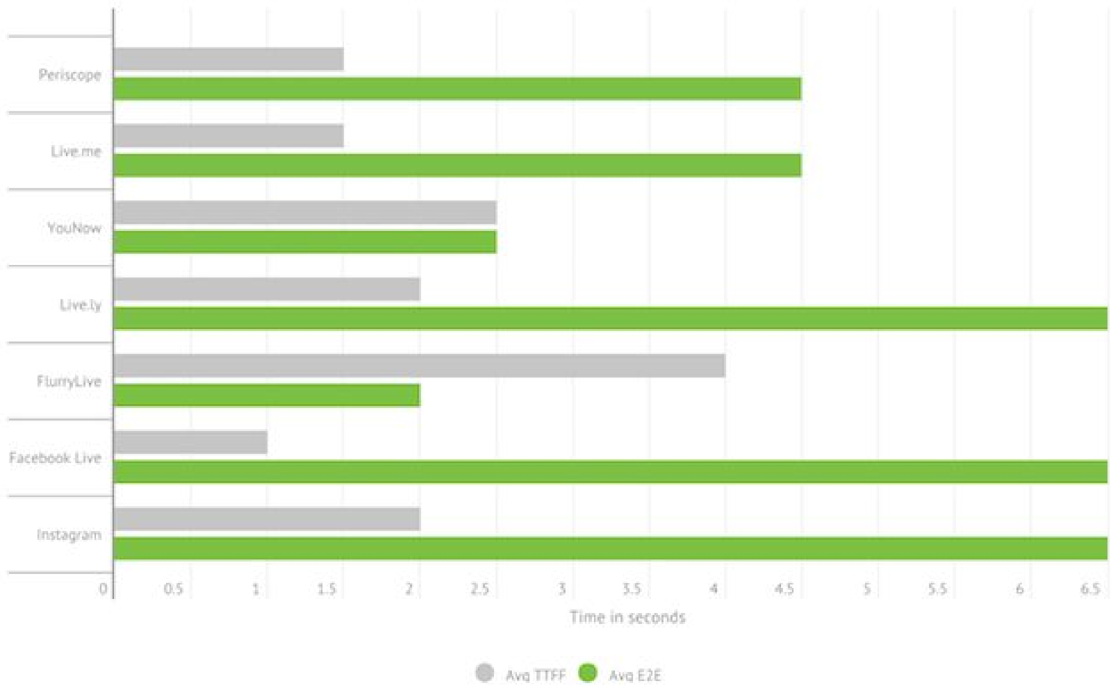
The earlier two are also analysed deeply in this report.

The report discusses the requirements on latency and the performance of streaming media content providers performing in terms of latency. To find out, Wowza conducted a study of several high-profile platforms across five different industry verticals. They tested each provider's performance on end-to-end latency and TTFF—and discovered some interesting trends.

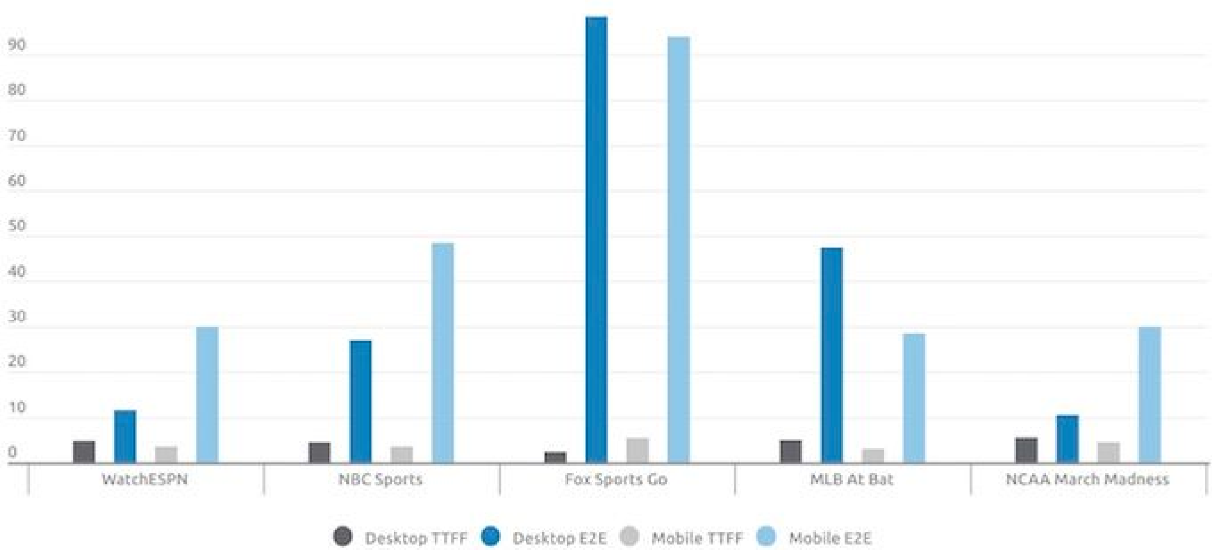


A key use case for low latency is gaming, but this is not discussed in the use cases in clause 4.

For Interactive COntent and User Generated Content apps (addressing some use cases), the performance is as follows:



For sports apps the performance is as follows:



The report also discusses performance of news channels and radio services in more details. The following insightful summary is provided.

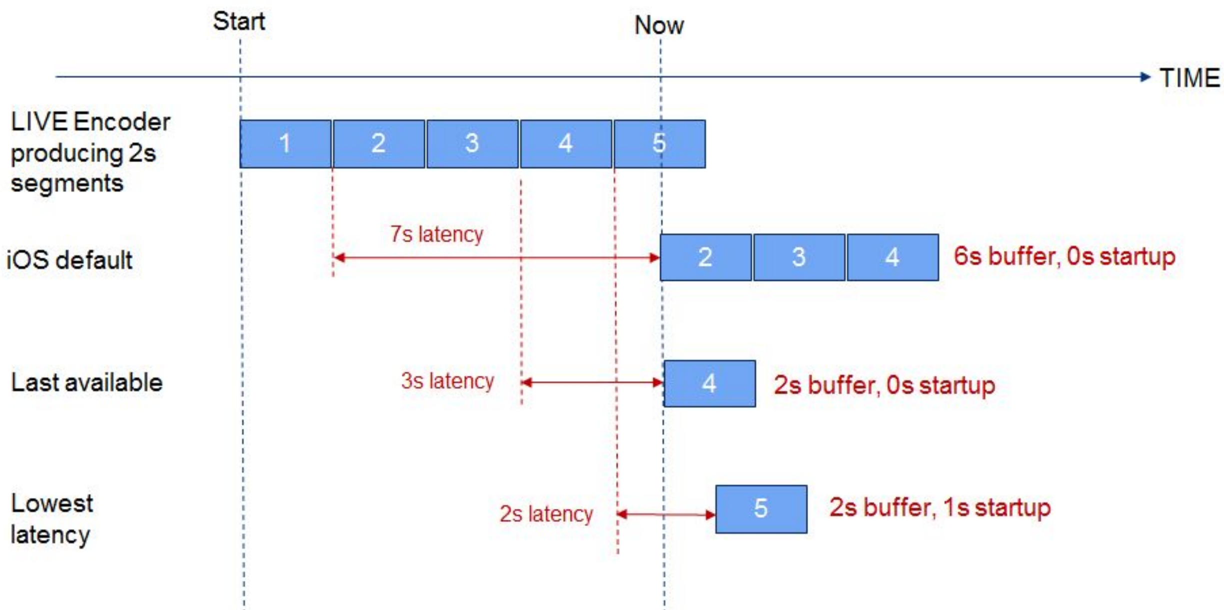
THE FOLLOWING TRENDS HOLD ACROSS INDUSTRIES:

- Near real-time streaming, facilitated by the lowest possible latency, is key for platforms where the UX focuses on interactivity (either viewer-to-broadcaster or viewer-to-viewer).

- Latency that is competitive with broadcast TV or OTA radio is crucial for platforms with a high proportion of time-bound coverage.
- For platforms where the UX involves streaming to large numbers of concurrent viewers on a range of devices and connections, higher latency is necessary to provide a high-quality experience.

7.6 Analysis and Background

The figure below shows the start-up latency and different client strategies. Typically in order to be able to build a buffer, the DASH client should start playback with the latest available segment, but with an older segment on order to build buffer afterwards.



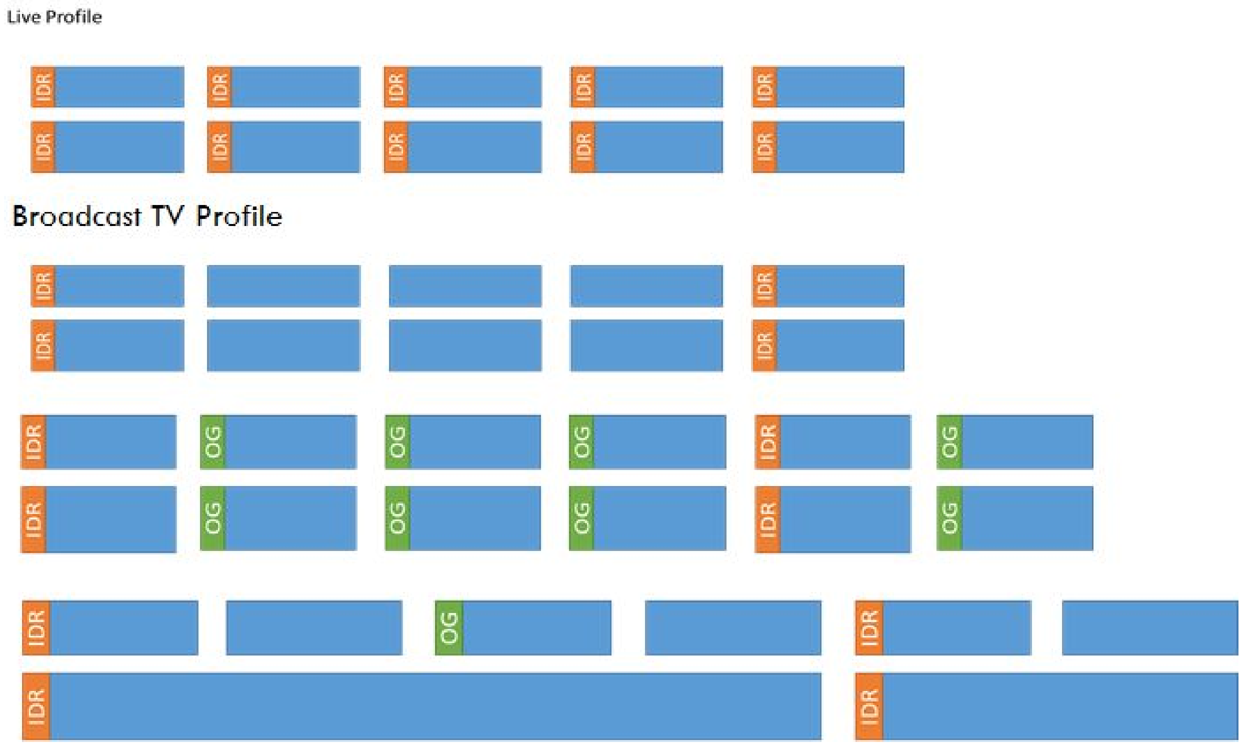
8 Available and Ongoing Work in Industry and Standards

8.1 MPEG-DASH

In MPEG DASH, two primary efforts for reducing latency have been considered:

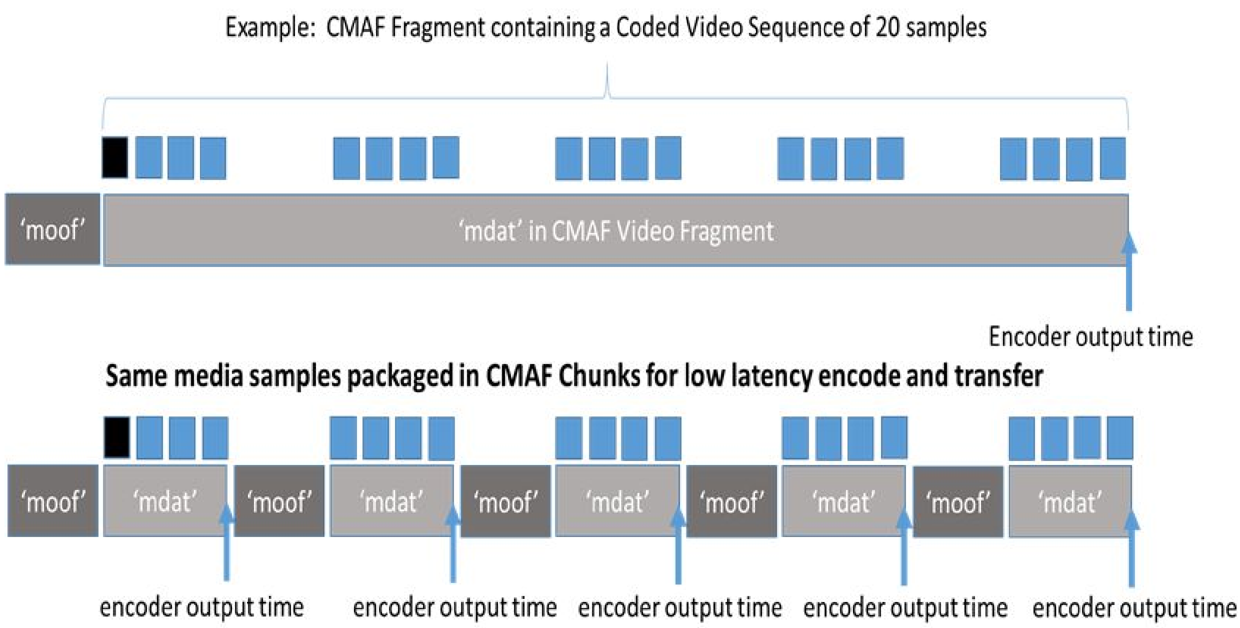
- Low latency segment mode available in Amd.2 of the second edition. This permits to signal to the DASH client that a segment can be accessed earlier than full availability at the server for progressive download/chunked delivery.
- Broadcast TV profile available in Amd.4 of the second edition: In this case multiple efforts are done to reduce EEL and LSD including the ability to create shorter segments without losing

compression efficiency, use inband Initialization Segments, and create Segment Sequences. The figure below shows the options to generate different segment patterns.



8.2 CMAF

Both modes are complemented by CMAF low latency chunk mode, i.e. CMAF chunks can be distributed with DASH using one of the above two methods as shown in figure below:



8.3 ATSC3.0

Based on a set of requirements for different categories developed by ATSC 3.0, a DASH profile was developed that addresses the use cases, but at the same time takes into account the convergence of ATSC 3.0 delivery formats with OTT delivery formats. As a baseline for the DASH formats, the DASH-IF Interoperability Point is considered, with the available extensions for different media profiles. A DASH-IF IOP provides a basic DASH profile for MPDs and segments formats, specific recommendations for live services based on this profile, enablers for targeted ad insertion, content protection recommendations as well as media profiles for video, audio and captioning. However, in order to address all requirements for ATSC 3.0, extensions to the latest DASH-IF's IOP are necessary and the relevant ones, along with the key use cases, are provided in the following.

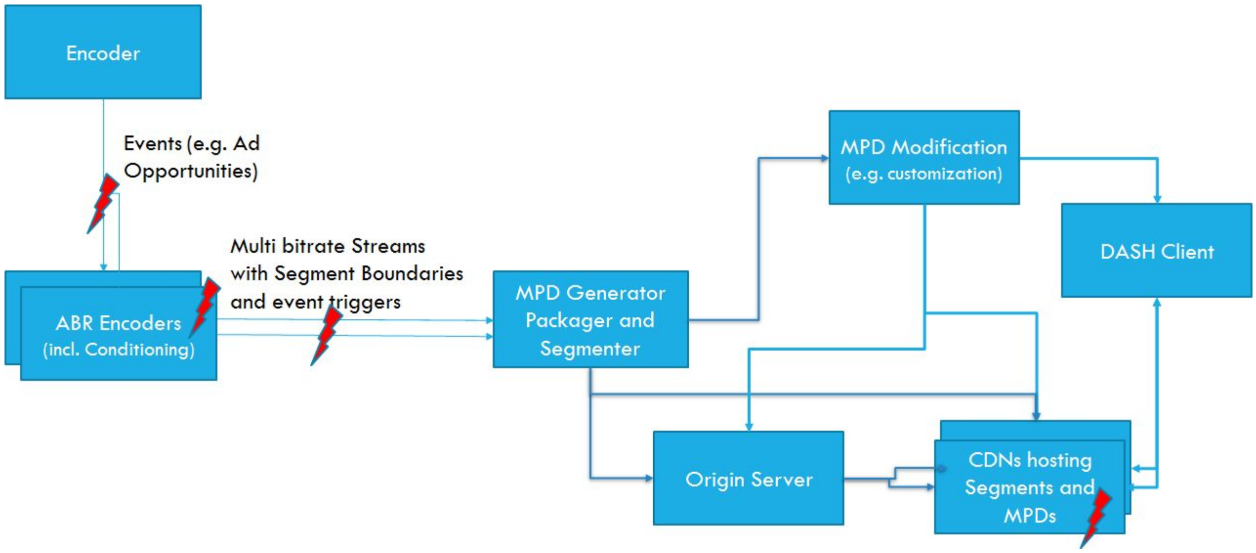
In Broadcast Distribution, the broadcast channel is the only communication channel available to the DASH Player. Therefore, the DASH Player can only receive MPDs and media segments through the broadcast channel. In contrast to hybrid delivery, no return channel capability is available. Key aspects for linear TV services, in particular, broadcast services, are end-to-end latency and rapid channel change times. The distribution format integrates with ROUTE/UDP/IP for broadcast. The distribution format needs to support synchronization of supplemental content, such as accessibility components, supplementary languages, etc. with primary content at the receiver, referred to as late binding. In addition to the broadcast channel, a broadband channel may also be available to the DASH Player. Since only a single MPD is used to signal details of Media Segments, the DASH Player may receive one MPD for entire program and then receives the corresponding Media Segments through the broadcast channel and/or the broadband channel.

Based on these requirements, a Broadcast TV profile is developed together with MPEG that addresses the use cases taking into account the following features beyond the DASH-IF IOP:

- In order to support low-latency, random access, adaptive switching, and highest compression efficiency, different segment types are defined in order to address the individual functionalities, namely delivery units, random access units and switching units. These segment types extend the DASH-IF IOP (5) as in the DASH-IF requirements, each Segment is at the same time delivery unit, random access unit and switch point. Whereas this simplified initial deployments, it did not address compression efficiency and latency requirements of ATSC. These extended segment types are also defined in DASH TV Broadcast profile, a new part of the upcoming 3rd edition of MPEG's DASH specification.
- Segment address and time signalling is restricted to Segment Timeline only in order to address different use cases, including switching and random access point signalling, gap signalling in case of losses, and support for redundant server setup.
- Segment addressing is primarily based on a number-based template in order to support efficient prediction on transport level and avoid repeated delivery of metadata.
- Extensions for metadata to support the codecs and requirements as documented in following.

8.4 DASH-IF

Beyond the live service guidelines, DASH-IF recently developed detailed content generation guidelines for advanced live services. In the considered scenario, a service provider wants to run a live DASH service according to the below Figure.



As an example, a generic encoder for a 24/7 linear program or a scheduled live event provides a production encoded stream. Such streams typically include inband events to signal program changes, ad insertion opportunities and other program changes. An example for such signalling are SCTE-35 [XXX] messages. The stream is then provided to one or more Adaptive Bitrate (ABR) encoders, which transcodes the incoming stream into multiple bitrates and also conditions the stream for segmentation and program changes. These multiple encoders may be used for increased ABR stream density and/are then distributed downstream for redundancy purposes. The resultant streams are received by theDASH generation engines that include: MPD generator, packager and segmenter.Typically the following functions are applied by the MPD packager:

- Segmentation based on in-band information in the streams produced by the ABR encoders
- Encapsulation into ISO BMFF container to generate DASH segments
- Dynamic MPD generation with proper customization options downstream
- Event handling of messages
- Any other other DASH related adaptation

Downstream, the segments may be hosted on a single origin server, or in one or multiple CDNs. The MPD may even be further customized downstream, for example to address specific receivers. Customization may include the removal of certain Adaptation Sets that are not suitable for the capabilities of downstream clients. Specific content may be spliced based on regional services, targeted ad insertion, media blackouts or other information. Events carried from the main encoder may be interpreted and removed by the MPD packager, or they may be carried through for downstream usage. Events may also added as MPD events to the MPD.

In different stages of the encoding and distribution, errors may occur (as indicated by lightning symbols in the diagram), that for itself need to be handled by the MPD Generator and packager, the DASH client, or both of them. The key issue for this section is the ability for the DASH Media Presentation Generator as shown in to generate services that can handle the incoming streams and provide offerings such that DASH clients following DASH-IF IOPs can support. Detailed guidelines for implementation on MPD Generators and Packagers in the next version of the DASH-IF IOPs.

9 Extracted Service Requirements

9.1 Summary of Use Cases and Expected Impacts

Using the use cases in clause 6 and the information this clause, the KPI impacts are summarized.

Use Case	Summary	KPI Impacts
1	On par with other Distribution Means	*EDL req*, typically as low as **3 seconds up to 10 seconds**. If the delay is larger, the service quality gradually degrades.
2A	ABR competing with Social Feeds originating from the venue	*EEL* typically as low as **5 seconds**
2B	ABR competing with Social Feeds originating from other TV viewers	*EEL* req typically a sum of the above two values, i.e. **8-15 seconds**
2C	ABR competing with Social Feeds originating from other TV viewers	In this case the EEL is typically in between 2A and 2B.
3	Companion Streams and Screens	*EDL* req, similar to use case 1, but if the delay is larger, than the service typically fails entirely.
4	Sports Bar	*EDL* req, typically as low as **3 seconds up to 10 seconds**.
5	Variably Configured Latency across Channels	*EDL* req, typically as low as **3 seconds up to 30 seconds**. However, the latency variations may be used for improved operational performance or reduced cost operations.
6	DASH in ABR Multicast	The use case does not add any new aspects for the user perception. It more to take into into account conversion from unicast to multicast and reverse. And possible protocol constraints.

		More a system use case.
7	DASH as broadcast format	The use case does not add any new aspects for the user perception. It more to take into into account conversion from unicast to multicast and reverse. And possible protocol constraints. More a system use case.
8	Enterprise Broadcast	*EEL* typically **1/1.5/2s to 5 s**. The delay requirements may be such low as interactivity may be involved.
9	Hybrid Broadcast	The use case does not add any new aspects for the user perception. It more to take into into account conversion from unicast to multicast and reverse. And possible protocol constraints. More a system use case.
10	Personal Broadcast	*EEL* typically as low as **5 seconds**. In addition, the use case Single bitrate potentially causing addition buffering limitations.
11	Channel Bouquet	*EEL* typically configurable between **3 and 30 seconds**. At the same time *start-up and channel change times* are as low as **1-2 seconds**.
12	Channel Bouquet from one Event	*EEL* typically configurable between **3 and 30 seconds**. At the same time *start-up and channel change times* are as low as **0.5-2 seconds**. Synchronization of the event streams.

9.2 Impacts on Performance

9.2.1 Introduction

The above use cases and the summary reveal that individual enablers as well as the combination of the technologies are necessary for the fulfillment of the service requirements. The enablers are typically provided by different ecosystem actors. In the following the actors are briefly described and their influence on the service performance is documented.

9.2.2 Content Provider

The Content provider controls the entire chain from content capturing to providing it to the Service Provider.
The content provider impacts among others:

- The delay from the live event to the Service provider
- Acquisition delays such as capturing, profanity and so on
- Possibly delays to react to service configurations such as SCTE-35 messages, virtual segmentation, etc.
- Multiple streams/channels from the venue with synchronization requirements
- Different delay and quality requirements for each service
- Impacts if the service is provided to one or multiple service providers for distribution

9.2.3 Service Provider

The Service Provider addresses encoding and packaging of the content, possibly adding and aggregating services, and provides this over one or different distribution networks.
The service provider impacts:

- The encoding of the content in terms of codecs, switch points, random access points, bitrate ladders and so on
- The packaging of the content in DASH, including MPD and segment formats
- The interface from the Packager to the distribution network
- The offering of the services in for example a bouquet
- The configuration of different delay and quality requirements for each service
- Program generation (of different) content, ad splicing, configuration changes
- The provisioning of the content to different distribution networks
- Security related aspects (encryption, etc.)

9.2.4 Distribution Network

The Distribution network addresses distribution of the content from an origin server to the DASH clients.
The distribution network impacts:

- The delay of the distribution
- The scalability of the distribution
- The number of concurrently served clients
- The distribution network may use different protocols, protocols that support lower latency or improved distribution such as broadcast or multicast, etc.
- Secure delivery such as https
- QoS guarantees, if the network provides QOS features.
- Policies and operation of CDN: proactive and reactive caching, etc.

9.2.5 DASH Client

The DASH client addresses downloading of the content and provides a continuous media stream to the decoding and rendering engine.
The DASH client impacts:

- The start-up delay (DASH client may have different configuration options, but someone needs to choose the proper configuration based on service information and requirements)
- The buffering delay (DASH client may have different configuration options, but someone needs to choose the proper configuration based on service information and requirements)
- The rate selection and ABR logic (DASH client may have different configuration options, but someone needs to choose the proper configuration based on service information and requirements)
- The selection of media and different encoding options
- The continuous media download and ensuring service continuity
- The accuracy of the clock

9.2.6 Media engine

The Media engine addresses playback of the content.
The Media engine impacts:

- Playback and rendering quality
- Handling of errors
- Handling of buffer underflows
- etc.

9.3 Scope of normative and guidelines work

This section summarizes where normative work may be done and where guidelines may be provided for proper service performance.

Normative Work:

- DASH Encoding & Packaging
- DASH client “implementation” configurations and signalling for configuration
- Media engine “requirements” or impacting factors

Guidelines:

- Content Generation (what can be harmful for delays)
- Network protocols and configuration, CDN operation, etc. This may include recommendations to use push-protocols (e.g. HTTP/2, Websockets, Broadcast/Multicast) to reduce latencies.

9.4 Typical KPI Service Requirements

A technical solution should enable:

- Live Edge Start-up Delay is comparable or better of existing TV services
- Distribution and encoding latency is comparable or better of existing TV services
- End-to-end latency is manageable such that social network feeds are not impacting the user experience.
- Multiple or all of the above aspects can be fulfilled at the same time
- For all cases the following aspects should be low:
  - Compression Efficiency Decrease
  - Reduction of Cache Efficiency
  - Increase of Total Number of Requests
  - Increase of Total Number of Invalid Requests
  - Increase of number of requests for non-contiguous sub-ranges
  - Increase of service rebuffering events
  - Increase of service rebuffering duration
  - Increase of service interruption
- The same service offering works for low-latency clients and regular existing clients.
- Necessary latency for service is determined upfront
- The parameters for latency and start-up delay are configurable per service in order to allow optimization of other KPIs.

10 Conclusion and Proposed Way Forward

Based on the report, the need for optimizing DASH services for live distribution is obvious. Different use cases are provided and service requirements are summarized. Also existing technologies are summarized that are not yet in DASH-IF and DVB DASH.

For DVB the following is recommended:

- Initiate a short effort to generate commercial requirements for DVB DASH live services based on the use cases and service requirements in the document with approval target for SB in Nov 2017.
- Collect additional information on deployment experience from different services
- Collect additional information on technology enablers to support low-latency DASH live services.

For DASH-IF the following is recommended:

- Initiate a work item to collect the relevant tools for low-latency in an Interoperability Point
- Provide alignment of DASH and CMAF to the extent possible
- Generate Test vectors and test services for such configurations
- Support dash.js team to implement a low latency mode
- Provide estimates on start-up delay and end-to-end latency performance for DASH-IF reference tools

11 Contributors

Name	**Company**
Thomas Stockhammer	Qualcomm Incorporated
Chris Poole	BBC
Thomas Swindells	Nokia
Will Law	Akamai
Iraj Sodagar	Microsoft
Ali Begen	Comcast
Thorsten Lohmar	Ericsson
Kilroy Hughes	CTA