

MPEG-DASH vs. Apple HLS vs. Microsoft Smooth Streaming vs. Adobe HDS



Written by:
[Christopher Mueller](#)
March 29th, 2015

All existing adaptive HTTP streaming technologies, such as the technologies, such as the proprietary Adobe HTTP Dynamic Streaming (HDS), Apple HTTP Live Streaming (HLS), Microsoft Smooth Streaming (MSS), and the only international standardized solution MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH) follow nearly the same principle

All existing adaptive HTTP streaming technologies, such as the proprietary **Adobe HTTP Dynamic Streaming (HDS)**, **Apple HTTP Live Streaming (HLS)**, **Microsoft Smooth Streaming (MSS)**, and the only international standardized solution **MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH)** follow nearly the same principle.

The basic idea is to generate multiple versions of the same content (e.g., different bitrates or spatial resolutions) and chop these versions into segments (e.g., two seconds). The segments are provided on a web server and can be downloaded through HTTP standard compliant GET requests. Typically, the relationship between the different versions is described by a

manifest represents the different qualities of the media content and the individual segments of each quality with HTTP Uniform Resource Locators (URLs). This structure provides the binding of the segments to the bitrate (resolution, etc.) among others (e.g., start time, duration of segments). As a consequence, each client will first request the manifest that contains the temporal and structural information for the media content, and based on that information it will request the individual segments that fit best for its requirements.

The adaptation to the bitrate or spatial resolution is done on the client side for each segment, e.g., the client can switch to a higher bitrate – if bandwidth permits – on a per segment basis, or to a lower bitrate – if bandwidth decreases. This has several advantages because the client knows its capabilities such as the received throughput, delay, device capabilities (e.g., screen resolution), etc. best.



Adaptive Streaming Feature Comparison

The following table shows a feature comparison of the proprietary

Apple HTTP Live Streaming (HLS), Microsoft Smooth Streaming (MSS), and the MPEG Dynamic Adaptive Streaming over HTTP (MPEG-DASH) standard. Please note that this comparison is based on the standard's capabilities and does not reflect the specific implementation of a certain vendor.

Feature	Adobe HDS	Apple HLS	MS Smooth Streaming	MPEG-DASH
Deployment on Ordinary HTTP Servers				
Official International Standard (e.g., ISO/IEC MPEG)				
Multiple Audio Channels (e.g., Languages, Comments, etc.)				
Flexible Content Protection with Common Encryption (DRM)				
Closed Captions / Subtitles				
Efficient Ad Insertion				
Fast Channel Switching				
Protocol Support's multiple CDNs in parallel				
HTML5 Support				
Support in HbbTV (version 1.5)				
HEVC Ready (UHD/4K)				
Agnostic to Video Codecs				
Agnostic to Audio Codecs				
ISO Base Media File Format Segments				
MPEG-2 TS Segments				
Segment Format Extensions beyond MPEG				

	✓	✓		✓
Support for non-multiplexed (separate Audio, Video) Content		✓	✓	✓
Definition of Quality Metrics				✓
Client Logging & Reporting				✓
Client Failover				✓
Remove and add Quality Levels during Streaming				✓
Multiple Video Views				✓
Efficient Trick Modes				✓

Deployment on Standard HTTP Servers

MPEG-DASH and also Apple HLS can be used with ordinary HTTP-Servers such as Apache, Nginx, IIS, etc. Adobe, as well as Microsoft, are using server side mechanisms that need additional logic on the server.



Official International Standard



MPEG-DASH is an international standard, ratified in 2012 and currently adopted by YouTube, Netflix, etc. Several members of different companies such as Microsoft, Adobe, Apple, Samsung, Akamai, Cisco, Dolby,

© 2014 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

Bitmovin, InterDigital, etc. have contributed to the standardization. The Apple HLS IETF Internet-Draft *) is driven by one company that is able to change its direction from one day to another. However, since May 2009, Apple has made no effort to move the HLS IETF Internet Draft to an international IETF standard.

***) Internet Draft:** Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. Internet-Drafts have no formal status, and are subject to change or removal at any time; therefore they should not be cited or quoted in any formal document.

Multiple Audio Channels

Switching between multiple audio channels is especially important for multi language content. MPEG-DASH supports this feature as well as Apple HLS and Microsoft Smooth Streaming.

Flexible Content Protection with Common Encryption

MPEG Common Encryption (CENC) allows content encrypted once, to be compatible with multiple DRM systems. This is possible, as nearly every DRM system supports AES as content encryption method and only the license key exchange between the client and the server is different.

Closed Captions / Subtitles

All formats support closed captions or subtitles. Usually they are referenced in the manifest and stored as an individual file, e.g., in [WebVTT](#) format.

Efficient Ad Insertion

Ad insertion is possible in all formats through chunk substitution. This means that individual chunks of the original video will simply be replaced by chunks that contain advertisements. MPEG-DASH enables a standardized interface through Periods that enables ad insertion in an efficient way. Which means that ordinary HTTP servers can still be used and no additional, proprietary logic is required to redirect requests for

Fast Channel Switching

Fast channel switching is a feature that is directly related to the chunk size, as smaller chunks allow faster channel change times than bigger chunks. Apple HLS typically uses 10 second chunks and is optimized for that chunk size. Adobe, Microsoft as well as MPEG-DASH are designed to work with 2 and 4 seconds chunks that allow faster channel change. Additionally, the overhead of the MP4 format used in MPEG-DASH and Microsoft has significantly lower overhead than the MPEG-2 Transport Stream (MP2TS) format used in Apple HLS. The tradeoff between large and small chunk sizes is that small chunk sizes enable fast channel switching, reducing startup latency for some systems (some players start the playback only if they have received the first chunk fully), and allow a more flexible adaptation behavior. On the other hand, small chunk sizes reduce coding efficiency as Group of Pictures (GOP) need to be smaller and therefore the codec can probably not fully benefit from more temporal redundancies when larger GOPs are also allowed. Additionally, decreasing the chunk size introduces overhead on the network layer as every chunk needs to be requested through HTTP, and therefore the protocol overhead gets introduced for each chunk.

Support Multiple CDNs in Parallel

Using multiple CDNs is possible with all formats. But only MPEG-DASH offers a signaling mechanism through BaseURLs on MPD level for the client where multiple CDNs are in use. The client application is therefore able to select the CDN that fits best, i.e., the CDN that gives the best performance in the network of the individual client. All formats that can not be deployed on ordinary HTTP servers, such as Adobe HDS and Microsoft Smooth Streaming, are more costly because they require specialized CDN features.



MPEG-DASH can utilize the [HTML5 Media Source Extensions \(MSE\)](#) to enable native playback in browsers.



Support in HbbTV (version 1.5 and 2.0)

MPEG-DASH is the official streaming format used in the HbbTV 1.5 and 2.0 specification.

Agnostic to Video Codecs

MPEG-DASH is agnostic to video codecs and can be used with several video codecs such as, H.264/AVC, [H.265/HEVC](#), MPEG-2 Video, VP8, VP9, etc.

Agnostic to Audio Codecs

MPEG-DASH is agnostic to audio codecs and can be used with several audio codecs such as, MP3, AAC, etc.

ISO Base Media File Format Segments

MPEG-DASH, Microsoft and Adobe can use ISO Base Media File Format based segments, e.g., MP4, that has a significant lower overhead than MPEG-2 TS segments used by Apple HLS.

MPEG-2 TS Segments

MPEG-DASH and Apple HLS can use MPEG-2 TS format segments that have a significant higher overhead than ISO Base Media File Format based segments. MPEG-2 TS consists of packets with 188 bytes in size, where each packet has headers with a varying size of 4 to 12 bytes. Therefore, the overhead caused by these headers increases proportionally with the segment size, which means that relative overhead does not tend to zero with increasing bitrates. Moreover, each MPEG-2 TS stream consists of several other data beside audio and video, such as Program Association Tables that introduce additional overhead. Additionally, audio and video

streams are encapsulated in Packetized Elementary Streams (PES) which introduces an extra overhead per audio sample or video frame.

HEVC Ready (UHD/4K)

MPEG-DASH is agnostic to video codecs and could be used with HEVC and 4K content encoded with [H.265/HEVC](#) or H.264/AVC.



Segment Format Extensions beyond MPEG

MPEG-DASH supports also container formats beyond MPEG such as WebM.

Support for Multiplexed (Audio + Video) Content

Adobe HDS, Apple HLS v6 and MPEG-DASH can be used with multiplexed audio video content, which means that audio and video elementary streams are packaged in one and the same segment.

Support for Non-Multiplexed (separate Audio, Video) Content

Apple HLS v8, MSS and MPEG-DASH can be used with separate audio video content.

Definition of Quality Metrics

MPEG-DASH defines quality metrics in the standard that enable the client to send specific reports about usage behavior, stalls, playback start, stop, etc. to a defined server.

Client Logging & Reporting

The quality metrics interface defined in the standard can be used to gather information from several different client vendors in the same way and enable universal reporting and client logging.

Client Failover

MPEG-DASH enables through BaseURLs that are also used to signal different CDNs client failover mechanisms, as it is possible for one client to choose several different locations to download a segment. A similar approach is available with Apple HLS using failover URLs, but in that case the MPEG-DASH specification enables more flexibility and more sophisticated use cases.

Remove and Add Quality Levels During Streaming

MPEG-DASH enables, through Periods, the dynamic insertion and deletion of quality representations during the streaming, with seamless handover at the client side.

Multiple Video Views

MPEG-DASH enables, through AdaptationSets, multiple views in one and the same MPD, and the client can seamlessly choose between these view angles during streaming. For example, a Formula 1 use case that has three camera perspectives would have three AdaptationSets that contain the Representations of the specific view angle. The client could then switch between these AdaptationSets which will present a different perspective for the user. Inside of an AdaptationSet (perspective) the client can adapt, based on the Representations that are available in the AdaptationSet, to the bandwidth conditions or the user context (device).



Efficient Trick Modes

MPEG-DASH enables, through AdaptationSets, efficient trick modes, allowing you to add specific Representations that contain only I Frames to simulate fast forward and rewind.

While [MPEG-DASH](#) offers nearly everything from a feature perspective, it is still a very young standard and not many encoders or players have implemented all of its features. But sooner or later, broad support will be available. If you want to [test MPEG-DASH](#) you can use the [Bitmovin encoding service](#). We offer a [free plan with 2.5GB](#) encoding output per month. That's great for testing MPEG-DASH and generating some content to play around with.

Newsletter

Sign up!

By submitting this form you acknowledge that you have read and agreed to the Bitmovin [privacy policy](#) and [terms](#).

Follow @bitmovin

6,050 followers

Latest from our blog

Fun with Container Formats – Part 3

July 12, 2019

Thanks for following us as we continue to dive into Fun with Container Formats.

This [...]

COMPANY

[ABOUT BITMOVIN](#)

[PRICING](#)

[AWARDS & MEMBERSHIPS](#)

[CAREERS](#)

[PARTNERS](#)

[CONTACTING BITMOVIN](#)

[PRIVACY POLICY](#)

[LEGAL NOTICE](#)

[TERMS AND CONDITIONS](#)

SUPPORT

[ENCODING](#)

[PLAYER](#)

[ANALYTICS](#)

[F.A.Q.](#)

DEVELOPMENT

[DEVELOPER PORTAL](#)

[GITHUB](#)

NEWS & EVENTS

[BLOG AND EVENTS](#)

[PRESS PAGE](#)

CONTACT

Bitmovin Inc

41 Drumm Street

San Francisco | CA 94111

| USA

1-833-248-6686

Schleppe Platz 7 | 9020

Klagenfurt

Austria | Europe

+43 463-203-014

[Contact Us](#)

©2019 Bitmovin Inc

[San Francisco](#) • [Vienna](#) • [Chicago](#) • [Seattle](#) • [New York](#) • [Klagenfurt](#) • [London](#) • [Hong Kong](#) • [Sao Paulo](#)