# A Survey on Adaptive Video Streaming Technologies

**Suzen Saju Kallungal**

*Abstract*— **With the growth in demand and use of various wireless mobile devices like smartphones, tablets, etc, video streaming has gained a lot of popularity. Video Streaming can be stated as a technique to transfer data such that it can be processed as a steady and continuous stream. Online video has taken a major leap with service providers having the daunting task of ensuring seamless viewing experience for all its subscribers. Things get more complicated when factors like network variations, variations in device capability, and the lot has to be managed so as to meet the end user QoE requirements. Adaptive video streaming is fast becoming the preferred method to stream Video-on-Demand (VoD) as well as live videos. This paper tries to provide an overview of the trends in adaptive streaming and surveys the different adaptive streaming technologies.**

*Index Terms*— **Adaptive Video Streaming, Seamless Viewing, QoE, VoD.**

## I. Introduction

Few years ago, the only way for content playback was to download the entire file, waiting for several minutes based on the connection speed, before we could watch the video.
With the effort from service providers to provide seamless viewing experience to its subscribers, online video delivery has reached new heights by now.
Three key technologies namely progressive download, RTMP/RTSP streaming and adaptive streaming fill all segments of online video delivery [10]. An evolution to simple download was server based delivery over HTTP. As the file is sufficiently downloaded (or buffered) on client machine, it can be played back. This basic version is called *Progressive Download*. Download of content happens sequentially and hence skipping ahead is not allowed. It was a significant improvement from earlier simple download where users could not play, until entire file was downloaded.
*RTMP/RTSP chunk based delivery* is a content delivery mechanism using specialized streaming servers over RTMP (Real Time Messaging Protocol) or RTSP (Real time streaming protocol) streaming protocols. It transfers chunks of media which get instantly consumed by the media player without any local caching. The technology takes advantage of delivery over UDP protocol (with TCP rollover) and can achieve much faster transfer rates. *Adaptive Streaming* is now the most popular form of video streaming with benefits of quality switching and ease of delivery over HTTP. It aggregates segments of multi-bit encoded videos which are indexed and referenced by the client using a manifest file. This file contains the index of chunks and their location. Client downloads the manifest file and periodically requests for the highest quality of video that can be supported by its environment. Table I provides a comparison of the three streaming technologies.

***Manuscript received Feb, 2017.***

***Suzen Saju Kallungal**, Departmeni of Information Technology, APJ Abdul Kalam Technological University, TocH Institute of Science and Technology, Arakkunnam, India.*

**Table 1 Comparison of Basic Streaming Technologies**

| | Progressive Download | Streaming | Adaptive Streaming |
|---|---|---|---|
| Basic Principle | Client requests for fileusing HTTP GET and server sends the entire file over HTTP. | Server sends chunks of data based on client request. Just in time transfer of data. | Content is encoded at multiple bit rates. A manifest file maintains the details of the chunks and their location. Client requests best suited chunk from the list. |
| Transport protocol | HTTP over TCP | RTMP,RTSP over TCP/UDP | Simple HTTP server over TCP |
| Bandwidth usage | Less efficient and wastage of bandwidth as the entire file may not be played. | More efficient as only part of the file is downloaded being played. | Chunks can be cached and reused. Thus saving bandwidth. |
| Content Security | Stored locally. Less secure | No temporary storage. More secure. | DRM integration possible for specific adaptive streaming technologies. |
| Advantages | Easy to setup. No special license required. | Can access any part of the video without waiting for entire download. | High flexibility to change video quality on the fly. |
| Disadvantages | Bandwidth is wasted on data which is downloaded but not watched. | Adds significant cost and complexity for setup and operations Requires special network configuration for port enabling | Requirements to have multiple encoded version requiring additional content processing and storage |
| Example of Online video platforms | YouTube, Vimeo | Hulu | BBC, Netflix |

## II. Adaptive Bitrate Video Delivery

Online Video exists in form of live streaming and Video on Demand (VOD). In Adaptive Bitrate (ABR) HTTP streaming, the source video is encoded into discrete file segments known as 'fragments' or 'chunks.' These files can include video data, audio data or such other data such as subtitles, program information or other metadata. These data may be multiplexed in the file fragment or can be separated into different fragment files. The fragments are hosted on an HTTP server from which they are served to clients [1]. A sequence of fragments is called a 'profile.' A client is enabled to 'adapt' to varying network conditions by selecting video fragments from profiles that are better suited to the conditions at that moment. Computing the available network bandwidth is easily accomplished by the client, by comparing the download time of a fragment with its size. The key components in an adaptive HTTP streaming data flow consist of an encoder or transcoder, a packager (sometimes called a 'segmenter' or a 'fragmenter') and a CDN (Content Delivery Network)

## III. Comparative Studies

To solve the quandary of ensuring optimal quality in spite of the bandwidth-hungry nature of video and the lack of QoS controls in unmanaged networks, Apple, Microsoft, Adobe and MPEG have developed adaptive delivery protocols. Unlike traditional VoD where video is delivered at a precise bitrate and the underlying network has to guarantee certain level of service, video services over HTTP use the public Internet. So, they have to adapt to varying bandwidth and network conditions during playback. The MPEG committee has developed an open standard known as Dynamic Adaptive Streaming over HTTP (DASH) to enable such adaptive services over the public Internet. Video services using DASH pre-fetch segments of video and play the downloaded segments continuously. This paper surveys the four primary HTTP adaptive streaming technologies: Apple's HTTP Live Streaming (HLS), Microsoft Silverlight Smooth Streaming (MSS), Adobe's HTTP Dynamic Streaming (HDS) and MPEG's Dynamic Adaptive Streaming over HTTP (DASH). [1]

### *A. HLS*

Apple introduced HTTP Live Streaming (HLS) in June 2009 with their iPhone OS 3.0. The operating principle of HLS is to work with segmented TS-based video streams or files. HLS uses the MPEG transport stream (TS), also used for satellite broadcasting and IPTV on managed networks [1]. The chosen HLS codec is MPEG H.264 for video and AAC for audio.

The way to achieve HLS streaming is to:

- Encode video in H.264/TS format at different bitrates
- Use a stream segmenter to generate short "chunks" of content –about 10 seconds each - and generate a playlist file (m3u or m3u8) indicating where to download the chunks
- Distribute through an HTTP server, and provide appropriate caching

Advantages:

- Simple and efficient adaptive bitrate solution to cope with the fact that bandwidth is not managed on open networks.
- It is based on Transport Stream transmission technology, making it easy to integrate into the existing digital TV world

Disadvantages:

- Currently, there is no way to provide more than one audio track on an HLS stream
- Proprietary and may be a significant barrier for free and open source players.

### *B. MSS*

Silverlight Smooth Streaming delivers streams as a sequence of ISO MPEG-4 files[1] Usually these are pushed by an encoder to a Microsoft IIS server (using HTTP POST), which aggregates them for each profile into an 'ismv' file for video and an 'isma' file for audio [4]. The IIS server also creates an XML manifest file that contains information about the bitrates and resolutions of the available profiles.

The general principle is quite similar to HLS streaming:

- Encode video in H264 and audio in AAC (or VC-1/WMA), in different bitrates;
- Use a stream segmenter to generate fragments and mux them into a PIFF container;
- Distribute through a HTTP web server, and provide appropriate caching.

Advantages

- Smooth Streaming natively supports multiple audio tracks, and multiple subtitles in the stream.
- DRM is already well integrated in Smooth Streaming

Disadvantages

- Smooth Streaming may not be adopted as fast as HLS
- It is based on patented audio and video codecs, so its use may be subject to license fees

### *C. HDS [3]*

A typical video stream is made up of:

- An XML-based manifest file (.f4m)
- Segmented files (.f4f) that contains fragmented MPEG-4 chunks
- Index files (.f4x) that contains specific information about the fragments inside the segmented files

Advantages

HDS is widely documented by Adobe for VOD applications, and they provide some sample in source-code format to help the community develop some value-added features.

Disadvantages

As for Apple HLS, there is currently no way to provide more than one audio track on an HDS stream.

### *D. DASH*

DASH is the most feature complete and complex of all the protocols, as it incorporates many features similar to those in HLS and MSS [2]

It's based upon:

- A manifest XML-based file that acts as a playlist and media presentation description, similar to Microsoft Smooth Streaming.
- A delivery format for video chunks, that can be an extension of either ISO Base file format, similar to MPEG4 container (fragmented MPEG4, as in Smooth Streaming or Adobe Dynamic Streaming) or MPEG-2 Transport Stream (as in HLS) 3GP container.

Advantages

- Takes the best of older technologies
- Based on an open standard

Disadvantages

- Very few players fully implement MPEG-DASH specifications (a preliminary work is done on VLC, with simple live 3GP profile)

## IV. Conclusion

This paper provides a bird's eye view of the existing HTTP Adaptive streaming technologies, comparing their basic features and principles, advantages and disadvantages. To conclude, no single format is self-composed to dominate another. An operator's final decision as to which format to be used should be based on the client devices served, DRM and the service delivered to customers. Future HTTP adaptive streaming technologies could be more QoE driven so that both the end-users as well as the stakeholders benefit from improved adaptation decisions.

## ACKNOWLEDGMENT

The author would like to thank the Department of Information Technology, TocH Institute of Science and Technology for their valuable support and guidance in conducting this survey.

## REFERENCES

[1] Yuval Fisher, " An Overview Of HTTP Adaptive Streaming Protocols For Tv Everywhere Delivery" *IEEE Trans. on Neural Networks*, 2014

[2] Iraj Sodagar et al., "The MPEG-DASH Standard for Multimedia Streaming Over the Internet," *IEEE Multimedia*, Dec.2011.

[3] Michael Seufert et al, **"**A Survey on Quality of Experience of HTTP Adaptive Streaming"**,** IEEE Communications Surveys & Tutorials 2015, vol: 17, Issue: 1 (469 – 492).

[4] Smooth Streaming Technical Overview, http://learn.iis.net/page.aspx/626/smooth-streaming-technical-overview/

[5] Jong-Min Jeong and Jong-Deok Kim, "Effective Bandwidth Measurement for Dynamic Adaptive Streaming over HTTP," IEEE ICOIN 2015

[6] Haiyan Yang and Xinxing JING, "*Adaptive Scalable Video Coding for Wireless Networks*", 2013 5th IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications

[7] *Xiaoqing Zhu and Bernd Girod,* "Video Streaming Over Wireless Networks", Information Systems Laboratory, Stanford University

[8] Tamizhselvi.S.P and Vijayalakshmi Muthuswamy, "Adaptive Video Streaming in Mobile Cloud Computing", 2014 IEEE International Conference on Computational Intelligence and Computing Research, pp: 1 – 4

[9] Lionel Bringuier, "Anevia OTT Streaming White Paper",2nd Edition, September 2011

[10] http://www.mediaentertainmentinfo.com/2015/04/6-concept-series-what-is-the-difference-between-progressive-download-rtmp-streaming-and-adaptive-streaming.html/

**Suzen Saju Kallungal** received her B.Tech degree in Computer Science and Engineering from Adi Shankara Institute of Engineering and Technology, MG University, Kerala in 2010. She is currently pursuing her M.Tech in Computer Science with specialization in Network Computing from TocH Institute of Science and Technology, APJ Abdul Kalam Technological University, Kerala, India**.**