White Paper

OTT Streaming – 2nd edition, September 2011

Lionel Bringuier, CTO, Anevia

This White Paper presents an overview of over the top (OTT) streaming and how it fits into the IPTV and VOD markets. It explains the principles of OTT, considers the differences between OTT and IPTV, looks at the challenges facing this new approach to service delivery and presents the major contenders aiming to become the industry's technical standard.

ANEVIA web site: http://www.anevia.com

Contents

Anevia OTT Streaming White Paper – Publication History

1st Edition (1.0) *Published October 20th, 2010*	2nd Edition (2.0) *Published September 7th, 2011*

What is over the top streaming?

OTT or over the top streaming is the delivery of video and audio media streams to different types of devices via the Internet. Unlike traditional IPTV, there is no need for a dedicated network or infrastructure provided by the operator, as OTT is transported through regular Internet data protocols and uses the open Internet over unmanaged networks.

What is the difference between OTT and IPTV?

By IPTV, we mean traditional IPTV which has already been widely deployed by numerous operators, namely those that offer a triple-play ADSL package to their customers.

IPTV is delivered over a dedicated, operator-managed network that is used only for broadcasting TV. The operator has full control over the network and can configure specific parameters, such as bandwidth consumption and jitter prevention to ensure a high level of service quality. Traditional IPTV uses TS (transport stream) transmission technology which is based on satellite TV broadcasting and delivers content over UDP in datagram mode.

OTT TV differs from IPTV as it transmits streams using HTTP, the protocol which has been used for decades to transport web pages over the Internet. HTTP is based on TCP, a connected transport protocol with more practical features than UDP. It is easier to track a TCP connection. As a result, a TCP connection can be easily managed through firewalls, NAT (network address translation) systems, home and office networks. It also enables anyone with sufficient web hosting capacity to broadcast audio and video content to a worldwide audience over the open Internet.

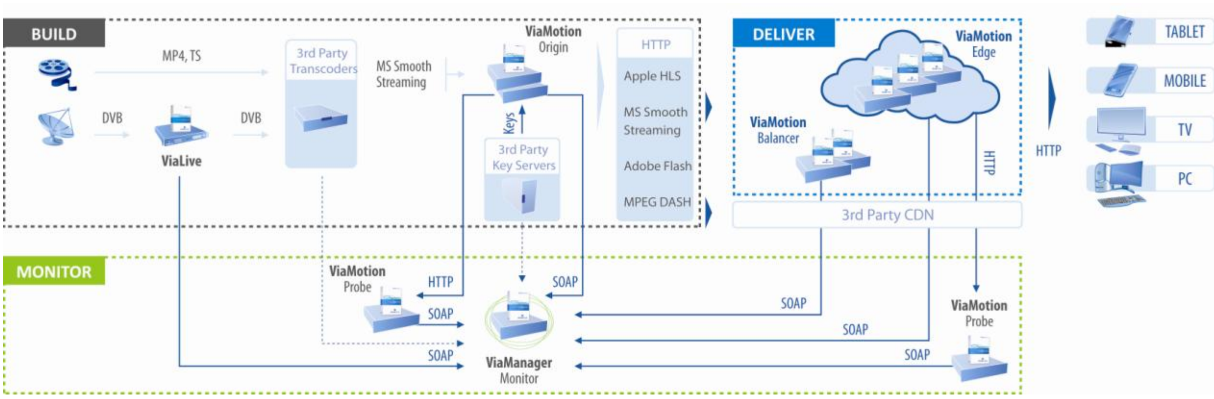
HTTP has already been used as a transport solution for video on demand (VOD) media embedded into web pages, especially on Adobe Flash-based sites, such as YouTube, Hulu and Dailymotion. However this solution does not stream in real-time, but instead relies on progressive downloading media files. The browser downloads the file from the HTTP web server and when it has a sufficient amount of data, starts to play the content while it continues to download the rest of the file. The main drawback to this approach is the length of time it takes to fill the initial buffer. Another issue associated with HTTP is streaming quality, which depends on the IP connection. Content streaming may be subject to stalling if there are fluctuations in bandwidth, leading to frame freezing. As a consequence, it is nearly impossible to use this solution to broadcast live channels.

Until recently, live broadcasting was therefore restricted to operator-managed IPTV networks using the UDP multicast protocol. The arrival of OTT streaming, however, has brought a new approach and it is now possible to achieve levels of streaming quality over HTTP that allow live content to be broadcast over the Internet.

What are the challenges in streaming content OTT?

If we take into account the principles described above, we can sum up the challenges faced by OTT service providers as follows:

- Video and audio content should be available wherever the Internet is accessible. HTTP must therefore be used as the transport protocol for these types of content;
- HTTP should be used for live TV broadcasting as well as for VOD content;
- As the open Internet is by definition an "unmanaged" network, end-user bandwidth cannot be controlled. This can lead to low streaming quality and negatively impact the user experience when watching TV. This issue is of particular importance for mobile networks;
- The proposed technology must be adapted for use on a full range of end-user devices (PC web browsers, STB/TV, mobile handsets, digital tablets, etc). This means that it must be easy to install and require very little system resources;
- It should also be easy to integrate into current digital TV workflows and ecosystems, because most content is now distributed using these formats and protocols (codecs, DRM, etc).



Which candidates are willing to take on these challenges?

Currently, there are four big players who propose their own OTT solutions. Unsurprisingly, these companies are also the same technology giants that are building the connected world in which we live:

- Apple®, promoting its HLS standard;
- Google™, pushing its own WebM technology;
- Microsoft®, with Silverlight Smooth Streaming;
- Adobe, with HTTP dynamic Streaming.

These companies have already achieved significant success in different areas of telecommunications and the Internet. However, at the time of writing, none of them has yet to emerge as a de facto standard for the media and broadcasting industry.

The battle between these protocols will be a long one. Apple®, Google™ and Microsoft® fought over your PC and Internet browser, then they battled over your mobile handset, and now they are looking to invade your living room.

Other traditional telecommunication-oriented organizations such as 3GPP or MPEG-LA have also joined the fight and this paper will also explore how a standardized protocol for OTT streaming, MPEG-DASH, may gain traction.

Why is everyone in the industry talking about OTT?

While OTT is key to the future business success of the aforementioned four big players, it is also disrupting the video delivery value chain, enabling many new players to enter the fray.

OTT for Telecom Companies

Up to now, high quality video delivery was the monopoly of managed network players, i.e. telecom operators (telcos) and Internet Service Providers (ISPs). The revolution came in 2002 when Fastweb (Italy) was among the first to propose a wide scale video over IP offer. France also played an active role with France Telecom and Free launching IPTV and VOD services as early as 2003.

As previously stated, the arrival of OTT means that it is no longer necessary to have a managed network to ensure quality of service in video delivery. This is a threat for telcos who fear "disintermediation" (the removal of intermediaries in the video supply chain – in this case the managed operator networks) and being relegated to simple Internet bandwidth providers, a position they have been fighting against for years.

There is, nonetheless, an OTT opportunity for telcos. Most of them are currently building new offers that enable them to reach new customers, and even to extend their video offers to subscribers of other ISPs.

Moreover, OTT can be also seen as a short-term enabler for multiscreen convergence (delivering video content to different user devices) because HTTP is used on PCs, Set-Top Boxes, connected TVs and mobile devices. Over the last few years, telecom infrastructure convergence has been built around the IMS (IP-Multimedia Subsystem) vision. However, OTT may be a smoother and faster path to multiscreen video convergence, allowing offers to arrive before IMS is widely adopted.

OTT for Content Providers

OTT enables new players to emerge by creating a direct link between content providers (TV channels, content aggregators, satellite and cable broadcasters) and end users. It enables the content providers to promote their video services directly to the viewer.

Content providers have suffered from the telco versus broadcaster battle over end user management while telcos have gained a significant advantage through IPTV and VOD offers included in triple-play packages. With OTT, content providers will be able to regain an edge by marketing and delivering their content directly to connected TVs and mobile devices.

OTT for Consumer Electronics Manufacturers

Before the arrival of OTT technologies, online video delivery was mainly limited to PCs and some high-end mobile phones. We now see the emergence of connected TVs entering the OTT arena. Connected TV enables consumer electronics manufacturers to bring value to TV delivery because it allows direct contact between the viewers and the content providers. Partnerships have been created between the TV set manufacturers and content providers to enhance their mutual offerings. Another new market for OTT content are tablets. The tablet wars, started by Apple's iPad, will continue to rage on over the next few years with fierce competition from both hardware and software manufacturers.

OTT for Electronic & General Retailers

Other examples of new emerging players able to take advantage of the opportunity in OTT are the electronic retailers who already have partnerships on the fixed and mobile telco markets, and MVNOs (mobile virtual network operators). They both have a strong customer base with a local presence. Their core business is to deliver devices to consumers such as TV sets, tablets and other connected video equipment. OTT can enable them to leverage their CRM (customer relationship management) with a complementary video offer, linked to the electronic devices that they are distributing.

Apple HLS

History

Apple introduced HTTP Live Streaming (HLS) in June 2009 with their iPhone OS 3.0 (which has since been renamed iOS). That makes HLS the oldest of the OTT technologies described in this white paper.

Nowadays, HLS streaming is without any doubt the most widespread protocol used for OTT, as it is available on all Apple devices (iPhone, iPad, iPod...) as well as on some software players and a number of set top boxes.

The keynote delivered by Steve Jobs on September 1st, 2010 was one of the first major events broadcasted live over HLS. It was also the day Jobs announced the second version of Apple TV, a set-top-box geared towards HLS streaming. The success of the iPad is largely based on users looking to use it for video application. A study by MeFeedia has shown that iPad owners watch three times more online video than traditional web users. Netflix and Hulu, who both offer Flash-based web sites, have launched their own iPad applications.. In July 2011, BBC launched the international version of its iPlayer iPad application allowing overseas subscribers to watch their favorite BBC shows over the top.

Principle

The operating principle of HLS is to work with segmented TS-based video streams or files. As a container HLS uses the MPEG transport stream (TS), also used for satellite broadcasting and IPTV on managed networks. The chosen HLS codec is MPEG H.264 for video and AAC for audio, widely used in the broadcast industry for many years.

The approach taken by Apple is based on using proven industry standards and modifying them slightly in order to fit with the requirements of an OTT solution. The less the modifications impact existing standards and technologies, the faster HLS will integrate into existing ecosystems.

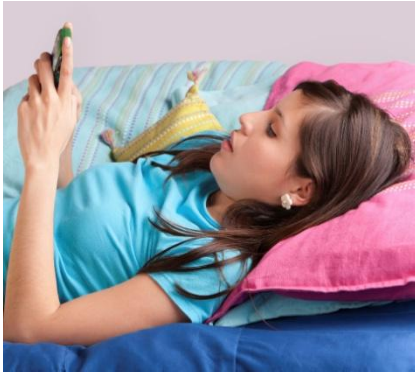
The way to achieve HLS streaming is to:

- Encode video in H.264/TS format (taken from a live feed or from a file), at different bitrates;
- Use a stream segmenter to generate short "chunks" of content – typically 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.

Another strength of HLS is its ability to implement adaptive bitrate streaming intelligently. Contrary to the techniques that are used in mobile RTP streaming, it is the end user device that decides the stream quality, according to the available bandwidth (and not the video server). This approach aims to ensure unbroken video streaming, thus creating a positive user experience through an unmanaged network:

- Index file is generated indicating different profiles (streaming qualities) available for one channel/content file;
- The receiving device (PC, mobile, STB) looks for the most suitable bitrate based on how long it takes to receive a chunk file;
- Each chunk file lasts 10 seconds, so the receiving device can automatically adapt the streaming with flexibility – in this case, every 10 seconds.

Ecosystem



HLS is natively supported by Apple devices that use iOS 3.0 and above: iPhone, iPad and iPod, as well as Macintosh computers running MacOS® X Snow Leopard. The video client used is QuickTime® X player, developed by Apple for their products. Here's the catch: even if Apple decided to port some of their software to a Windows-based PC platform, iTunes or Safari for example, they haven't ported QuickTime X Player to a Windows-based PC platform yet. As a result, there is no "official" client for the HLS streaming system, except on devices manufactured by Apple.

However, since the principle behind HLS is fairly simple, designing a client software for HLS streaming is quite straightforward. Verimatrix, Widevine, NDS, Latens and SecureMedia are a few examples of DRM companies that provide a solution for Windows-based PC platforms so they can play media streamed from an HLS server – as a point of interest, they also integrated their own DRM in the system. If we take a closer look at DRM on HLS, the HLS RFC describes how to scramble a HLS stream (128 bit AES algorithm) but not how to get the keys from the DRM key server. This implies that the DRM implementations are usually not compatible.

But HLS clients aren't limited to the Apple, PC and mobile markets: they are also gaining traction among set-top-box (STB) manufactures. Airties, Netgem and Amino, for example, already provide STBs capable of playing media streamed from an HLS server. Anevia's ViaMotion Origin server is able to stream to HLS compliant STBs; especially hybrid STBs that combine support for IPTV and OTT TV, or even DVB and OTT TV, in a single box. Many other devices are expected to appear on the market - Apple wants HLS to become an industry standard and has submitted a draft to the IETF in order for it to become a RFC, called 'HTTP Live Streaming'.

Advantages

- As of September 2011, Apple has sold over 130 million iPhones, 60 million iPod Touch players, and more than 30 million iPads, even though the product was only released last year. The potential audience for HLS streaming is therefore huge, particularly for portable devices. And predictions for 2012 sales figures are dizzying, as some analysts are predicting that up to 100 million iPhone 5s and 60 million iPads will be sold over the coming year. The iPad's success is particularly impressive as in 2011 it representing more than 80% of the whole tablet market.
- It provides a simple and efficient adaptive bitrate solution to cope with the fact that bandwidth is not managed on open networks.
- It is easy to integrate at the reception device level and can therefore be deployed on a wide range of set-top-boxes and devices. The fact that the H.264 codec was chosen implies that many chip manufacturers can provide H.264 hardware decoders as of today: demand on CPU power and on mobile battery power is low.
- It is based on Transport Stream transmission technology, making it easy to integrate into the existing digital TV world. A lot of IPTV DRM providers have already adopted this standard.

Limitations

- The adaptive bitrate solution is located solely in the device client. This "democratic" approach could hinder some uses in the professional / corporate world where administrators may wish to fine-tune the available video quality for certain specific content.
- Why didn't Apple propose its own client software for Windows PC? There is no native support in the major web browsers, and the lack of plugins to simplify integration makes it difficult to use as a web TV standard.
- Being limited to the MPEG standards means that the providers of HLS-compatible equipment may be liable to pay a license fee to MPEG LA. While it is no problem for Apple to pay an additional few cents in licensing fees when they sell iPhones, iPods or iPads that cost several hundreds of dollars, it may be a significant barrier for players in the "free and open-source" culture – namely the web browser providers who do not sell their products. Mozilla Foundation or Opera Software would not be very pleased to pay royalties to MPEG LA every time their browser software is downloaded. Recently, MPEG LA chose to soften their politics regarding free content over the Internet, but their global position remains unclear.
- DRM encryption is done through the encryption of the entire chunks. The transport layer is thus also encrypted, which is a barrier for some features, such as dynamic trick modes.
- Currently, there is no way to provide more than one audio track on an HLS stream. iOS5 should provide an "alternate audio track feature", but the number of available tracks will be limited to two, whereas there is an unlimited number of tracks available on Smooth Streaming or MPEG-DASH.

Viewpoint!

Jean-Baptiste Kempf, VideoLAN / VLC

Jean-Baptiste Kempf is Chairman of the VideoLAN project. He joined VLC as a developer in 2006, when he was a student at the Ecole Centrale de Paris.

What is the status of HLS support in VLC?

As there were no free HLS players for PC platforms, Anevia decided to sponsor and participate in developing HLS support for the VLC media player. The development project was led by Jean-Paul Saman, one of VLC's main contributors. Nonius Software, a middleware solution provider and an Anevia partner, also sponsored the project.

Does VLC intend to support other OTT streaming standards in addition to HLS?

VLC is a free and open-source multimedia player and toolbox. As such, it is designed to play the most common multimedia streams on a network (as well as local files, disks, etc.). The more we support, therefore, the better. So, once we have finished the project to support HLS, we intend to begin support of WebM. The VLC team is also closely monitoring Smooth Streaming, but as the industry is not currently showing much interest, there is less urgency to support it.

How do you see the development of streaming on PCs in the coming years?

The PC as a platform for watching TV and movies is doomed! Most recent industry innovations have focused on the mobile and tablet platforms. These platforms are significantly changing the way we create multimedia software because the decoding must be done on the DSP and GPU for performance reasons. This also makes it more expensive to develop and harder to debug.

VLC is already ported on Windows CE, Maemo and iOS (iPad and iPhone). We aim to be able to port it on Android and MeeGo very soon.

Google WebM

History

WebM was announced in May 2010, during Google I/O 2010. Promotion of the technology made it clear that WebM aimed to provide an OTT solution that would be royalty-free and usable on an open basis by all Internet companies and communities. In order to achieve this, Google decided to provide its VP8 video codec under a BSD license.

The chosen audio codec is Vorbis, and the container is based on a profile from Matroska. Although Vorbis and Matroska (.mkv) are already known and used in some products in the digital media industry, VP8 is a brand new codec, originally developed by On2 Technologies, before it was acquired by Google in early 2010.

A few days after its announcement, the WebM format was already supported by more than forty software publishers and hardware vendors, including ARM, Intel, Mozilla Foundation and Opera Software.

Principle

The philosophy of WebM is different from the other OTT techniques. It does not require segmentation of the media into chunks, because with WebM, one media stream is seen as one file.

To stream WebM from a live or VOD file:

- Encode the video and audio content in VP8 and Vorbis respectively, in different bitrates;
- Mux them into a WebM file, which must be automatically refreshed in real time if you plan to do live streaming;
- Use a HTTP server to deliver the WebM file.

Note that there is an added complexity to live TV streaming, where the muxing must be done constantly, and the resulting file will never be the same over time. As a result, caching on WebM is much more difficult than with chunks of video files. This makes it hard to integrate added-value features such as trick play, playlist or circular buffer. However, it's not impossible – Anevia has succeeded in developing a full-featured OTT solution based on WebM that can stream both live TV content as well as video on demand. In its favor, WebM has the advantage that it can be used directly as a storage format.

The adaptive bitrate process is also very different from the other OTT solutions because it is the server that chooses the audio/video streaming bitrate before muxing (multiplexing). The server has an output buffer in which it pushes all the packets ready to be sent. As it sends the content of the buffer to the network, it detects if there is enough bandwidth to reach the client. If not, it scales down to a lower bitrate.

Ecosystem

WebM appears to be chosen and driven by the Internet-centered community. It is supported by almost any web browser that can run on Windows, MacOS X or Linux OS. Either WebM is natively supported in modern browsers, starting with Mozilla Firefox 4, Opera 10.60, Chrome / Chromium in their latest versions, or it can be embedded in other web browsers through plug-ins. Thus, Microsoft Internet Explorer 9, Apple Safari (desktop version) and any Linux browser connected to GStreamer multimedia framework can play WebM streamed media.

Moreover, WebM has a tight link to the new HTML5 <video> tag, because the VP8 and H.264 codecs are both supported by HTML5 standards. VP8 is free, and as you know, Internet communities love anything that is free. So, in the future, we may well have VP8 and H.264 coexisting in the <video> tag, as nowadays we have JPEG and PNG coexisting in the <img> tag.

On the set-top-box side, we cannot overlook the buzz generated by the highly anticipated Google TV. In an approach which aims to be much more than a simple, direct response to Apple TV, Google has been experimenting new partnership proposals in order to push their TV ecosystem. As a result, Sony has already launched its first HD TV incorporating the Google TV platform and Logitech announced its first Google TV "Revue" STB. Up until now, those devices were pricey and not very successful. In August 2011 Logitech announced a massive price drop for its Google TV "Revue" STB with from $299 to $99. This aggressive price tag puts the Revue as a direct competitor to Apple TV STB.

Last but not least, regarding mobile streaming, Android started supporting WebM on its 2.3.3 version in February 2011. The fact that Android has been embraced by many handset manufacturers, such as Motorola, HTC, Samsung and Sony-Ericsson among others, could form the path to the success of WebM in the mobile world. August 2011 was an intense month for Google, as they bought Motorola Mobility, manufacturer of the praised Xoom Android tablets, but also a major player in the STB business, which could change the perspectives for GoogleTV.

Advantages

- The choice of Matroska for the container profile is interesting: Matroska uses EBML, a binary format derivative of XML. It allows the capabilities of the container to be extended without breaking the compatibility with older parsers. For example, it already includes a menu system (similar to the "chapters" of a DVD) and supports multiple audio and video tracks with labels attached to them, clean 3D handling, closed captions and subtitles, etc. Moreover, Matroska uses less bandwidth than TS encapsulation – this is an important point, especially for mobile devices.
- Native playback for three of the main web browsers is also an advantage. If you have ever tried to watch a Flash-based video clip using Linux OS, you should know that plugins are not necessarily very stable.
- The adaptive bitrate is managed by the server, but there are tricks that can enable the client software to ask the server to switch to higher or lower bitrate.

Limitations

- The lack of chips widely available for VP8 hardware decoding is a major drawback for WebM. When compared to the support of H.264, which has become very common on medium to high-range mobile devices, it is a good illustration of the difficulty in imposing a new codec to the industry. Rockchip was the first manufacturer of a chip able to do 1080p VP8 decoding in early 2011. They were followed in mid-2011 by ZiiLabs and Nvidia with its Tegra 2 chip, capable of encoding and decoding VP8 in addition to H.264. Nvidia Tegra 2 is becoming more and popular among Tablet and Smartphones manufacturers, especially those based on Android OS.
- In addition to the codec issue, and contrary to HLS, there is also the fact that the STBs have to support the Matroska container. While it's not a significant issue for players from the web environment, it may become one for those who come from the satellite receiver domain, which more traditionally supports TS. However, this could be balanced by the fact that some STBs are powered by an Opera browser, which promotes the WebM standard (even though Opera on STBs only currently supports the H.264 codec).
- There is also an issue with WebM caching, which can be tricky. It can only be done using dedicated IP streaming servers. It can't be done using web caches.

- Officially, WebM does not mention DRM systems. However Matroska can support encryption very easily, even if DRM-enabled .mkv files are not very common yet. As Widevine was acquired by Google, we may bet that a DRM for WebM will be available soon, as they already have a working solution for Apple HLS.

Viewpoint!

Steve Lhomme, Matroska

Steve Lhomme is the creator of the Matroska container, and Chairman of the Matroska Foundation. Prior to that, for three years, he was one of the main developers for the mobile and desktop CorePlayer.

What impact has Google's adoption of WebM had on Matroska?

It took us by surprise at first, but it was very exciting. Matroska has always been thought of as a one size fits all container for local storage, file streaming and even live streaming. But it is hard to advertise a format when you are a tiny group of people against companies like Apple, Microsoft, Real Networks and DivX. We already converted DivX to the quality levels of Matroska. Google could have gone with other containers or even created their own, but they were really convinced by the technical quality of our format.

What is the technical challenge for a container?

Unfortunately audio/video containers are usually the less considered element of the multimedia toolbox. They are usually built with the goal of making a new custom codec work. But they are not meant to be used with all codecs in all possible configurations. They are only extensible to a certain point and usually carry heavy legacy from the time they were designed. Matroska was designed to look forward 10 years to make sure it would still be relevant, and still allow all the existing systems to work in Matroska. It is not perfect, but it certainly fits the bill.

What about the choice of the VP8 codec?

VP8 really resulted from the necessity for Mozilla and Opera to provide video in their browsers, but they could not afford to pay the MPEG LA license fees. Google mostly uses H264, but it also realizes there has to be another choice. In the long term it makes as much sense and maybe even more to use VP8 because it's cheaper and just as good as H.264. In the end more creative use and customization may appear from this freedom. And it will surely be favored by any startup that wants their product to be cost competitive. H.264 currently has overwhelming momentum, but I think little by little VP8 will make significant ground on the web. A key factor will be when hardware decoders appear.

Microsoft Smooth Streaming

History

Smooth Streaming is the streaming protocol provided as an extension of Silverlight 3.0, and its specifications were published by Microsoft in September 2009. Video streaming was a major advance in Silverlight's capabilities, and the team in charge of designing Smooth Streaming was formed from engineers who had already worked on the software side of Zune.

Concerning HD content delivery for the masses, Microsoft created a stir in September 2010 when Stephen McGill, head of the British Xbox division, was reported as saying: "People have moved through from DVDs to digital streaming, so we can offer full HD 1080p Blu-ray quality streaming instantly, no download, no delay. So, who needs Blu-ray?"

Microsoft chose to support the H.264 and AAC codecs. This choice was driven by the objective to propose easy hardware decoding in order to achieve HD (720p/1080p) streaming on web applications. However, Smooth Streaming also supports Microsoft proprietary codec WMA as well as the SMPTE standard VC-1, or any other codec supported by the 3GP container format.

Pre-versions of Smooth Streaming were tested on a large scale before its official release. It was used, for example, for the web broadcasting of Michael Jackson's funeral and for the Roland Garros tennis tournament in HD.

Principle

Smooth Streaming is based on fragmented files, with a PIFF (Protected Interoperable File Format) container, which is extended from 3gp format, and an underlying SSTP (Smooth Streaming Transport Protocol) layer.

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.

The client software starts by requesting a manifest from the server. The manifest response from the server lists the available media, tracks and bitrates. The client then asks for one or more fragments corresponding to the requirements on the list to be sent by the streaming server over HTTP. Like HLS, when using Smooth Streaming it is the client that manages the choice of adaptive bitrate.

Finally, it should also be noted that DRM are particularly well integrated into Smooth Streaming, with the possibility to use several DRM layers in the same file.

Ecosystem

The Smooth Streaming ecosystem is managed by Microsoft. There is no dedicated certification program, and any encoder, distribution server or client is able to integrate into their Smooth Streaming ecosystem. However, for the origin streaming server, it seems that Microsoft has only authorized certification of its own IIS 7.0 server.

This appears to be an odd move from Microsoft, because the specifications for the Smooth Streaming protocols are open and publicly available. Some streaming server providers, such as Anevia, can provide streaming servers that are fully compliant with the published specifications from Microsoft. All in all, the integration process should be no more trouble than for any other interoperability and inter-working project.

On the mobile side, there is a lot at stake for Microsoft with the release of Windows Phone 7. It's clearly their last hope to get back into the mobile and smartphone industry and take market share from the well established Apple iOS and Google Android platforms. It was no surprise when Microsoft announced that their new mobile OS will support Silverlight Smooth Streaming. However, it was much less clear if they plan to support any other OTT streaming systems. Moreover, Microsoft wants its OTT protocol to be broadly adopted on other mobile platforms. As a result, in 2011, they started to provide a Smooth Streaming client library that is available for iOS and Android.

Advantages

- Smooth Streaming natively supports multiple audio tracks, and multiple subtitles in the stream. This gives the possibility to provide the same viewing experience as a digital broadcast (ie DVB-T or DVB-S TV).
- Contrary to the other OTT systems, DRM is already well integrated in Smooth Streaming. Of course, Microsoft has its PlayReady solution, which is widely adopted, but the specifications published by

Microsoft are precise enough to have DRM vendors also propose their own DRM solution for Smooth Streaming.

- In the web streaming field, Microsoft Smooth Streaming has acquired a strong image gained from successful live HD (720p/1080p) streaming of major events. The video quality of the live broadcasting was praised, compared to that provided by Flash-based sites at the same time. It's worth putting the emphasis on the fact that Microsoft Silverlight player analyses the decoding capacities of the client before choosing a high video quality, in order to ensure a smooth rendering.
- Something that could be considered an advantage as well as a limitation is the fact that Microsoft specifications are very detailed, and illustrated with many examples. It makes it much easier to fully understand the way Smooth Streaming is meant to be working, but is also longer to implement (see below).

Limitations

- As mentioned in the previous point, Microsoft specifications are very detailed, so solution support takes much more time to implement. The result is that Smooth Streaming may not be adopted as fast as HLS, which was designed as an easy derivation from the TS standard.
- Although Smooth Streaming was clearly designed for web streaming, the player is always managed through the Silverlight plugin. Even Microsoft's Internet Explorer requires the external plugin. Like HLS, Smooth Streaming is also limited by the fact that it is impossible to manage bandwidth from a centralized point on the network: everything is decided solely by the device clients.
- Finally, Smooth Streaming is based on patented audio and video codecs, so its use may be subject to license fees, payable to MPEG LA or Microsoft.

Adobe HTTP Dynamic Streaming

History

If we look at the total amount of video streamed over the Internet since the beginning of this century, Adobe is clearly the main contributor with the incredible number of Flash-based video clips available on YouTube, Dailymotion, Hulu and hundreds of specialized video sites. However, most of those video clips are only recorded clips in .FLV format (no live streaming), unprotected by any DRM, and obviously not implementing adaptive bitrate mechanisms.

Adobe Flash (formerly Macromedia Flash) was introduced back in 1996, and according to Adobe statistics from 2010, Flash players reach 98% of all US web users, and 99.3% penetration of PC and Mac desktop computers. However the first version of Flash capable of playing video clips was version 6, released in March 2002, and the real OTT answer from Adobe to Apple and Microsoft came with HTTP Dynamic Streaming (HDS), included in release 10.1, in June 2010. This version filled the gap for video streaming capabilities, enabling adaptive bitrate mechanisms, live streaming and the use of Adobe's own content protection and DRM system (Flash Access).

Principle

On a technical level, the principle behind HDS is fairly similar to Microsoft Smooth Streaming. 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.

The supported codecs for HDS are H.264 and VP6 for video, and AAC or MP3 for audio tracks. The .f4m manifest file contains a "bootstrap", which is equivalent to the initialization segment of MPEG-DASH (see corresponding chapter), and helps the video player know what segment of which fragment to start with, based on the current time of video playback. In this manifest file, HDS also indicates if multi-bitrate is available so that the player can determine which is the best content for playback.

The video player is written in ActionScript, and can run through Flash Plugin (version 10.1 and above) or Adobe AIR framework. Adobe provides a reference video player, named Open Source Media Framework (OSMF) and for a positive experience, there are already a bunch of flash / javascript video players available on the Internet built upon it, which are fully customizable and ready to play HDS video streams.

As far as content protection is concerned, the only DRM system integrated with HDS is Adobe's own system: Flash Access Server. It is a feature-full scrambling system, giving the administrator the choice of very interesting policies for access control, and which is able to encrypt transport layer as well as the video chunks for maximum protection. The drawback of this double encryption system is the CPU required, which could prevent it to work on low-end STB and devices.

Advantages

- Adobe succeeded in imposing its framework as a mandatory plug-in for a rich web experience very early in Internet history. And, as the Flash plug-in has nowadays the ability of auto-update itself, there are all chances any modern PC can play any Adobe HDS stream.
- The manifest files provide an interesting memory and network footprint, as they are by default zipped and base64-encoded.
- 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 (for example, how to build a descrambler over the player available inside the Flash plug-in).

Limitations

- We will not cover the ongoing battle between Apple and Adobe resulting in no support for Flash on Apple devices. But even for Android tablets and smartphones, Flash support is very dependent on the specific Android version, and even for a given Android version, its Flash support may depend on the hardware platform vendor. However, we have had some positive experience in playing Flash streams out of an Anevia ViaMotion server on a RIM BlackBerry smartphone, a BlackBerry PlayBook or an HP WebOS Touchpad tablets. Even if their sales are lagging behind the iPad, those devices are unable to play any other OTT protocol for the moment.
- The only DRM supported by HDS is Flash Access, provided by Adobe's Flash Access Server. Contrary to the documentation that they provide for VOD applications, Adobe gives very little information about the way their DRM is working, especially for live applications. Finding any solid information requires to get through their commercial support.
- As for Apple HLS, there is currently no way to provide more than one audio track on an HDS stream. There are beta versions providing an "alternate audio track feature", but only for VOD files, not for live streaming.

MPEG-DASH

History

As we have seen in the previous chapters, the OTT technology market remains a battlefield with no clear winner at the time of writing. The four major technology contenders are all coming from the Internet market, without no historical roots in the broadcast and telecommunications industries.

The 3GPP and OIPF published their own specifications for adaptive streaming protocols in 2010; Adaptive HTTP Streaming (AHS, part of 3GPP R9) and HTTP Adaptive Streaming (HAS), but adoption of these protocols has yet to take off. The MPEG-LA and ISO organizations have taken the lead by creating a single cross-industry standard to unify adaptive streaming over HTTP protocols: MPEG-DASH (Dynamic Adaptive Streaming over HTTP). The first DASH draft specification was published in February 2011.

Principle

The technical principle is very similar to the previous OTT techniques. 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)
  - MPEG-2 Transport Stream (as in HLS)
  - 3GP container.

Due to popular demand, MPEG-DASH introduces an optional feature: the initialization segment. An initialization segment is in MPEG2-TS format and contains a single program, its program specific information (PAT, PMT) and some optional codec and DRM specific information. The advantage is that there is no need to duplicate this information in the subsequent media segments. This allows for fast channel switching which contributes to an improved quality of experience similar to traditional TV.

Standardizing streaming is only a part of the problem. MPEG-DASH comes with a sibling project addressing DRM interoperability.

This issue is currently tackled by UltraViolet, pushed by a 70+ member consortium named Digital Entertainment Content Ecosystem (DECE). DECE is made up of majors Hollywood studios, consumer electronics manufacturers, DRM vendors, software and network providers, and they aimed at creating a cloud-based Digital Rights Management system that would enable the end-users to share purchased digital contents between all the devices they own. "Buy once, play it anywhere". Most big digital media ecosystem companies are taking part including Sony, Microsoft, Google / Widevine, Adobe, Cisco, HP, IBM, Intel, Motorola Mobility (now Google), Samsung, LG Electronics and studios such as Warner Bros, NBC Universal, Fox, Paramount, Sony Pictures and Lionsgate. However, we should not overlook the fact that both Apple and Disney do not support DECE nor Ultraviolet.

The principle behind UltraViolet is that a user registers an account from UltraViolet, and can link up to 12 household devices for streaming purchased media. Those devices may include connected TVs, tablets, smartphones, PC, and game consoles. Up to three streams may be transmitted simultaneously to the registered devices. When a user starts watching a video stream on the device, there is a check of the granted digital rights from the device to UltraViolet cloud. Note that only the rights are stored on the cloud, the media assets and streaming servers are usually stored and streamed from another are on the Internet.

UltraViolet relies on the Common File Format (CFF), which is based upon ISO base media file format (MPEG-4 part 12), as Microsoft PIFF is –there are many points in common between those formats. This common file format is chosen in order to play on all UltraViolet players, and compatible with all DECE-supported DRM systems. The chosen container format is fragmented MPEG4, and the encryption algorithm fixed to AES scrambling, with the DRM-specific information located in the header. Five DRM technologies have been chosen to be compatible with UltraViolet. Those DRM technologies have proven track records of successful deployments on STB, PC, smartphones, tablets or connected gaming consoles. These DRMs are:

- Microsoft PlayReady
- Google Widevine
- Adobe Flash Access 2.0
- Marlin
- OMA CMLA-OMA v2

Advantages

- From a purely technical point of view, MPEG-DASH looks like the “best of breed” of all OTT techniques. It took the best ideas of the older technologies, and while remaining mostly compatible.
- As a standard pushed by the ISO and MPEG-LA, it can gain a certain form of legitimacy for telcos, as it better designed to meet their specific industry practices than the web and software-driven OTT technologies.
- UltraViolet is a smart response to the “buy once, play anywhere” problem. Only a very few users are Apple or Android enthusiasts who keep the same ecosystem for all their devices. Most of them live in a multi-vendor world and own, for example, an Android-powered Samsung smartphone, an iPad tablet, and a STB provided by their telecom operator; they remain clueless as to the DRM and streaming protocols used by those devices; they just want it to work.

Limitations

- From the client side, there are very few players that fully implement MPEG-DASH specifications (a preliminary work is done on VLC, with simple live 3GP profile). Things may change fast, as most of UltraViolet DRM vendors announce their players. Besides, at the time of writing, the MPEG-DASH specifications are still in draft mode.
- Apple and the Walt Disney Company are two major absentees to the UltraViolet consortium. Steve Jobs, the former CEO and co-founder of Apple, is also a Walt Disney board member and private share-holder. UltraViolet is a potential threat to the iTunes / iPhone / iPad / AppleTV DRM system that allows users to watch the same video stream on many devices as long as those are Apple devices. Therefore ignoring Apple and Disney within a media consortium is a risky bet.

In a nutshell, which OTT solution is the best?

Obviously, there is no clear cut and definitive answer to that question; all three approaches have their advantages and drawbacks. The real question is: **What are you going to use OTT for?**

Do you plan to launch an iPhone app? Or a corporate web site? Or a mobile service? Do you want to include connected TVs? Or are you planning a new service with a dedicated set top box, created from scratch?

These questions should be the starting point for your choice of solution or solutions. Next you will need to choose a DRM, which may depend on the content provider. How much caching capabilities will be required? Do you have to integrate any specific encoders, or a specific head-end? All of this can be summed up in one question: **Into which ecosystem do you want to integrate your service?**

Of course, we all want to know which technology will dominate the future of OTT (whether it’s the best is another matter!). However, it’s very difficult to predict a winner since each of the OTT technologies presented in this paper is promoted by a major IT or telecommunications industry player and therefore are each creating a big impact on the market. At the time of this writing, HLS seemed to be the current industry leader, driven in particular by the amazing commercial success of Apple devices. But things could change rapidly: in Q2 2010, the sales of Google Android-powered devices superseded those from Apple, for the first time.

The quality and quantity of a TV service proved to be a competitive advantage when network operators launched multi-play services over their managed IPTV networks. Today, OTT streaming represents an important opportunity for many new actors, enabling them to promote innovative services without having to deploy complex network infrastructures. But the questions raised above must be addressed before deciding on an OTT solution.

About the Author

Lionel Bringuier
CTO, Anevia

Lionel Bringuier has over 10 years software engineering experience in the Telecommunications industry managing real time and mission critical services. Prior to joining Anevia he held senior positions at Matra Nortel Communications (now EADS Telecom), Netcentrex/Comverse and SFR where he was director of IMS-based triple play deployments. As CTO at Anevia, he is responsible for guiding the company in its technical direction, for managing cross-product architecture, standards initiatives and R&D resources. He is a graduate in engineering from the Ecole Centrale de Nantes.

About Anevia

Anevia provides video solutions and service infrastructure for the delivery of live TV and video on demand (VOD) services to TV, PC, Internet-connected and mobile devices.

With over 1,000 deployments in 70 countries, representing several millions of users and over 25,000 live channels, Anevia is a reference in solution delivery to telecom operators, broadcast service providers and the hospitality market.

Anevia’s solution portfolio includes DVB to IP gateways, content distribution, network management, time-shifting applications and video servers. The 3Screens™ platform delivers services across mobile phone, TV and PC including VOD, network personal video recorder, Catch-up TV, Start Over, and Pause TV.

For more information, please visit: http://www.anevia.com

Keywords: **Over The Top TV (OTT), streaming over IP, origin server, IPTV, multiscreen, Internet TV, Connected TV, mobile TV, HTTP Live Streaming (HLS), Silverlight Smooth Streaming, WebM, MPEG-DASH, Adobe HDS (HTTP Dynamic Streaming), 3Screens convergence.**

Anevia: 1 rue René Anjolvy, 94250 Gentilly, France, Tel: +33 (0)1 41 98 32 40