MPEG-DASH and Hybridcast

Hiroshi Fujisawa, Internet Service Systems Research Division

Twenty-four broadcasting stations have started offering Hybridcast services, and approximately 4 million Hybridcast receivers have been sold since its launch in September 2013. Hybridcast’s application engine is an open HTML5 platform that reduces the costs of developing TV receivers and applications and makes it easy for the server and TV receiver to work together in performing complex application processes. These features greatly contribute enhancements of the functions of the integrated broadcast-broadband services. Moreover, the addition of a Video-On-Demand (VOD) standard that supports MPEG-DASH (Dynamic Adaptive Streaming over HTTP) has opened up possibilities for new services that harmonize services across broadcasting and broadband. This article describes how MPEG-DASH is handled in the Hybridcast Technical Specification Version 2 published by IPTV Forum Japan. It also discusses the use of MPEG-DASH media players and other applications involving MPEG-DASH in Hybridcast services.

1. Introduction

IPTV Forum Japan, an organization that promotes the standardization of technical specifications for open IPTV*1 services and next-generation smart TV*2 services, released version 2 of the Hybridcast technical specifications in June 2014*1,2). The new specifications enhance the functionality of Hybridcast by providing support for viewing recorded content and compatibility with third-party*3 applications, called “non-broadcast-oriented managed” applications*4. Likewise, the new specifications clearly define compatibility with the HTML5 <video> element as an extension of VOD functions.

The <video> element is one of the most popular new elements of HTML5, and it provides the ability to directly control playback of videos on the web browser. In previous HTML versions, video playback required downloading separate software with a player function and embedding it into the web browser as a plug-in (software for adding functions). A typical example of such software is Adobe Flash. The demand for playing video on smartphones and other mobile devices, however, revealed bottlenecks between plug-ins and browsers that increase the CPU processing burden and consumption of battery power. This situation is what lead to the development of the <video> element for directly playing video within the browser.

Video streaming using Hypertext Transfer Protocol (HTTP) rapidly deploying in addition to using the video dedicated delivery protocols (such as Real Time Streaming Protocol (RTSP)*3), Real Time Messaging Protocol (RTMP)*4, and Microsoft Media Server Protocol (MMSP)*5). Apple HTTP Live Streaming (HLS), Adobe HTTP Dynamic Streaming (HDS), Microsoft Smooth Streaming, and MPEG-DASH are examples of HTTP streaming systems. HTTP Streaming delivers video content using the web servers and conventional content delivery network (CDN) architectures for large-scale content distribution. It thereby

---

*1 Services that enable viewing movies and programs, anytime, anywhere, by connecting the TV to the Internet.

*2 Smart TVs are TVs and set-top boxes that enable the use of applications for IPTV services and games. Next-generation smart TVs are smart TVs that support integrated broadcast-broadband services and can be used in new ways.

*3 Applications that are validated through means other than broadcast signals (e.g. application signatures)

*4 Operators other than broadcasters who provide a wide array of services (application developers, receiver manufacturers, etc.)
facilitates the construction of an environment for delivery of videos at a low cost. In particular, MPEG-DASH has shown its potential for broadcasting; it has been incorporated in the European DVB\textsuperscript{6} and HbbTV\textsuperscript{7} standards and is being used in BBC’s iPlayer, YouTube, Netflix, and other popular video delivery services for TV.

In this article, we describe the VOD protocol (incorporating MPEG-DASH and HTML5 \texttt{<video>} element) in version 2 of the Hybridcast technical specifications published by IPTV Forum Japan. We also describe a media player supporting this VOD protocol and give examples of its use in Hybridcast services.

2. Overview of MPEG-DASH technology

The MPEG-DASH video-delivery protocol is an international standard published by ISO/IEC (International Organization for Standardization/International Electrotechnical Commission; 23009-1). In this section, we give a brief explanation of its operation.

As shown in Figure 1, the video service provider generates streams from audio-visual footage and partitions the streams sequentially into files (segments) lasting a few seconds. The provider then creates Media Presentation Description (MDP) files, which describe the video encoding information (encoding method, bit rate, etc.) and segment configuration, and sends them together with the segments to the web server for delivery.

Media devices communicate with the web servers via the Internet by using HTTP for transferring video as well as image, text, and other general web content. The media devices use the MPD files to determine the correct segmentation with which to play out the video.

MPEG-DASH does not require dedicated equipment for video delivery. Moreover, it enables adaptive streaming, wherein the quality of video displayed on the device is chosen in response to fluctuations in the underlying network bandwidth conditions. As shown in Figure 2, multiple streams, each having a particular quality level (resolution, bit rate, etc.), are generated from a single piece of video footage and are divided into segments. MPD files containing video quality data on all the segments are then generated and sent to the media device. The media device uses the information contained in the MPD files to select and play the segments of the highest quality that can be currently supported by the network and viewing environment.

3. IPTV Forum Japan (IPTVFJ) MPEG-DASH Profile

In December 2015, IPTV Forum Japan published
Hybridcast Operation Standards⁶. This document includes a profile⁵ for operating VOD services based on MPEG-DASH. It also gives operation standards for application programming interfaces (API)⁶ conforming to the Media Source Extensions/Encrypted Media Extensions (MSE/EME) standard, which is currently going through the World Wide Web Consortium (W3C) recommendation process. MSE/EME is a standard that provides an API for describing the playback client (the software needed for playing video on the receiver) in JavaScript. MSE is a standard that provides an API for controlling video playback, while EME is a standard that provides an API for handling Digital Rights Management (DRM) schemes used for protecting content. The use of these APIs eliminates the need for embedding a complex playback control component for MPEG-DASH in the receiver, by letting an HTML5-ready web browser (hereinafter, HTML5 browser) handle the process. For example, the need for a playback client that can adaptively switch video streams according to the state of the network can be addressed by developing an HTML5 application that is capable of adaptively switching segments for playback using an MSE API. The MSE/EME API not only reduces the costs of developing receivers; it enables functional extensions for new services to be developed on the new applications by the service providers rather than having to rely on receiver manufacturers to update their equipment.

This profile supports H.264/ MPEG-4 Advanced Video Coding (AVC) or H.265/ MPEG-H High Efficiency Video Coding (HEVC), and up to 4K resolution. In addition, it has provisions regarding MPD, subtitles, digital rights management (DRM), etc.

4. Media player and playback control

In the IPTVFJ MPEG-DASH profile, media device
functions required by video service providers are implemented through a playback client (i.e., media player) that operates on media player JavaScript library including MSE/EME for HTML5 browsers. A popular media player library that works in this way is dash.js\(^{10}\) of the DASH Industry Forum. Although it is easy to play MPEG-DASH content on PC web browsers, it is harder to playback such content on TVs, which have less memory and CPU resources compared with PCs.

To create a media player that meets the requirements of Hybridcast on TV services, NHK developed a JavaScript library that has playback control functions (hereinafter called the playback control function library)\(^{11}\). This library was built using the MSE/EME, and it enables control functions based on MPD files, such as for adaptively switching streams depending on the status of the network or for carrying out DRM processing. These new functions are added to the basic video control functions of the <video> element, such as playback and pause. Figure 3 shows the relationship between the media player and the playback control function library. The API provided with the playback control function library can be used to add user interfaces (UI) and functions (service control) suited to particular services and develop appropriate media players for them. One feature of this library is that it allows playback on TV. Since it can be operated on any HTML5-ready device, a playback environment can easily be created on smartphones, tablets, and PCs.

5. Usage example of playback control function library on Hybridcast

The playback control function library developed by NHK supports adaptive bitrate playback for switching bit rates on the basis of the state of the network. It also enables a variety of functions in combination with Hybridcast and HTML5. Moreover, since these functions make it possible to develop media players directly on the basis of the service provider’s requirements, commercial broadcasters, receiver manufacturers, and video content providers can all come up with a wide range of applications on their own or together.

This section describes the characteristic functions of the playback control function library as well as examples of its use in Hybridcast services.

5.1 Ensuring safety and security

A Hybridcast television can receive broadcast signals simultaneously with MPEG-DASH Internet video. For example, if an emergency occurs while the TV is showing VOD content, the TV will also receive emergency notifications through broadcast (Figure 4).
5.2 Support for subtitles

The library can be combined with various other libraries. For example, creative subtitle services can be provided by combining it with one for displaying broadcast subtitles created with ARIB Timed Text Markup Language (TTML) on an HTML5 browser. Figure 5 shows an example of a subtitle service that combines this library with a TTML subtitle player library developed by NHK.

5.3 High-definition video playback

Since the library uses memory efficiently, it can play high-resolution video on TV without any problems. Figure 6 shows an experiment conducted on December 11, 2015 in which on-air, 4K video was delivered through Hybridcast and shown on a media player that was developed on the basis of our library by Fuji Television Network, Inc. The player had a 4K video playback function and a function for simultaneously receiving VOD and broadcast signals (Section 5.1). It could receive 4K video from the Internet at the same time as it received 2K broadcast programs.

5.4 Inserting video using multi-period function

The library also supports a “multi-period” function for switching and playing video delivered from different sources. This function makes it possible to implement video insertion technologies for targeted advertising wherein commercials are displayed in accordance with the viewer’s attributes and preferences. Figure 7 shows an example of a media player incorporating the multi-period function that was developed by Tokyo Broadcasting System.
Television, Inc. The player is compliant with the Video Ad-Serving Template (VAST)*7 and Digital Video Multiple Ad Playlist (VMAP)*8 standards for video advertisement delivery developed by the Interactive Advertising Bureau, an interactive advertising organization in the U.S. In this example, on-air video is converted into VOD at near-real-time speed, and commercials are selected and inserted depending on the viewer’s attributes as the video is played.

6. Summary

This article described MPEG-DASH, which is part of IPTV Forum’s Hybridcast Technical Specifications.
Version 2. Hybridcast’s use of (open-platform) HTML5 and the release of IPTVFJ MPEG-DASH profile, an open technology standard for video delivery, have widened the possibilities for developing new television functions through software. Furthermore, with the introduction of the MPEG-DASH playback control function library developed by NHK, high-quality and high-performance video delivery services for television have become as accessible as the technologies for creating webpages. This development will pave the way for many service providers and developers to create new video services that use both broadcast and broadband.

We would like to thank Tokyo Broadcasting System Television, Inc. and Fuji Television Network, Inc. for their cooperation in writing this article.

References
1) IPTVFJ STD-0010 Integrated Broadcast-Broadband System Specification
2) IPTVFJ STD-0011 HTML5 Browser Specification
6) ETSI TS 103 285, “MPEG-DASH Profile for Transport of ISO BMFF Based DVB Services over IP Based Networks”
7) ETSI TS 102 796, “Hybrid Broadcast Broadband TV”
9) IPTV-FJ STD-0013 Hybridcast Operation Standards