mobitv
CONNECTED MEDIA SOLUTIONS

Media Delivery Solution Overview

MobiTV

April 2015

mobitv
CONNECTED MEDIA SOLUTIONS

Contents

List of Figures

1 Foreword

1.1 About this Document

This document describes MobiTV's Media Delivery solution, which includes a comprehensive system for the segmentation, encryption, and distribution of live and recorded video.

1.2 About MobiTV, Inc.

MobiTV is a global leader in delivering live and on-demand video to any screen, connecting media reliably and securely anytime, anywhere, on any device. The company's end-to-end platform delivers a true TV everywhere experience that helps service providers reduce time to market and control costs associated with the deployment and operation of high concurrency, multiple platform services. MobiTV's connected media solutions solve the complexity of delivering video across networks, operating systems and devices while managing associated rights. The company efficiently optimizes for network conditions to deliver multiscreen media services that center on empowering the viewer. MobiTV's connected media solutions are tailored for wireless operators, as well as broadband providers and other OTT providers. MobiTV powers mobile and multiscreen solutions for leading service providers including AT&T, Deutsche Telekom, SoftBank, Sprint, T-Mobile, US Cellular, and Verizon among others.

2 Media Distribution 3.0 Overview

The MobiTV Media Distribution 3.0 system delivers Live TV and VOD streaming media content to most screens in the connected media ecosystem by supporting multiple input and output formats. MobiTV supports media input from live encoders and VOD files. Media Distribution 3.0 supports the MPEG-DASH standard for digital media. Video and audio are stored in the MobiTV Common Storage Format (CSF), which is based upon the MPEG-DASH standard. From the CSF, the content is transmuxed into the output format requested by the viewing device. For DRM, the Common Encryption Format (CENC) is used to encrypt a single time and compatible DRM technologies are supported.

2.1 System Diagrams

This section contains two system diagrams, one for Live TV and one for VOD media distribution.

All media tracks (video, audio, subtitles) are stored as separate files, to allow the client device to switch between bitrates and languages. An XML-based file called the Media Presentation Description (MPD) provides a manifest for all content.

Note that the live encoders must be configured properly for multiple audio files.

Requests for live, VOD, or recorded media are initially sent to the Media Muxer, which retrieves the segment in Common Storage Format and converts it into the appropriate format for the device. For scalability, the converted segments are cached in Varnish in FMP4/DASH/HLS format after an initial request.

The Segmenters generate thumbnail images for use in the Visual Seek feature. The SpriteService combines several thumbnail images into one image to minimize traffic to the server.

The Stream Manager manages the maximum number of streams available to any one device. The maximum number of concurrent live streams is configured in the Service Policies. The maximum number of concurrent streams per a VOD asset is configured in the Content Policies. See the *Service Management Portal* datasheet for more information.

The Electronic Program Guide (EPG) is processed through an adapter and stored in a Solr database. It is served to a cache by the Guide Manager for access by the client device.

When the client device requests content, the CDN fetches and distributes both the MPDs and the media segments. The CDN first checks its own cache. If the requested item is not found in the CDN cache, then it is fetched from the Media Muxer.

The CDN distributes and caches the data following the HTTP 1.1 specification. To view the specification, access http://www.w3.org/Protocols/rfc2616/rfc2616.html.

2.1.1 Live TV Distribution

Following is a high-level view of the major components in the MobiTV Media Distribution system for delivering Live TV content:

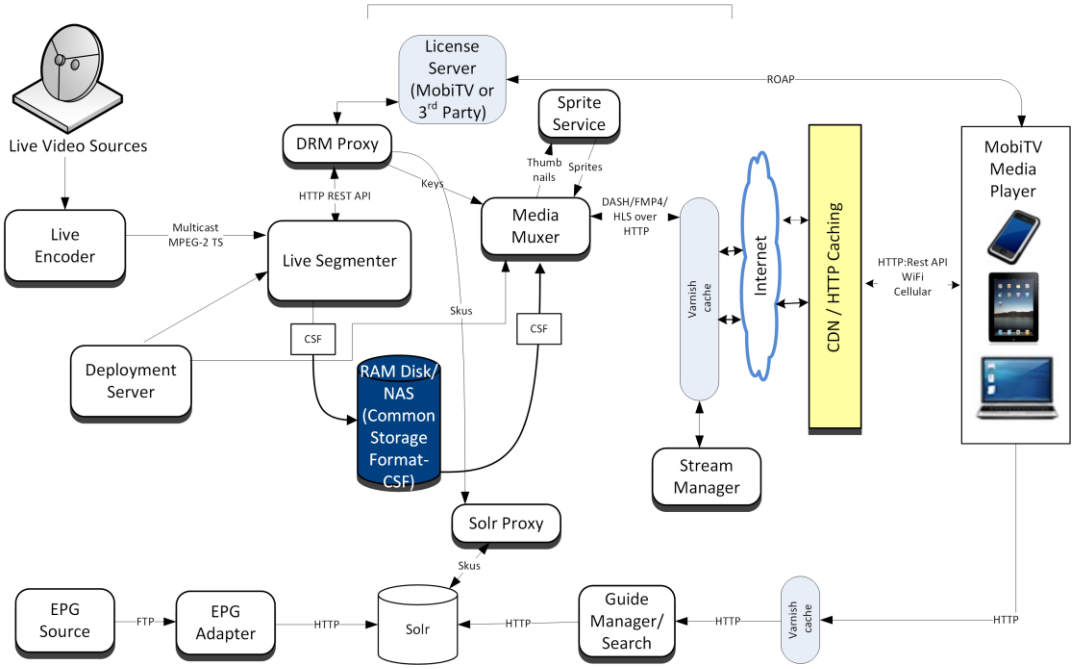


Figure 1: MobiTV Live TV Distribution System

The Live Segmenters receive live video from the Live Encoders for processing into segments. Following is an overview of the Live TV media delivery process:

1. Live Encoder receives one live video stream encodes it into multiple MPEG-2 Transport Streams with varying bitrates and resolutions.
2. The MobiTV Live Segmenter reads the MPEG-2 TS over multicast and produces MPEG-DASH segments, thumbnails, and the manifest file (Media Presentation Description or MPD) for video and audio in a range of bitrates and resolutions. The configuration for the Segmenter is supplied by the Deployment Server. The Live Segmenter may be configured to produce HLS directly.
3. The segments are stored in the RAM in the Common Storage Format (CSF), a format based upon MPEG-DASH. There is an option to produce HLS segments directly on the Segmenter.
4. If required, channels are encrypted using Common Encryption Format (CENC), also an MPEG-DASH standard.

5. When a channel is requested to be played by the client device, the CDN checks in the Varnish cache for segments. If not found in the cache, the Media Muxer pulls the segments and MPD from RAM storage. The segments are transmuxed by the Media Muxer into the output format required by the device. Transmuxing is not required for HLS media produced by the Segmenter, but the Live Segmenter must be configured in the Deployment Server for this option.
6. The DRM Proxy provides decryption and encryption keys to the Media Muxer. Licenses are issued by the License Server. The DRM Proxy retrieves the product SKUs from Solr.
7. Streaming begins if user rights and number of concurrent streams are accepted. Segments are cached for future use.
8. Thumbnails are generated and grouped into sprites by the SpriteService. The sprites are served to the client for use in Visual Seek.

2.1.2 VOD Distribution

For VOD content, Content Partners provide encoded VOD content over FTP, HTTP, or RSS feeds. The preferred formats for VOD content files are mp4 and 3gp. These files are transcoded into MPEG-4 format by the Media Transcoding Manager and then processed into segments by the VOD Segmenter. The content is also processed by the MobiTV Content Management System. After the segments and MPD are generated by the VOD Segmenter, the flow is the same as for live content.

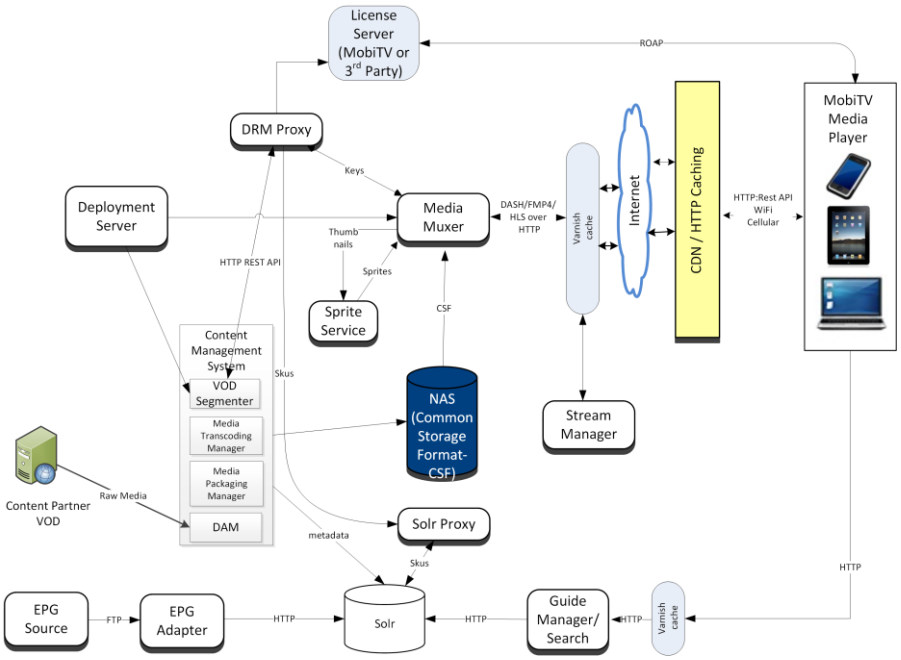


Figure 2: MobiTV VOD Distribution System

2.1.3 Media Playback

The media playback process is similar for live content, VOD, and recordings, as shown below:

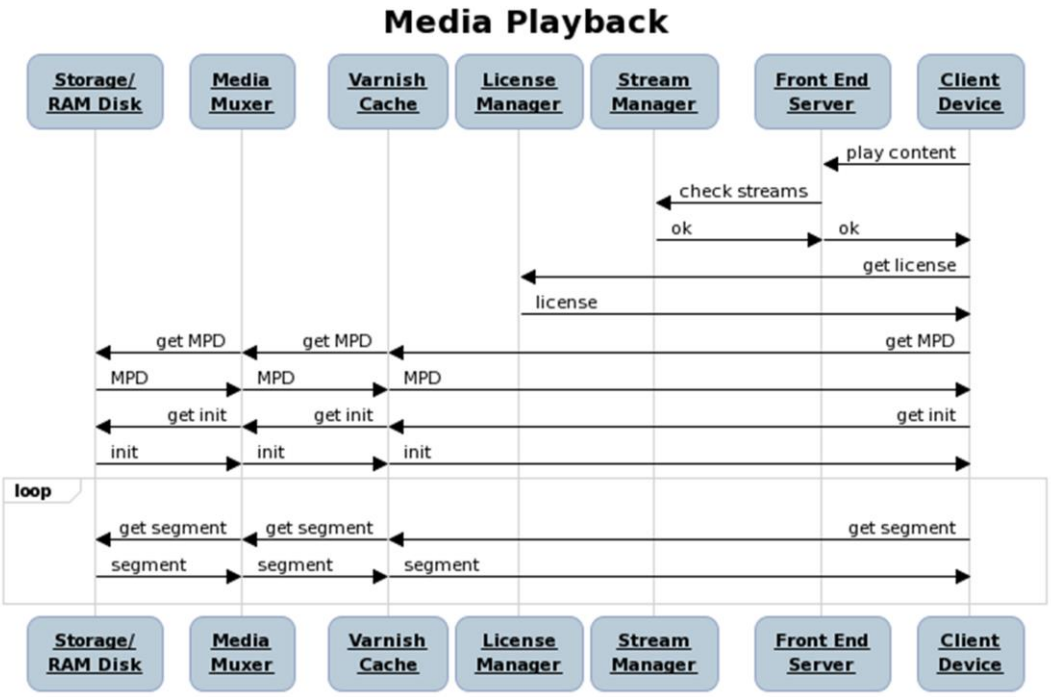


Figure 3: Media Playback for Live/VOD/Recordings

The first step in media playback is the Stream Manager confirms that the maximum stream limit has not been reached and the client device is approved for playback. Then the client device obtains a license in order to play encrypted content. The client device obtains the MPD, a manifest of all content. The MPD directs the client to the *init* segment to begin playback, with subsequent segments following automatically.

2.2 Supported Input Formats

For Live TV, the supported input format from the live encoders is MPEG-2 TS.

For VOD, the preferred file formats are mp4, 3gp, and mov.

2.3 Supported Output Formats

Following are the output formats, client devices and DRM technologies supported by MobiTV Media Distribution 3.0:

Output Format	Supported Client Devices	Supported DRM Technology
MobiTV FMP4	Android Set Top Box MobiTV Connect HDMI Accessory	MobiTV DRM
MPEG-DASH	Android Set Top Box MobiTV Connect HDMI Accessory	MobiTV DRM, Widevine
HLS	iOS	MobiTV DRM, Apple HLS Encryption

2.3.1 MobiTV FMP4

MobiTV Fragmented MP4 is MobiTV 's proprietary media format. It is based on the MPEG-4 (part 12) standard file format, with proprietary methods developed by MobiTV for network distribution. MobiTV's fragmented MP4 format is designed to be transmitted over standard HTTP/TCP protocols to produce a stateless video distribution method. Fragmented MP4 works together with the MobiTV media player or native decoders to perform many functions, including: bandwidth adaptation, policy enforcement for media delivery, integration with DRM libraries, and content decryption.

MobiTV's fragmented MP4 format is based on the industry standard ISO/IEC 14496-12:2005, Base Media File Format specification. This specification for coding audio/visual objects describes smaller, fragmented files with H.264 video codec support. The H.264 codec is a widely-used industry standard for video compression. It provides optimal video quality at substantially lower bit rates than previous standards. MobiTV currently supports Baseline, Main, and High compression profiles of the H.264 encoding standard.

FMP4 streams are generated by the Media Muxer from segments stored in Common Storage Format.

2.3.2 MPEG-DASH

MPEG-DASH (DASH) is a streaming technology designed by the Moving Pictures Expert Group. It is intended to replace the multiple commercial formats currently used with a standard format. For more information on the DASH standard, see http://dashif.org/mpeg-dash/

DASH segments require no transmuxing by the Media Muxer. The Common Storage Format is very similar to DASH.

The benefits of DASH include the following:

- A standard protocol enables operators to encode and store content only once, thus decreasing the cost of production, encoding, storage, and transport of digital media.
- With support for multiple DRM systems, operators are not locked into a single provider.
- It was designed to represent a superset of all existing solutions.
- It is vendor-neutral, allowing content publishers to generate a single set of files for encoding and streaming that should be compatible with as many devices as possible,
- DASH is attracting mobile operators for deploying eMBMS for mobile TV broadcast over LTE.

DASH provides many capabilities above other formats, such as:

- Common encryption and multiple DRM support. The content can be encrypted once and delivered to client using various DRM schemes. Operators are freed from reliance on a single DRM provider.
- Support for multiple language tracks for a video stream
- Compact manifest. The segments' address URLs can be signaled using a template scheme resulting in a compact MPD.
- Support for subtitles

2.3.3 HLS

Apple's HLS uses manifest files, similar to the DASH MPD, to indicate the order in which segments are played. The HLS manifest files are called the m3u8 file.

The main content of the m3u8 file is an explicit list of URLs to segments that should be played after each other. The duration of the segment files is recorded in the m3u8 file, which makes seek possible. For VOD, there is a marker at the end indicating that the video ends there. For Live, there is no such marker, and the client must repeatedly ask for new versions of the m3u8 file.

For HLS Live, there is a configuration option to produce the media segments directly with the Live Segmenter, or with the Media Muxer. This configuration is managed using the Deployment Server. The Live Segmenter will produce the HLS segments and corresponding segment manifests. In this case, the Media Muxer does not transmux the HLS segments.

For HLS Live, the Media Muxer generates the required m3u8 files from the DASH MPD when playback is requested. The Media Muxer also retrieves authentication keys and sprites for Visual Seek.

HLS v5 adds functionality to provide alternate audio/video tracks.

3 Key Features of MobiTV Media Distribution 3.0

This section describes the most important features of MobiTV Media Distribution 3.0.

3.1 Common Storage Format

All media files are stored on disk using ISO Base File Format (ISO BMFF) following the guidelines in the DASH264 specification. This is known as the Common Storage Format (CSF).

The Common Storage Format supports the following features:

- Efficient handling of multiple audio tracks (for multiple languages) by storing video and audio separately
- Comprehensive set of variant profiles for adaptive bitrate streaming
- Support for Visual Seek FFW/RW with keyframe thumbnail images in a live stream
- Text-based multiple language subtitling and closed captioning
- Rich subtitling including full character set and layout
- Encrypted or non-encrypted storage
- Support for Common Encryption Format, with support for multiple DRM technologies
- On-the-fly conversion to HLS, DASH-ISOFF, and MobiTV FMP4
- Support for H.264
- Support for audio formats: AAC-LC, HE-AAC v1, HE-AAC v2, AC3

The content is stored as non-encrypted or encrypted as required. DRM information is inserted into encrypted segments, in compliance with the Common Encryption standard.

Additional files such as the Media Presentation Description (MPD) manifest file and subtitle files are stored in the filesystem along with the media segments. The Common Storage Format segments are converted by the Media Muxer into the formats required by a client device upon request for playback, and then cached.

Note that HLS for Live TV can optionally be generated by the Live Segmenter, and the Media Muxer would not convert the segments.

3.2 Common Encryption Format

Common Encryption Format (CENC) is an MPEG standard for encrypting media contents. CENC supports a multi-DRM solution, where content is encrypted once with the content encryption key and one out of many DRM systems can be used by the client device to receive the decryption key. Thus a single encrypted file can be decrypted by one or more DRM technologies.

Media Distribution 3.0 supports the following DRM systems:

- MobiTV DRM
- Widevine
- Apple HLS Encryption

A component called the DRM Proxy communicates with external DRM systems and provides a caching proxy between the DRM License Server and the Media Muxer and Segmenters.

All audio samples are encrypted, but only a selection of video frames is encrypted.

3.2.1 Encryption

Encryption is one of the tasks performed by the Segmenters in conjunction with the License Manager. To provide encryption, the Segmenters generate a series of rotating keys used to obtain corresponding Short Term Key Messages (STKMs) from the License Manager, via the DRM Proxy. The STKMs are included in the media segments.

MobiTV supports industry standard encryption building blocks such as:

- AES encryption, which is certified by the NSA for use in some Type 1 products (the highest level of certification possible) AES encrypts the content.
- RSA (2048-bit) encryption, which is commonly used in many commercial products today. RSA encrypts the content encryption keys.
- SHA-256 (one-way message hash) validates the integrity and the authenticity of the license.

For live streams, encryption is configured per channel. Unencrypted content is not available on the channels configured for encryption. For VOD, encryption is configured per file or per content provider account.

3.3 Media Presentation Description (MPD) Manifests

DASH uses an XML manifest file called the Media Presentation Description (MPD) to describe the available content.

The MPD contains information about the media segments, including the URL addresses, byte-ranges, different bitrates and resolutions (“profiles”), and encryption method. The MPD is client-agnostic and runs across screens and platforms. It provides sufficient information for a DASH player to download the preferred resolution and bitrate, user preferences (for example, audio language), and network bandwidth.

To play the content, the MobiTV Media Player on the client device first obtains the MPD. By parsing the MPD, the client device learns about the program timing, media-content availability, media types, resolutions, minimum and maximum bandwidths, and the existence of various encoded alternatives of multimedia components, accessibility features and required digital rights management (DRM), media-component locations on the network, and other content characteristics. Using this information, the client device selects the appropriate encoded alternative and starts streaming the content by fetching the segments using HTTP GET requests.

The manifest file comes in two forms:

- Dynamic, for Live TV. Dynamic MPDs represent a selection of segments that are available at any point in time. Some segments will be added over time and some will be removed over time in a sliding window of availability. Note that ongoing catchup programs (start time has passed but not the end time of a program) use dynamic MPDs.
- Static, for VOD content. Static MPDs represent a fixed set of media segments, and all the media segments are available. Note that completed Catchup programs (start time and end time have passed) use static MPDs.

The Live and VOD Segmenters generate one MPD called Manifest.mpd when the media segments are generated. This manifest contains all available audio/video tracks for a particular channel/asset. The Media Muxer uses the Manifest.mpd to generate the multi-bitrate manifests configured in the Deployment Server. The Media Muxer generates the MPDs required for media playback when the client device requests content.

3.3.1 VOD MPD

The VOD MPD contains the following elements:

- type: static: list of segments is constant; fixed segments
- availabilityStartTime
- timeShiftBufferDepth
- minimumUpdatePeriod
- startNumber
- mediaPresentationDuration: length of program
- publishTime: time of creation

For HLS delivery, the Media Muxer generates m3u8 playlists from the MPD.

Sample MPD for VOD

```
<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
           xmlns:mobitv="urn:mobitv"
           xmlns="urn:mpeg:DASH:schema:MPD:2011"
           xsi:schemaLocation="urn:mpeg:DASH:schema:MPD:2011 DASH-MPD.xsd"
           profiles="urn:mpeg:dash:profile:isoff-live:2011,urn:com:dashif:dash264"
           maxSegmentDuration="PT6S"
           minBufferTime="PT2S"
           type="static"
           mediaPresentationDuration="PT21M27S">
<Period>
      <AdaptationSet contentType="audio" mimeType="audio/mp4" lang="eng"
segmentAlignment="true" startWithSAP="1">
         <SegmentTemplate initialization="$RepresentationID$/init.mp4"
startNumber="1" media="$RepresentationID$/$Number$.m4s" duration="6"/>
         <Representation id="A6" codecs="mp4a.40.2" bandwidth="64000"
audioSamplingRate="48000" />
         <Representation id="A9" codecs="mp4a.40.2" bandwidth="96000"
audioSamplingRate="48000" />
      </AdaptationSet>
      <AdaptationSet contentType="video" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" par="16:9" minWidth="640" maxWidth="640"
minHeight="360" maxHeight="360" maxFrameRate="30">
         <SegmentTemplate initialization="$RepresentationID$/init.mp4"
startNumber="1" media="$RepresentationID$/$Number$.m4s" duration="6"/>
         <Representation id="V6" codecs="avc1.4d001f" bandwidth="600000"
width="640" height="360" frameRate="30" sar="1:1"/>
         <Representation id="V9" codecs="avc1.4d001f" bandwidth="900000"
width="640" height="360" frameRate="30" sar="1:1"/>
      </AdaptationSet>
   </Period>
</MPD>
```

3.3.2 Live MPD

The Live MPD contains the following elements:

- type: dynamic: list of segments changes every 6 seconds
- availabilityStartTime, for example: 1970-01-01 00:00:00
- timeShiftBufferDepth
- minimumUpdatePeriod, for example: 1 year
- startNumber
- mediaPresentationDuration
- publishTime: time for last written segment when MPD is requested

Sample MPD for Live

```
<?xml version="1.0" encoding="utf-8"?>
<MPD xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xmlns:mobitv="urn:mobitv"
```

```
            xmlns="urn:mpeg:DASH:schema:MPD:2011"
            xsi:schemaLocation="urn:mpeg:DASH:schema:MPD:2011 DASH-MPD.xsd"
            profiles="urn:mpeg:dash:profile:isoff-live:2011,urn:com:dashif:dash264"
            maxSegmentDuration="PT6S"
            minBufferTime="PT2S"
            type="dynamic"
            timeShiftBufferDepth="PT1M"
            minimumUpdatePeriod="PT8760H"
            availabilityStartTime="1970-01-01T00:00:00Z">
<Period>
      <AdaptationSet contentType="audio" mimeType="audio/mp4" lang="eng"
segmentAlignment="true" startWithSAP="1">
         <SegmentTemplate initialization="$RepresentationID$/init.mp4"
startNumber="0" media="$RepresentationID$/$Number$.m4s" duration="6"/>
         <Representation id="A9" codecs="mp4a.40.2" bandwidth="96000"
audioSamplingRate="48000” />
         <Representation id="A6" codecs="mp4a.40.2" bandwidth="64000"
audioSamplingRate="48000” />
      </AdaptationSet>
      <AdaptationSet contentType="video" mimeType="video/mp4"
segmentAlignment="true" startWithSAP="1" par="16:9" minWidth="640" maxWidth="640"
minHeight="360" maxHeight="360" maxFrameRate="30">
         <SegmentTemplate initialization="$RepresentationID$/init.mp4"
startNumber="0" media="$RepresentationID$/$Number$.m4s" duration="6"/>
         <Representation id="V9" codecs="avc1.4d001f" bandwidth="900000"
width="640" height="360" frameRate="30" sar="1:1"/>
         <Representation id="V6" codecs="avc1.4d001f" bandwidth="600000"
width="640" height="360" frameRate="30" sar="1:1"/>
      </AdaptationSet>
   </Period>
</MPD>
```

3.4 Closed Captions (both Live and VOD)

For both VOD and Live, Closed Captions are supported in the format CEA608/708. If contained in the H264 video stream as SEI NAL Units, the Closed Captions are passed through and processed as part of the stream.

For VOD content the system supports converting SAMI, SCC, SRT, TTML or WebVTT input subtitling formats to WebVTT or SMTE Timed Text (TTML) output format. The system also supports ingesting SCC input into the Closed Caption output stream.

3.4.1 Closed Caption Formats

The CEA 606/708 standards supplied by the FCC dictate the requirements for Closed Captions.

CEA 608 allowed for basic formatting and colors. CEA 708 requires more extensive formatting options, including user-selected formats.

For this release, <italic> and <bold> input format tags are converted to the supported output format. All other tags in the input subtitling files will be removed.

3.5 Subtitles (VOD)

Subtitles are supported in VOD when supplied as separate files.

For VOD content, consumers select a subtitle language from a set of available languages. The subtitles in the selected language are loaded from an external file into the MobiTV Content Management System (CMS), processed into TTML and WebVTT by the VOD Segmenter and rendered on the client device.

The supported input formats for processing by the VOD Segmenter are the following:

- SAMI
- SRT
- TTML
- WebVTT (for HLS devices) – required for advanced formatting features
- SMPTE-TT (for DASH devices) – required for advanced formatting features

Note that the WebVTT and SMPTE-TT input formats are required to include advanced formatting supported by the CEA 708 specification.

The CMS provides the following output formats:

- TTML for DASH devices
- WebVTT for HLS devices
- FMP4 subtitles are injected into the stream in CEA608 format

3.6 Multiscreen Delivery

Multiscreen playback preserves user authentication, authorization rights, and DRM licenses when transferring to another registered device or a companion device.

Companion devices may be used to control playback (similar to a remote control) on a set top box or HDMI Accessory.

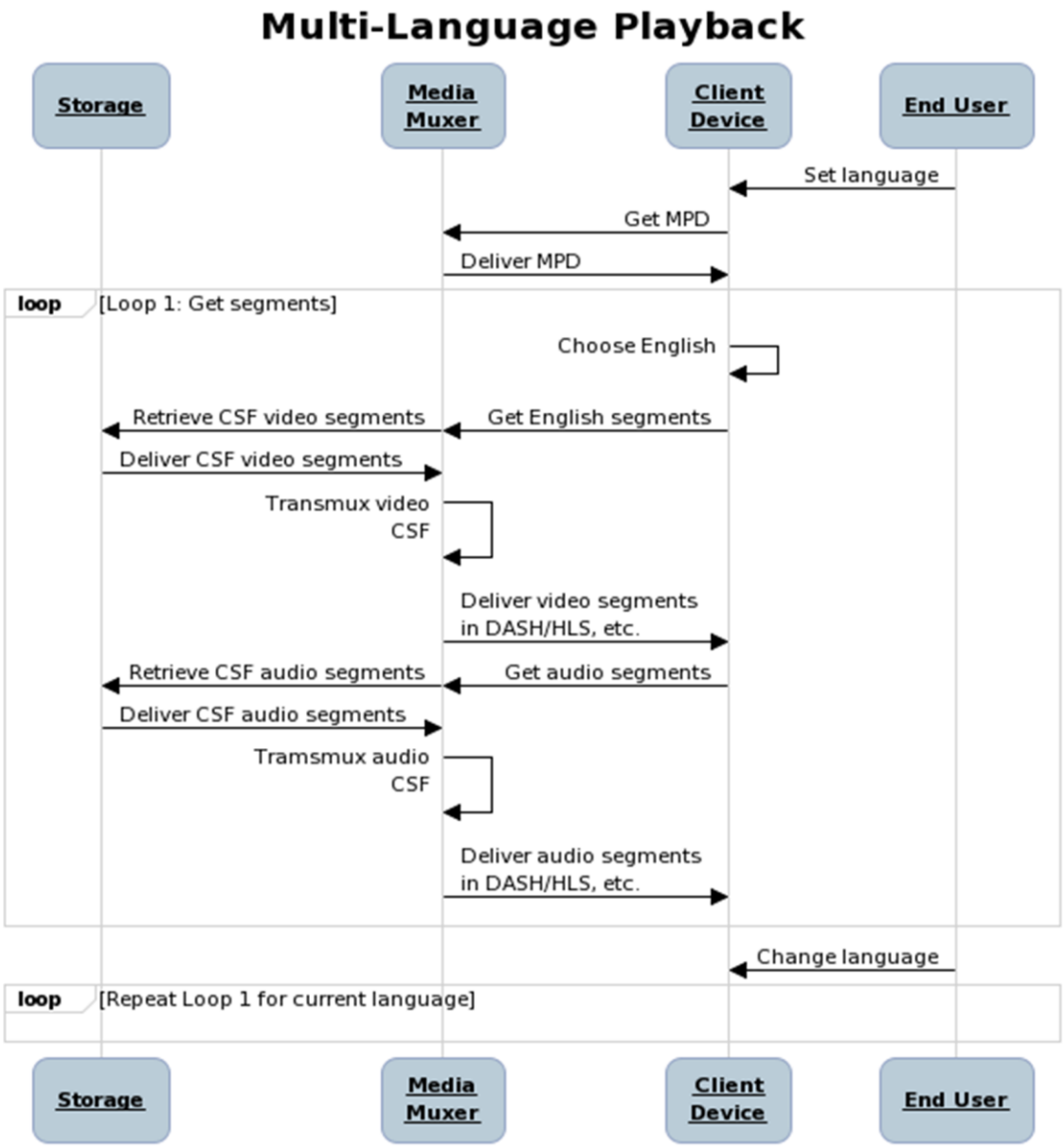
3.7 Multiple Audio Tracks and Languages

With the Common Storage Format, VOD and Live content is transcoded with separate video and audio tracks. To accommodate multiple languages per video track, multiple audio

tracks will be generated for a single video segment if necessary. This feature allows multiple languages per video track, as each language is processed into a separate audio track. Video can be played in the preferred language as set by the user on a client device.

For Live content, if the source output from the live encoder includes multiple audio tracks, the Live Segmenter will save the audio tracks separately, and include all of them in the MPD for distribution with the video stream.

For VOD content, the audio tracks may be provided as separate files, or embedded in the



VOD stream and processed into separate files by the VOD Segmenter.Following is the media playback sequence for multiple languages:

3.8 Adaptive Streaming

DASH is a "dynamic adaptive streaming" technology. This means that the media content is transcoded into small segments using a series of "profiles" containing multiple bitrates and resolutions. The client device selects which bitrate stream to play and then adjusts over time as network conditions fluctuate. For example, the client initially requests a lower bit-rate when starting a stream, but switches to a higher bitrate if network conditions allow for a greater download speed. Lower bitrates are delivered during periods of network congestion or when switching to lower bandwidth networks (for example, from WiFi to 3G). Adaptive streaming delivers the highest quality video for available bandwidth on the wireless network.

3.9 TimeshiftTV- CatchupTV

TimeshiftTV and CatchupTV features are optional MobiTV Media Distribution services. Both allow the end user to watch Live TV content after its original broadcast. TimeshiftTV allows the end user to manipulate playback of an ongoing live program. CatchupTV generates recordings of live programs during their initial broadcast and makes them available for playback on an on-demand basis for a configurable time period (for example, two weeks). CatchupTV creation can be configured on a per-channel basis, and may be enabled/disabled on a per-program basis.

Refer to the document *TimeshiftTV-CatchupTV Overview* for a detailed description of the TimeshiftTV-CatchupTV feature.

3.9.1 Media Controls

Media control actions preserve DRM licenses when resuming after a pause to keep content secure. The trick play media controls described below are supported by the TimeshiftTV feature. The following media controls are available:

- Pause and resume – Consumers may pause a video on one device and resume on that same device or at the same point on another registered device.
- Start Over – Consumers may go back to the beginning of live program.
- Visual Seek within Live or VOD stream– A consumer may rewind/seek within a live broadcast. Thumbnail images are generated to select viewing point.

3.9.2 Visual Seek

Visual Seek displays thumbnail images on the touchscreen client device, allowing the user to move forward or backward in the video stream by selecting an image. Visual Seek gives the end user full control to explore the scenes from a video stream and to select a starting point for their viewing.

The Visual Seek user interface is presented as an overlay and is available in several styles, including Bubble, Filmstrip, and Grid.

4 Key Components of MobiTV Media Distribution 3.0

4.1 Media Muxer

The Media Muxer is a streaming server that handles playback requests from the client device and converts segments from the Common Storage Format to the output streaming format compatible with the client device. See *Supported Output Formats* for a list of supported output formats.

The Media Muxer generates and serves the MPDs to the CDN and the cache. It also requests encryption keys from the License Manager through the DRM Proxy.

4.1.1 Media Muxer Configuration

The Media Muxer receives its configuration from the Deployment Server, including the following parameters:

- Content types (live, VOD, recording) and location (full path) on the NAS
- URLs for License Manager, Deployment Server, and the DRM Proxy (if encryption is required)

4.1.2 Transmuxing with the Media Muxer

The Media Muxer converts segments stored in the Common Storage Format to a format compatible with the client device (FMP4, DASH, or HLS). All content types (live, VOD, and recordings) are accessed through the Media Muxer. Note that the Catchup TV feature requires a Media Muxer.

When the client requests a segment or manifest, the Media Muxer determines what output format is needed. For HLS or other formats, the Media Muxer converts the CSF files to the requested format on the fly. (Note that here is a configurable option to generate HLS segments for Live TV in the Live Segmenter. In this case, the Media Muxer would not be involved in transmuxing the CSF segments into HLS segments.)

Responses from the Media Muxer can be cached in any standard CDN. The Media Muxer adds cache headers where appropriate.

The Media Muxer communicates with the DRM License Managers through a DRM Proxy. The DRM Proxy hides the differences in the License Manager APIs when operating in a multi-DRM environment.

4.1.3 Codecs

Media Muxer supports the following standard codecs:

- H264 (baseline, main and high profile) as video codec
- AAC Low Complexity (LC) as audio codec
- Closed Captioning in the format CEA608/708 contained in the H264 video stream as SEI NAL Units.
- Subtitle format (VOD): TTML and WebVTT

4.1.4 Solr

A Solr database provides the Media Muxer with information to verify which content is mapped to a service in the MobiTV Service Management Portal. Since the service SKUs are not stored in the content segments, the verification is performed by the Media Muxer when a playback request is received.

4.2 Sprite Server

The Sprite Server combines a series of thumbnails into one image to minimize network traffic when fetching images for the Visual Seek interface.

The Segmenters generate thumbnail images from the i-frames of segmented H264 media files. FFmpeg converts the thumbnails to JPEG. The Sprite Server merges a group of thumbnail images into a single JPEG file. Sprite Server is a standard Java Web App running on Tomcat, and communicates with the Media Muxer.

Following is an overview of the Visual Seek technology:

- Available for both live, catchup, and VOD content
- Configured in the Deployment Server
- Width and height of thumbnail images are configurable
- Segmenters create thumbnails from the i-frames
- An open source multimedia transcoder application, FFMpeg, converts H264 i-frames into JPEG for thumbnail images
- VOD Segmenter uses the highest bitrate track as the source for the thumbnails

4.3 Live Segmenter

The Live Segmenter generates Live segments with the accompanying MPD manifest. The Live Segmenter ingests content from the Live Encoders and produces segments in all variants (variants are “recipes” of bitrates and resolution).

The Live Segmenter creates several files during the segmentation process:

- The Manifest (.mpd),
- Initialization segment (.mp4)
- Media segments (.m4s)

- Thumbnails

The Manifest is updated for each written segment to ensure that new clients accessing the MPD obtain the latest written segment.

During segmentation, the Live (and VOD) Segmenters generate one MPD called Manifest.mpd. This manifest contains all available audio/video tracks for this particular channel (or asset). Manifest.mpd is then used as a base by the Media Muxer to generate all the other multi-bitrate manifests that are configured in the Deployment Server. This MPD generation in the Media Muxer happens in response to playback requests.

The Live Segmenter reads MPEG2-TS streams over multicast and produces MPEG-DASH segments for storage in Common Storage Format. The segmentation follows the guidelines in the DASH264 specification. This means that all media tracks (video, audio, subtitling) are stored as separate files.

Note that live encoders require a specific configuration to handle multiple audio tracks.

For encrypted channels, the Live Segmenter obtains a license from the License Manager by querying the DRMProxy for the appropriate DRM technology. Channels are encrypted using Common Encryption Format (CENC), also an MPEG standard. The Live Segmenter inserts a Short Term Key Message (STKM) into the Protection Scheme Specific Header (PSSH) box, as described in the Common Encryption specification (ISO.IEC 23001-7).

The Live Segmenter supports the H264 (baseline, main, and high profile) video codec and AAC Low Complexity (LC) as the audio codec. Closed Captioning in the format CEA608/708 is passed through the Live Segmenter if it is contained in the H264 video stream as SEI NAL Units.

Live media segments are stored in the RAM on the Live Segmenter, in the Common Storage Format (CSF). MPDs are also stored in RAM.

The Live Segmenter has a configurable option to produce HLS segments directly.

4.3.1 Live Segmenter Configuration

The MobiTV Deployment Server manages the configuration of the Live Segmenters. Sample configuration parameters include channel information, where to read the multicast source content from the encoders, where to store the output segments, and encryption information.

4.3.2 Live Output URL Format

The base live URL format is defined below.

```
<output format>/live/<channel>/[<sku>/]
[<max_bitrate>/<max_width>/<max_height>/]<requested data>
```

```
<output format> = dash|hls3|hls5

<channel> = an identifier for the specific channel

<sku> = optional value defining the sku for encrypted content

<max_bitrate>, <max_width> and <max_height> = optional but if included then all three
must be included. Valid values are integer value. Use 0 if not filtering on a specific
variable.

<requested data> is dependent on output format
```

4.4 VOD Segmenter

The VOD Segmenter reads ISO base media file format (MPEG-4 Part 12) files and produces MPEG-DASH segments for storage in Common Storage Format. VOD content is provided by Content Providers in mp4 or 3gp files and transcoded by the Media Transcoding Manager into MPEG-4 format.

VOD ingestion is similar to Live ingestion, except the content is uploaded as a file from a Content Provider, and completed segments are stored on a NAS.

The VOD Segmenter follows the guidelines in the DASH264 specification, as follows:

- Supports mp4, 3gp, and mov files as input format
- Supports H264 Baseline, Main and High profile as video codecs
- Supports AAC-LC as audio codec
- Stores media content according to MPEG-DASH using ISO-FF
- Stores each media track as separate files
- Creates a static Media Presentation Description (MPD) along with the media content

The VOD Segmenter works with the MobiTV Content Management System to store and manage VOD assets.

VOD Segmenter also reads subtitle files in a specified directory and creates DASH content. The VOD Segmenter supports SRT, WebVTT, SAMI, SCC and TTML as input formats and produces WebVTT (for HLS) and TTML (for DASH) outputs.

After the segments and MPD are produced, the Media Distribution follows the same process as for Live TV.

4.4.1 Subtitles (VOD Only)

For VOD content, consumers select a subtitle language from a set of available languages. Each language requires a separate subtitle file. The selected subtitles are loaded from an external file into the MobiTV Content Management System (CMS), processed into TTML and WebVTT, and rendered on the client device.

The supported input formats for processing by the CMS are the following:

- SAMI
- SRT
- TTML
- WebVTT (for HLS devices) – required for advanced formatting features
- SMPTE-TT (for DASH devices) – required for advanced formatting features

Note that the WebVTT and SMPTE-TT input formats are required to include advanced formatting supported by the CEA 708 specification.

The CMS provides the following output formats:

- TTML for DASH devices
- WebVTT for HLS devices
- FMP4 subtitles are injected into the stream in CEA608 format

4.4.2 VOD Output URL Format

The base VOD URL format is defined below.

```
<output format>/vod/<provider>/<asset>/[<sku>]
[<max_bitrate>/<max_width>/<max_height>/]<requested data>

<output format> = dash|hls3|hls5

<provider> = an identifier for the provider of this content as defined in CMS

<asset> = an identifier for the asset of this content as defined in CMS

<sku> = optional value defining the sku for encrypted content

<max_bitrate>, <max_width> and <max_height> = optional but if included then all three
must be included. Valid values are integer value. Use 0 if not filtering on a specific
variable.

<requested data> is dependent on output format
```

4.5 Deployment Server

The Deployment Server is a centralized server for managing configurations for producing the multi-bitrate profiles for Live and VOD content. It communicates with the Media Muxers and Segmenters to provide the configuration for both input and output media.

The Deployment Server is a standard J2EE web application deployed in a servlet container on Apache Tomcat. It provides an HTTP JSON REST interface. Configurations are stored in

a database and merged into a JSON formatted response for delivery to the Media Muxer and Segmenters. It includes a web-based User Interface for entering configurations.

Configurations are stored in a database (Oracle or MySQL) and merged into a JSON formatted response to satisfy client requests.

The Deployment Server configures generic global parameters for a deployment, including the following:

- Subtitle out formats
- Encryption mode
- Output directories for both Live and VOD
- Segment length
- Closed Caption filtering
- Key rotation interval
- Enabling thumbnails
- Setting buffer lengths for live and catchup TV
- Defining tracks (audio, video, and subtitles) and variants

The Deployment Server also configures the Media Muxer and Segmenters, as follows:

- The Media Muxer retrieves configurations for output formats (HLS, DASH, FMP4) for variant groups, track groups and other output-specific configurations.
- The Live Segmenter retrieves configurations for tracks, nodes, channels, channel groups, and channel templates.
- The VOD Segmenter retrieves configurations for tracks and output directory from the Deployment server database. Additional configurations for the VOD Segmenter are passed as command line switches.

Note: Refer to the *Deployment Server User Guide* for details on using the Deployment Server to configure deployments for Media Distribution 3.0 solutions.

4.6 Common Storage

All media files are stored internally in a common storage ISO based file format following the guidelines in the DASH264 specification.

The content is stored as non-encrypted or encrypted MPEG-DASH segment files. Audio and video content are stored in separate files.

The common storage also includes additional files such as the manifest file and subtitling information. If enabled, thumbnails will be stored together with the media content. Live

segments and thumbnails are typically stored in RAM while VOD segments are stored on shared storage (NAS). Catchup content is copied from RAM to the shared storage.

4.7 DRM Proxy

In a multi-DRM solution each specific License Manager will have a slightly different API. The DRM Proxy hides the differences in the License Manager APIs when deploying in a multi-DRM environment.

The DRM Proxy has two purposes:

1) Provide a unified API between the Media Muxer, Segmenters, and License Manager
2) Serve as a cache to store key information data to limit the number of requests to the License Managers.

The Segmenters communicate with the DRM Proxy to encrypt segments. The Media Muxer contacts the DRM Proxy to decrypt the segments and re-encrypt them using the appropriate DRM solution and format.

4.8 License Manager

The Media Distribution System interacts with the License Manager when creating encrypted content. The License Manager also needs a Rights Manager and an OCSPResponder to be able to create the correct content licenses and certificates.

5 CDN Integration

A CDN improves the scalability of the streaming servers. The CDN increases the speed of media delivery by caching results geographically closer to users.

5.1 CDN Integration Requirements

The CDN requires certain configurations to manage the cache correctly:

1. The CDN must ignore the query string in request URL when determining whether to cache fragments.
2. The CDN must include query strings in log events. This is necessary for effective reporting.
3. The CDN must be able to cache HTTP status code 503 server responses.
4. When the CDN serves a client a cached HTTP 503 response, the header must include how long the response has been cached. When client requests fragments that are not yet available, a 503 response is returned with both Cache-Control: max-age and Retry-After are set to when it is supposed to be available.
5. The HTTP standard cache header parameter "age" is used to determine the cumulative age of cached data.
6. Max-age is used for all normal responses (200 and 503) The CDN must adhere to the max-age attribute in the cache-control header. The CDN may discard packets before the max-age is reached, but max-age must not be exceeded.
7. The CDN must be able to interpret and express the max-age attribute as a numerical value.

5.2 Accessing Segments

For delivery using HTTP, each video and audio segment is assigned a unique URL with a specific format.

The request for content sends a URL request to the CDN or Media Muxer. The segments have a persistent URL so, when using a CDN, two separate devices watching the same media can be serviced efficiently from a cache. URL requests are compliant with CDN policies and all responses from MobiTV servers include applicable headers to optimize CDN caching performance.

5.3 Caching Requirements

To improve performance, data elements are cached on the client devices and the CDN.

Requests from clients to the Media Muxer can be cached in any standard behavior CDN. In order to utilize the caching functionality the Media Muxer adds cache headers where appropriate.

5.3.1 Cache-Control Header

- For all CDN cache durations, server will send the max-age value
- For resources that cannot be cached on the CDN, the server will send "private"
- For resources that can be cached on the CDN, the server will send "public"
- For non-cacheable resources the server will send no-store, no-cache, max-age=0, no-transform

5.3.2 CDN Cache Keys

The following CDN configurations must be made either based on the DNS name, or a URL pattern.

- The CDN cache key MUST include the URL PATH
- The CDN cache key MUST NOT include the URL QUERY STRING
- The CDN cache key SHOULD include the request Accept-Encoding header

6 Scalability

6.1 Media Muxer

The Media Muxer is a stateless service and instances are triggered by incoming client requests. This means the Media Muxer can easily be deployed in a cluster using a Load Balancer to distribute load over the servers. In case additional performance is needed just add more servers to the pool. The Media Muxer can be deployed internally near the origin sources and it can be deployed externally e.g. in a CDN. . For external deployment, HTTPS is used to communicate with the License Manager, for secure delivery of encryption keys. A Varnish cache is typically installed on the same node at the Media Muxer.

The Media Muxer is an Apache module running in an Apache server. If the Media Muxer crashes or exits unexpectedly, Apache will immediately spawn a new worker to take its place. Since the module is stateless, a restart of the application is sufficient to continue serving client requests.

6.2 Live Segmenters

A single Live Segmenter can process several channels. Multiple Live Segmenters are typically deployed to generate identical segments all the time. The redundant Segmenters work with a load balancer to ensure high availability. The Varnish cache accesses one Segmenter at a time in a round-robin fashion.

6.3 DRM Proxy

The DRM Proxy is a stateless service and instances are triggered by requests. This means the DRM Proxy can easily be deployed in a cluster using a Load Balancer to distribute load over the servers. In case additional performance is needed just add more servers to the pool. The DRM Proxy should be deployed close to the Media Muxer since there might be one call for each segment that needs to be decrypted/encrypted. Preferably the DRM Proxy should be installed on the same node as the Media Muxer.

6.4 Deployment Server

The Deployment Server is stateless which means in order to achieve High Availability several instances can be added to a cluster behind a load balancer.