

Transcoding Architecture with FFmpeg

Jess Portnoy

Kaltura, Inc

jess.portnoy@kaltura.com

January 27, 2016

The session will cover the following sub-topics:

- Video file detection and transcoding
 - * Transcoding Profiles and Flavor Params
 - * Mediainfo
 - * FFmpeg
 - * A walk through the KDL [Kaltura Decision Layer] code
- Playback
 - * Playback methods [Progressive HTTP, HDS, HLS, DASH]
 - * What is a playback manifest?
 - * An end to end walk through of the playback process

The following projects and their role in our architecture will be surveyed (All open source)

- Mediainfo - used to analyse source videos to determine how best to handle them
- FFmpeg and Mencoder - used to optimise source videos so that they'll be best viewed by various devices; i.e: Desktops, tablets and cellphones running different OSes
- ImageMagic - used for image manipulation and thumbnail creation
- nginx-vod-module - An Nginx module for On-the-fly repackaging of MP4 files to DASH, HDS, HLS, MSS
- The Kaltura HTML5 player

Transcoding Profiles and Flavor Params

In today's world, we have many devices capable of playing videos to name just a few super groups:

- PCs
- Cell Phones
- Tablets

Of course, each such super group has many devices and their hardware and software specs vary quite a bit.

Transcoding profiles enable us to instruct FFmpeg how to create multiple files [flavors] from the original source video.

When it's playback time, the best flavor will be chosen, based on the device and network conditions.

KDL - What is it?

KDL is the Kaltura Decision Layer, it is part of Kaltura's core code.

Using FOSS tools such as Mediainfo and FFmpeg, KDL's job is to determine how to best optimise a given source video into different flavors, to be consumed on various devices, in varying network conditions.

Like most of Kaltura's core functionality, KDL is written in PHP.

Video Ingestion - General Process Outline

- User uploads a video file
- Video file specs are analysed using mediainfo
- Based on the flavors defined in the transcoding profile, KDL will trigger FFmpeg commands to transcode the source video into flavor assets

Delivery Profiles

As mentioned before, there are many different devices out there but, just as important, the networks to which these devices are connected to, will offer varying bandwidths.

This is where Adaptive bitrate streaming comes into play. It works by detecting a user's bandwidth and CPU capacity in real time and adjusting the quality of a video stream accordingly.

Delivery Profiles - cont'd

The most commonly used implementations of Adaptive bitrate streaming nowadays are:

- MPEG-DASH
- HDS
- HLS

Kaltura handles this by defining a set of delivery profiles. Each profile has its own parameters such as the base URL from which the serve the video, the streamer type and optionally, the token mechanism to use, if you wish to limit access to your content.

Playback - General Process Outline

- User clicks play
- Player calls the Kaltura playManifest API
- A manifest is returned to the player according to the device and delivery method
- If an adaptive bitrate method is used, the flavor served may seamlessly vary depending on network conditions

On-the-fly repackaging of MP4 files to DASH, HDS, HLS, MSS

Kaltura's Nginx VOD Module enables you to repackage an MP4 file so that it can be served using DASH, HDS, HLS or MSS.

The module supports the following work modes:

- Local - serve locally accessible files (local disk/NFS mounted)
- Remote - serve files accessible via HTTP using range requests
- Mapped - perform an HTTP request to map the input URI to a locally accessible file

It can therefore, be used in stand alone mode to do the repackaging or, integrate with Kaltura's server.

Just like the Kaltura server, this is a FOSS project and its code is available at <https://github.com/kaltura/nginx-vod-module>

References



FFmpeg - <https://www.ffmpeg.org>



Mediainfo - <https://mediaarea.net>



Kaltura Server repo - <https://github.com/kaltura/server>



Nginx VOD module - <https://github.com/kaltura/nginx-vod-module>



Kaltura HTML5 player - <https://github.com/kaltura/mwembed>



HDS -
https://en.wikipedia.org/wiki/Adaptive_bitrate_streaming#Adobe_HTTP_Dynamic



HLS -
https://en.wikipedia.org/wiki/Adaptive_bitrate_streaming#Apple_HTTP_Adaptive



DASH -
https://en.wikipedia.org/wiki/Adaptive_bitrate_streaming#MPEG-DASH



Kaltura community site - <http://kaltura.org>



Community forums - <http://kaltura.discoursehosting.net>

The End && questions