A Brief History of Netflix Streaming

May 2013

David Ronca, Manager Encoding Tools
Introduction

- Netflix started streaming in 2007 with IE-plugin hosting Windows Media Player
- Today, > 80 million active Netflix devices phones, tablets, game consoles, and TV’s
Overview

• Major milestones of our streaming evolution
• Key codec and packaging decisions
• The evolution of our adaptive streaming model
• How we did iPad in < 60 days
• Move to standards-based streaming
Terminology

- Muxed (multiplexed) Stream: A/V interleaved in one stream

- Unmuxed (non-multiplexed) Stream: Separate A/V Streams
A Humble Beginning
The Streaming Client 1.0

- IE ActiveX Plugin
- Hosted Windows Media Player
- Muxed A/V: VC-1 Video, WMA Audio
- WMDRM
- ASF Container
Microsoft and Netflix Unveil Partnership to Instantly Stream Movies and TV Episodes to the TV via Xbox LIVE
July 14, 2008
Exclusive agreement creates game-changing home entertainment experience; more than 10,000 movies and TV episodes will be available to watch instantly at no additional cost to paying subscribers.

The Netflix Player by Roku

Instinctly. Netflix Now Gives You More: Movies and TV Episodes In Your Living Room

$99.99 One-time purchase

Small as a paperback book. Fills any room with instant entertainment. 30-day money back guarantee

Over 10,000 movies and TV episodes instantly
No change in monthly Netflix costs and you continue to get your DVDs by mail
Easily connects directly to your TV
Pause, rewind or play anytime - just like a DVO
Guaranteed to work with your TV

Choose Movies on Your TV
Movies and TV episodes in your Netflix Instant Queue appear right on your TV screen. Use the included remote control to browse through the items you've added to your Instant Queue and click something to watch —

What's being said...
"The Netflix Player by Roku revolusorizes the way entertainment is served. It's the best new benefit a Netflix member could have."
Reed Hastings
CEO, Netflix
The Silverlight Player

- PC/Mac
- A/B Testing
- Adaptive Streaming
- Unmuxed A/V Streams
- PlayReady DRM
- ASF Container
- Chunk Index
Adaptive Streaming Silverlight Style

- Fixed size closed-GOP Chunks, same for all bit rates
- Per Stream Index in ASF Header Object
- One audio stream selected based on initial BW
- Client varies video bitrate to manage buffer

```
500 kbps
Index 1 2 3 4 5 6 7 8 ... n

1000 kbps
Index 1 2 3 4 5 6 7 8 ... n

1500 kbps
Index 1 2 3 4 5 6 7 8 ... n

Index  Audio
```
The Silverlight Player Hack

- Silverlight 2.0 did not allow HTTP 1.1 Range headers (needed for range requests)
- Chunked download requires range request
- The Solution: “Range Request in URL”
  - Client inserts range request byte range into URL
  - CDN translates to Range Request

http://nflx.i.4535ed7e.x.lcdn.nflximg.com/98/1949855098.ismv/range/0-6359?etime=20130327151800&movieHash=810&encoded=05c27a4abb6c2ec423f02&random=460429451 HTTP/1.1
Silverlight Innovations

• 2-get rule
  – Max 2 http gets to pull stream headers
  – Index at front of file

• A/B testable client

• Heuristics-driven adaptive streaming
  – Metric is re-buffer rate
  – A/B test to maximize quality and minimize re-buffers
CE2 SDK

- Flash UI
- Adaptive Streaming
- Unmuxed A/V Streams
- VC1/WMA/DD+
- WMDRM
- ASF Container
- Chunk Index
PS3 V1 (Vega)
Vega: A Blu-Ray Hack

- PS3 is a very powerful BD Player
- BD-Live application (BD-Java)
- Progressive Playlist playback
- Adaptive Streaming: client-side muxing
- AVC Video, AC3 Audio
- AACS DRM
- First LOLOMO UI
Vega Streaming Model

• Lazy TDM M2TS Transport Stream Stream Muxing
• From the Spec
  – PSI arrival interval 100ms max
  – Min 20 PSI sets for 2-second GOP (we used 21)
  – 1-second max audio buffer
• Allows for “coarse” audio interleaving
Vega Stream Format

- Single file with 2-second chunks
- Each chunk divided into 3 segments of .667s
- Each M2TS file has header with index
- Same model for A & V
- Client pulls all stream headers at startup
Vega Adaptive Client

- Client produces local muxed files
- Files are fed to BD engine with Progressive Playlist
- Padding to AACS encryption block boundry
Vega Hacks

• Need to mux encrypted streams
• Solution all chunk segments padded to 6K block
  – Segments 1 & 2 padded with packets from “borrowed” from next segment
  – Segment 3 padded with Null Packets

```
Seg 1  Pad  Seg 2  Pad  Seg 3  Pad

1 GOP or audio Chunk
```

• M2TS Continuity count aligned across all video streams
  – Solution: Each video chunk starts with cc == 0.
  – Add video filler packets to get CC to 15

```
Seg 1  Pad  Seg 2  Pad  Seg 3  CC  Pad

1 GOP
```
Vega Encoding Challenge

- Highly complex streams
- Unusual Netflix model, no OTS tools available
- Non-conformed streams would crash BD system
- Tight schedule for encoding, packaging, and deploying assets
The Vega Encoding Solution

• Custom tools for BD packaging
• Custom tools for BD Stream verification.
• First Netflix project to use EC2-based encoding
• Encoding Tools team is started
• Media Pipeline team team comes of age
Wii V1 (aka Link)
Link Streaming Model

- H263-Advanced Simple Video
- Ogg-Vorbis Audio
- Widevine DRM
- ASF Container
- Chunk Index
Issue: Profile Bloat

- Overlapping profiles necessitated by DRM
- Assume 40,000,000 customers == 120,000+ viewables (6X)
- 25-50 TB per profile (late 2009)
CE3 Profile (V1)

- H264 Baseline & Main
- AAC/DD+
- Fragmented ISO container
- Microsoft Piff
- Multi-DRM
- Internal packaging tools
iPad/iPhone (NATO)
The Challenge

• On February 6, 2010, Apple invited Netflix to be part of the iPad launch (April 3, 2010)
• HTTP Live Streaming (HLS) model incompatible with Netflix systems in most every way
• 1.5 weeks later, we had HLS adaptive working
• 4/3/2010 Netflix was live on iPad with ~20,000 titles (2 bitrates each)
Http Live Streaming Issues

• Expiring URLs complicate M3U playlist
• DRM (or lack thereof)
• M2TS vs. fragmented ISO
• Physical vs. virtual chunked files
• Muxed vs. unmuxed
• Precise audio alignment across chunks
HLS Chunked Files

- Precise alignment of audio across all bitrates
- Easy if all bitrates muxed at one time on same system
- Difficult in distributed cloud environment
HLS Muxed M2TS (Vega Revisited)

- Audio and video muxed using index
- Precise Audio alignment
- Move muxing to backend
- About 1 day to convert tools
M3U Playlists

- Playlist management difficult with expiring URLs
- Add Index to muxed files
- Move Playlist generation to client
Physical Chunking (billion files model)

- NCCP Service not designed for chunked files
- Troublesome for CDNs (at Netflix scale)
- Solution: range-request in URL (from Silverlight)
- Virtual chunking looks like physical chunking
Sample M3U PlayList

playlist "http://netflix-532.vo.llnwd.net/s/s11/904/1377925904.ts.prdy?p=60&e=1296814051&h=673020a1f218df112dde1a403c5163f4"
#EXTM3U
#EXT-X-TARGETDURATION:10.010000
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-KEY:METHOD=AES-128,URI="https://ihost.netflix.com:4343/keys/crypt0.key"
#EXTINF:10.010000
http://netflix-532.vo.llnwd.net/s/s11/904/1377925904.ts.prdy/range/31020-304763?p=60&e=1296814051&h=673020a1f218df112dde1a403c5163f4
#EXTINF:10.010000
http://netflix-532.vo.llnwd.net/s/s11/904/1377925904.ts.prdy/range/304764-496715?p=60&e=1296814051&h=673020a1f218df112dde1a403c5163f4
#EXTINF:10.010000
http://netflix-532.vo.llnwd.net/s/s11/904/1377925904.ts.prdy/range/496716-676827?p=60&e=1296814051&h=673020a1f218df112dde1a403c5163f4
#EXTINF:10.010000
iPad Summary

• Vega tools adapted to HLS M2TS model
• Client-side playlist generation
• Range-request-in-URL for virtual chunking
• Encoding started ~2 weeks before launch
• Live on 4/3 with ~22,000 titles
• 2 bitrates per title
• iPhone followed in fall 2010
PS3 V2 (Rigel)

- Native PS3 Application
- Vega end-of-life
- CE3 Profile
- PlayReady DRM
• Wii Store App
• H263/OggVorbis
• FMP4-Piff
• PlayReady DRM
• In 2010, Netflix joined the DASH committee
• One motivation, PIFF-DASH compatibility
• DASH streams also PIFF compatible
CE3 Profile V2 (DASH)

- Android Devices
- Roku 2
- xBox, PS/3, Wii, WiiU
Profile Summary

CE1
- Muxed ASF
- WMDRM
  - WMA
  - VC-1

Silverlight
- Unmuxed ASF
- PlayReady DRM
  - WMA
  - VC-1

CE2
- Unmuxed ASF
- WMDRM
  - WMA
  - VC-1

Vega (PS3)
- Unmuxed M2TS
  - AACS
  - AC-3
  - H264

Link (Wii)
- Unmuxed ASF
  - Widevine
  - Ogg
  - H263

CE3-DASH
- Unmuxed FMP4
- PlayReady/Widevine
  - OggVorbis
  - DD+
  - HE-AAC
  - H264

Kirby-PIFF
- Unmuxed FMP4
- PlayReady
  - OggVorbis
  - DD+
  - HE-AAC
  - H263

iOS1
- Muxed M2TS
- PlayReady/NFKE
  - DD
  - HE-AAC
  - H264

iOS2
- Unmuxed M2TS
- PlayReady/NFKE
  - DD
  - HE-AAC
  - H264

CE4*
- HEVC (UltraHD)

* Planned
Summary

- Netflix streaming started very simple
- Incremental evolution
- Silverlight was platform for early innovation
- Early development model: “go to device”
- Now standards-based
- Agile development
- Hack when necessary
- Innovation in other areas such as UI, heuristics, etc.
Questions?

David Ronca
Manager, Encoding Tools
dronca@netflix.com