HTTP Adaptive Streaming in practice

Mark Watson

(with thanks to the Netflix adaptive streaming team!)

ACM MMSys 2011 – 22-24 February 2011, San Jose, CA
Netflix Overview

- Started with DVD-by-mail, now primarily Internet streaming

- 20+ million\(^1\) subscribers, growing rapidly (>15% of US households subscribe to Netflix)

- USA-only for ten years, Canada in 2010, further expansion in 2011+

- Unlimited Streaming = $7.99/month
  - Plus 1 DVD at a time = $9.99/month

\(^1\) subscriber reported 1/26/11
Partner Products
Contents

• Why HTTP adaptive streaming?
• Streaming approaches
• Measuring quality and the value of quality
• Adaptation algorithms and open problems
Why HTTP Adaptive Streaming?

Commodity service

Competing providers

Economies of scale

Netflix Confidential
Client-centric approach

• Client has the best view of network conditions
• No session state in network
  – Redundancy
  – Scalability
• Faster innovation and experimentation
• BUT, relies on client for operational metrics
  – only the client knows what really happened anyway
Scalability examples

• Microsoft streaming of 2008 Olympics
  – 4 Petabytes live & VoD content in one month
  – North America (av. user bandwidth 2Mbit/s)
  – Millions of simultaneous sessions
  – Over existing infrastructure

• Netflix
  – 20% of North American Internet traffic at peak hours
  – Millions of hours of content every day
  – Bitrates up to 4.8Mbit/s
  – Almost no dedicated infrastructure
    • Control servers in AWS
    • Content delivery through CDNs
Streaming bitrate performance

(just one device type)
Streaming rebuffer rates

(just one device type)
Contents

• Why HTTP adaptive streaming?
• Streaming approaches
• Measuring quality and the value of quality
• Adaptivity algorithms and open problems
Adaptive streaming in practice

- HTTP Live Streaming
- Smooth Streaming
- Adobe HTTP Dynamic Streaming
- MPEG DASH

Small media chunks ("streamlets")
Chunks created at origin server
HTTP Byte Range requests
Adaptive streaming in practice

- Move Networks, Inc.
- Adobe HTTP Dynamic Streaming
- Microsoft Silverlight
- MPEG DASH
- Netflix

- HTTP Live Streaming
- Smooth Streaming
- Combined A/V streams only
- Separate Audio/Video
Adaptive streaming in practice

HTTP Live Streaming
- No switchpoint alignment

Adobe HTTP Dynamic Streaming

MPEG DASH
- Switchpoint alignment (optional)

Microsoft Silverlight
Smooth Streaming

Move Networks, Inc.
Switchpoint Alignment

Stream 1

Stream 2

Switch point

Stream 1

Stream 2

Switch point
Adaptive streaming summary

• For On Demand
  – Chunks are unnecessary and costly
  – Byte Range requests have caching and flexibility advantages
  – Separate audio/video essential for language support

• For Live
  – Chunks are unavoidable
  – Still value in decoupling request size from chunk size
  – Multiple language audio tracks are rare
  – May need manifest updates

• For both
  – Switch point alignment required for most CE decoding pipelines
MPEG DASH

- Supports both unchunked & chunked
- Supports both separate & combined A/V
- Index formats for efficient byte range operation
- ISO Base Media File Format w/common encryption
- Rigorous definition of stream alignment requirements
- Signaling of different alignment modes
- Many useful stream and track annotations

Currently the best candidate for an open standard for adaptive streaming
Contents

• Why HTTP adaptive streaming?
• Streaming approaches
• Measuring quality and the value of quality
• Adaptivity algorithms and open problems
Measuring quality

• Reliable transport => all-or-nothing delivery

• Quality characterized by
  – Video quality
    • At startup, average and variability
  – Re-buffer rate
    • Re-buffers per viewing hour, duration of re-buffer pauses
  – Startup delay
    • Time from use action to first frame displayed
Importance of client metrics

• Metrics are operationally essential
  – Detecting and debugging failures
  – Managing performance
  – Experimentation

• Absence of server-side metrics places onus on client

• What do we need?
  – Reports of what the user did (or didn’t) see
    • Which part of which stream presented when
  – Reports of what happened on the network
    • Requests sent, responses received, timing, throughput
Contents

• Why HTTP adaptive streaming?
• Streaming approaches
• Measuring quality and the value of quality
• Adaptivity algorithms and open problems
Adaptation problem

Choose sequence and timing of requests to

Minimize probability of re-buffers

Maximize quality
Adaptation problem: Inputs

History

Current state

Possible choices

Capturing and representing all this information is not easy!
Adaptation problem: logic

History

Current state

Possible choices

Model of future bandwidth

Expected performance for each choice
Adaptation problem: example

- Model of future bandwidth
  - Constant
  - Equal to average over last 10s

- Analysis of choices
  - Construct “plan” for each choice
  - Determine re-buffers for each plan
Adaptation problem: future work

• Good models of future bandwidth based on history
  – Short term history
  – Long term history (across multiple sessions)
• Tractable representations of future choices
  – Including scalability, multiple streams
• Convolution of future bandwidth models with possible plans
Conclusions

• Asynchronous delivery of same content to many users is a first-class network service
  – HTTP CDNs may not be the “perfect” architecture, but it’s working pretty well at scale

• Many variations on HTTP Adaptive Streaming theme in deployed systems and emerging standards
  – MPEG DASH provides sufficient flexibility here

• Adaptation is not straightforward
  – How to model bandwidth future based on history?
  – How to efficiently search choice space to maximise quality goals?
  – What are the quality goals?
Questions?

Mark Watson, watsm@netflix.com