Delivering Video at Scale

Simon Frost
2.5 million streams a day
What’s the problem?
Distributing video online is hard

http://www.flickr.com/photos/55853619@N00/145052885/
The workflow for delivering video can be quite long and intensive. Broadcast and physical media, very specialised, with one distribution mechanism. Internet distribution is much harder and we typically have to think about scalability earlier.

Workflows can be complicated.
Video makes up a lot of internet traffic but not activity
Wednesday, 23 June 2010

What if I told you there was a secret to get all of this right
and then you had to wait
and then you had to wait

(annoying, isn’t it)
4 out of 5 viewers leave if a stream buffers once


You can have the best content in the world but no-one is watching

Users watch for longer if quality is good
How your users see online video

http://www.flickr.com/photos/86533050@N00/315599041/
Why is there a problem?
The internet was not designed for video

It was designed for other priorities
We already have a highly available network – the TV broadcast networks we use every day
But we can create new services, it’s not broadcast
http://www.flickr.com/photos/sflaw/222795669/
The biggest problem is bandwidth
TV networks have guaranteed bandwidth and adding new TVs doesn’t mean we need more bandwidth.
Net is peer-to-peer.
Adding a device requires bandwidth and it’s a contended resource
http://www.flickr.com/photos/nicocavallotto/424047642/
The internet is a global network and it’s a network of networks. There’s no optimisation in route. When I download data, it may come from the internet, go one of country and back.
Radio magnetic spectrum is policed
The internet is a free-for-all, that email you won’t need for a few hours is in contention with the video you need right now
ISPs shape traffic
http://www.flickr.com/photos/galred/323356684
Any TV is capable of viewing any digital video stream. They are designed to do one thing. Devices connected to the internet come cover a wide range of performance characteristics. Only a few devices can play HD quality video.

http://www.flickr.com/photos/49503114775@N01/255241547/
Can be expensive; no fixed cost for distribution
You have to pay for storage and bandwidth, you may have to pay for hardware
1http://www.flickr.com/photos/77156371@N00/20243332/
How do we solve the problem?

How do we solve these problems
By application of certain techniques, and making the right trade-offs
We’re in the business of good enough
two for starters...
Before we go any further – let’s be realistic
use a CDN – use many CDNs

● Being clever with bandwidth

● Reducing latency
Thank you...

We could end there
It’s not that simple, a CDN will only get you part of the way there.
A CDN will provide cost-effective bandwidth and storage
We can start solving by looking at what is causing these bandwidth problems
http://www.flickr.com/photos/fornal/398136952
What is digital video?

So what is digital video? Like any standard digital media, uses algorithms to create a compressed stream of data, which is then wrapped in a digital file format.
File Formats

- Wrap data streams e.g. video, audio
- Contains structural and descriptive metadata
- Examples: MPEG4, Windows Media, avi, webM, FLV
Codecs

- Control compressing and unconpressing of media
- Examples: H.264, Windows Media, VP6/8
Transcoding

- Conversion of one format to another
- CPU and time-intensive
- But scalability can start here
There's a myriad of options. too many to go into in this talk
There's an artistic interpretation here - does it look good
http://www.flickr.com/photos/lrosa/263066362/sizes/o/
Bitrate

- Higher bitrates offer higher quality...
- ...although not all codecs are equal
- Optimise for frame size
- Constant Bitrate (CBR) vs Variable Bitrate (VBR)
It goes without saying then, that the lower bitrate, the less bandwidth you’ll need
And you probably need to support many
Blu Ray up to 40 Mbit/s
HDTV – 9.7 to 16 Mbit/s
DVD quality – 5 Mbit/s
Standard def TV ~ 3.5 Mbit/s
Online Video ~ 0.5 – 3 Mbit/s
Video is a succession of frames
framerates – number of frames/sec
Analogue video was uncompressed pictures – interlacing
Digital video – send a whole frame, then send differences in next sequence of frames
I – intra-coded – full frame
P – predictive coded – frames that look back
B – bidirectionally predictive coded frames look forward to next frame – typically take a lot of processing though
Keyframes are what really need to learn though
Less keyframes – smaller file size but noise on network causes problems
More keyframes – easier to search
Remember this, as it’s a key requirement of ABR
Download vs streaming

The key methods of delivery:
download is what we’re used to
streaming - data is pulled as the client needs it
streaming is instant – download needs entire file so you can get progressive download
download is good for small files, low quality video
larger files, higher quality demand better management of bandwidth and resources
streaming provides this
p2p – designed to lower cost of your bandwidth – but requires well seeded content. many
times falls back to a standard download model

- Use progressive
- Good for short-form low quality
- Can’t adapt to congestion
- P2P
- Use VBR
Streaming

- Good for long-form high quality video
- More instant
- Allows better control of resources
- Another technology to learn and manage e.g. Adobe’s Real Time Messaging Protocol (RTMP)
- Use CBR

Wednesday, 23 June 2010
Adaptive Bitrate

- Good for user experience
- Transcode multiple streams
- Use a low keyframe interval
- Requires tuning
- Good support across the industry

Wednesday, 23 June 2010
Can be expensive – you’ll need many streams
Large files – as low keyframe interval
But provides best spreading of cost
All major platforms offer this – need to transcode for it
# Video platforms

<table>
<thead>
<tr>
<th></th>
<th>Installed Base</th>
<th>Codecs</th>
<th>ABR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flash</td>
<td>Most PCs and Macs</td>
<td>Good support</td>
<td>Yes</td>
</tr>
<tr>
<td>Silverlight</td>
<td>Some PCs and Macs</td>
<td>Good support</td>
<td>Yes</td>
</tr>
<tr>
<td>Apple</td>
<td>Macs, iPhones, iPads, iPods</td>
<td>H.264</td>
<td>Yes</td>
</tr>
<tr>
<td>HTML5</td>
<td>Modern browsers</td>
<td>No standard codec</td>
<td>No</td>
</tr>
</tbody>
</table>
White-label Platforms

- Offer ABR
- Good support for codecs and formats
- Multi-CDN support
- e.g. Oolya, Brightcove
Latency helped by CDNs
But there’s very little we can do here – unless you’re an ISP e.g. Last mile peering
http://www.flickr.com/photos/66578088@N00/709470877/
HTTP Streaming

- It’s HTTP!
- Can be cached
- Good support from platforms and CDNs
Kill the unicorn of net neutrality is the only way
http://www.flickr.com/photos/michaelgoodin/3736488245
Use a platform which offers hardware decoding
Some ABR offers CPU QOS, but it’s not perfect
http://www.flickr.com/photos/rogersmith/280267119
It’s hard to find the right balance
Only target the devices you think you can
There is an overlap between many of these though
...and finally
This is the future; much of what I said, still needs to be implemented
http://www.flickr.com/photos/telstar/2067218329/sizes/o/
- No plugins
- HTTP
- Easy to deploy
HTML5 - what’s missing

- Adaptive bitrate
- Standard codec
- DRM
How to scale

- Transcode it right
- Optimise video for the client
- Use one or more CDNs
- Use the right method - download or streaming
- If you stream - use ABR and HTTP
Thank you

- Simon Frost
- simon@automatedmedia.co.uk
- @spli

- [http://uk.linkedin.com/in/simonafrost](http://uk.linkedin.com/in/simonafrost)

- Thanks for images: Roger Smith, Michael Goodin, Ack Ook, Luigi Rosa, Bob.Fornal, jonnystyles, eurleif, gailf548, nico.cavallotto, sfllaw, Telstar Logistics, Chuckumentary, Marvin (PA)