Producing for Multiple Screen Delivery

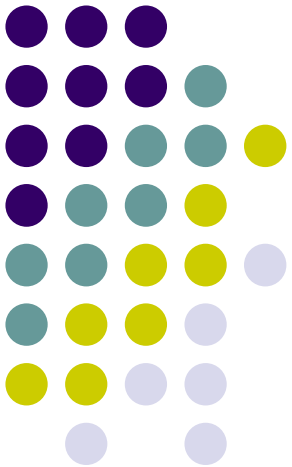
Jan Ozer

www.streaminglearningcenter.com

jozer@mindspring.com/

276-238-9135

@janozer

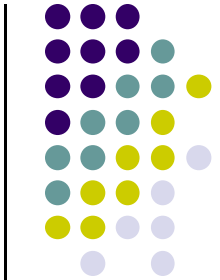


For Additional Information

- This presentation was created by the author of *Producing Streaming Video for Multiple Screen Delivery*
- For more information, see:

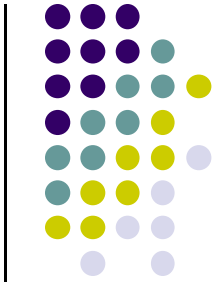
  Bit.ly/Ozer_multi

Agenda



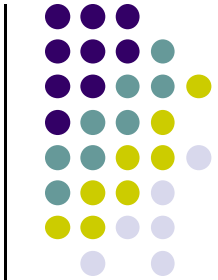
- Overview/Development alternatives
- About multiple screen delivery
- H.264 overview
- Introduction to adaptive bitrate streaming (ABR)
- ABR alternatives
- Configuring your ABR streams

Premises



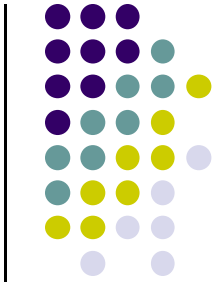
- To maximize revenue, video producers must distribute to all available screens:
  - Computers (Windows, Mac)
  - Mobile (iOS, Android, other)
  - Over the Top (OTT) (Roku, Apple TV, Boxee)

Caveats



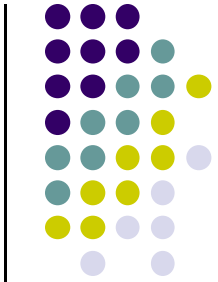
- Very complex, dynamic subject
- I'm an encoding guy, not a programmer
  - Particularly important in later stages
- It's not technology, it's religion
  - When there's a will, there's a way

Development Approaches



Desktop	Browser	Plugin	OTS Player	Custom Player
Reach	Depends (HTML5 about 70-90%)	Plug-in dependent (Flash is good)	Depends upon dependencies	Ditto
Cost	Nil	Nil	$300 +	?
Complexity	None	None	Minimal	Programmer required
Features (caption/ DRM)	Basic	Basic	More advanced	Custom

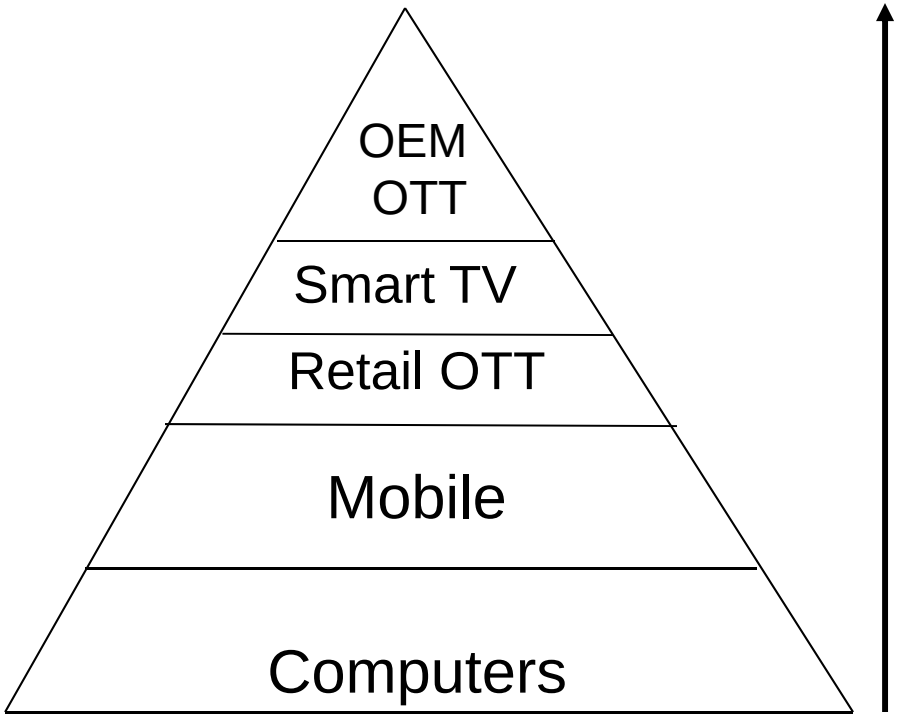
Development Approaches



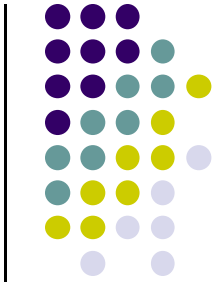
Mobile	Native	App
Reach	100%	100%
Cost	Nil	?
Complexity	None	Programmer required
Features (caption/ DRM)	Basic	Custom

About Multiple Screen Delivery

- As you climb the pyramid
  - Number of viewers *decreases*
  - Cost and complexity *increases*
  - Cost per viewer *skyrockets*
- Where do you want to go today?



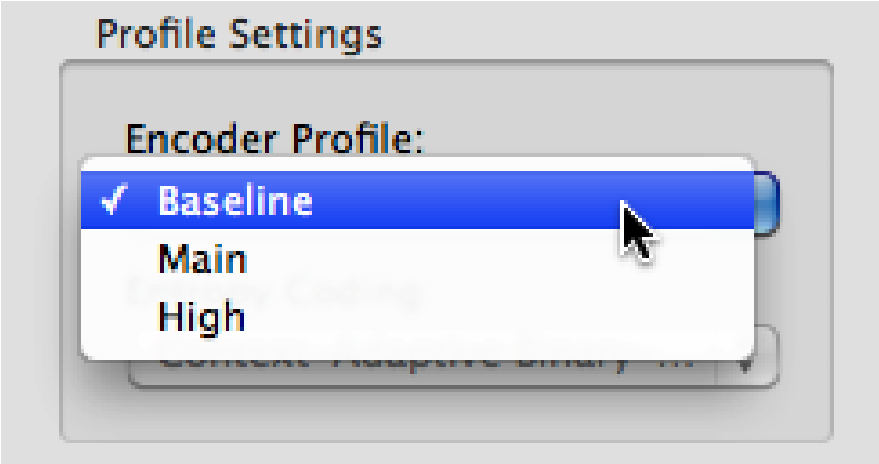
Rule Number 1: Use H.264



- High quality compression technology
- Plays almost everywhere
  - Computers
    - Flash, QuickTime, Silverlight
    - Via HTML5 (Chrome, IE9, Safari, NOT Firefox, Opera)
  - Mobile–plays natively on all mobile devices
  - OTT – plays natively on all OTT devices

H.264 Encoding - Basics

- Profiles/Levels
  - Most critical compatibility-related setting
    - Encode using wrong profile, file won't play
  - Available on all encoding tools



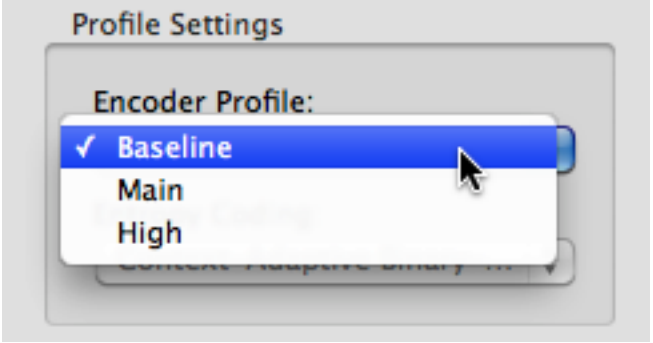
What are H.264 Profiles?

- "Define a set of coding tools or algorithms that can be used in generating a bitstream"

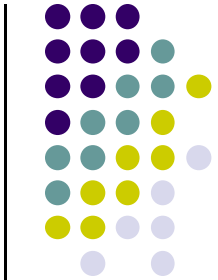
	Baseline	Extended	Main	High
I and P Slices	Yes	Yes	Yes	Yes
B Slices	No	Yes	Yes	Yes
Multiple Reference Frames	Yes	Yes	Yes	Yes
In-Loop Deblocking Filter	Yes	Yes	Yes	Yes
CAVLC Entropy Coding	Yes	Yes	Yes	Yes
CABAC Entropy Coding	No	No	Yes	Yes
Interlaced Coding (PicAFF, MBAFF)	No	Yes	Yes	Yes
8x8 vs. 4x4 Transform Adaptivity	No	No	No	Yes
Quantization Scaling Matrices	No	No	No	Yes
Separate Cb and Cr QP control	No	No	No	Yes
Separate Color Plane Coding	No	No	No	No
Predictive Lossless Coding	No	No	No	No
	Baseline	Extended	Main	High

Why Do Profiles Exist?

- Meeting point for hardware vendor and video producer
- **Example:** Original video capable iPod is baseline only: why?
  - Lower power (cheaper) CPU
  - Less power consumption
- Video producer: to produce that will play on iPod, use Baseline profile

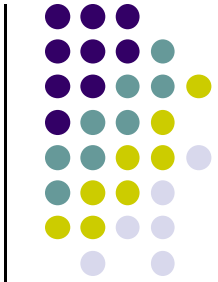


Rule Number 2



- Don’t exceed profile of target device
  - Exclusively a concern with mobile
- Computers and OTT devices can play High profile (any level)

Required Profiles for iDevices



	Original iPod (to-5g)	iPod nano/ classic/ iPod touch/iPhone to V4	iPhone 4 /iPod touch 4/ iPad 1&2	iPhone 4S	IPhone 5/ New iPad
Video codec	H.264	H.264	H.264	H.264	H.264
Profile/level	Baseline to Level 1.3	Baseline to Level 3.0	Main to Level 3.1	High to Level 4.1	High to Level 4.1
Max video data rate	768 kbps	2.5 Mbps	14 Mbps	50 Mbps	62.5 mbps for level
Max video resolution	320x240	640x480	720p	1080p	108p
Frame rate	30 fps	30 fps	30 fps	30 fps	30 fps
Audio codec	AAC-LC	AAC-LC	AAC-LC	AAC-LC	AAC-LC
Max audio data rate	160 kbps	160 kbps	160 kbps	160 kbps	160 kbps
Audio params	48 kHz, stereo	48 kHz, stereo	48 kHz, stereo	48 kHz, stereo	48 kHz, stereo

- Four playback classes
- Must configure video appropriately to play on target

Required Profiles for Android

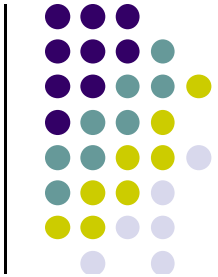
http://developer.android.com/guide/appendix/media-formats.html

	SD (Low quality)	SD (High quality)	HD (Not available on all devices)
Video codec	H.264 Baseline Profile	H.264 Baseline Profile	H.264 Baseline Profile
Video resolution	176 x 144 px	480 x 360 px	1280 x 720 px
Video frame rate	12 fps	30 fps	30 fps
Video bitrate	56 Kbps	500 Kbps	2 Mbps
Audio codec	AAC-LC	AAC-LC	AAC-LC
Audio channels	1 (mono)	2 (stereo)	2 (stereo)
Audio bitrate	24 Kbps	128 Kbps	192 Kbps

- These recommendations are least common denominator software-only playback
  - Most devices probably more capable (no table because too many manufacturers)
  - For 100% compatibility, use these

Required Profiles for Windows Phone

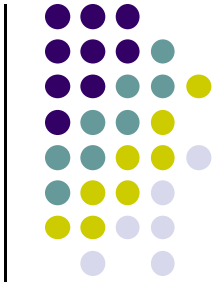
http://http://bit.ly/winphonemedia



- “Make sure your files do not exceed the parameters listed in the table for H.264 video on a 7x27a processor.”
- Smooth Streaming Media Element (SSME) allows for dynamic resolution changes. This option is only supported on 8x55 based devices. The 8x50 and 7x27a processors do not support this feature.

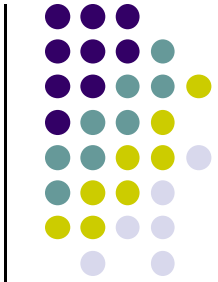
Feature	H.263	VC1	VC1	MPEG-4 Pt 2	H.264	H.264	H.264
Containers	3GP, 3G2	WMV	WMV	3GP, 3G2, MP4, M4V	3GP, 3G2, MP4, M4V, MOV	3GP, 3G2, MP4, M4V, MOV	3GP, 3G2, MP4, M4V, MOV
Profile	0	Simple	Main	Simple	Baseline	Main	High
Level	30	Main	Low	3	2.0	1.3 - CABAC, 2.0 - CAVLC	1.3 - CABAC, 2.0 - CAVLC
Max average video bit rate	2 Mbps	2 Mbps	1 Mbps	2 Mbps	2 Mbps	CABAC: 2 Mbps, CAVLC: 768 Kbps	CABAC: 2 Mbps, CAVLC: 768 Kbps
Max peak video bit rate	4 Mbps	Not Available	Not Available	4 Mbps	4 Mbps	4 Mbps	4 Mbps
Max resolution and frame rate	800×480 @ 30 fps	800×480 @ 30 fps	400×240 @ 30 fps	800×480 @ 30 fps	800×480 @ 30 fps	800×480 @ 30 fps	800×480 @ 30 fps
Smooth streaming support	No	No	No	No	Yes	Yes	Yes

Big Decision #1



- One set of files – all Baseline – that plays everywhere?
  - Cheapest, easiest
- Separate files:
  - Baseline – Very old iOS and Android
  - Main – Old iOS and Android
  - High – new iOS, computers and OTT
  - Optimal quality, but more encoding, storage and administrative

Apple Recommendations – TN2224



16:9 Aspect Ratio

	Dimensions	Frame Rate *	Total Bit Rate	Audio Bit Rate	Keyframe**	Restrict Profile to:	
CELL	480x320	na	64	64	na	na	
CELL	416x234	10 to 12	264	64	30 to 36	Baseline, 3.0	High
CELL	480x270	12 to 15	464	64	36 to 45	Baseline, 3.0	High
WIFI	640x360	29.97	664	64	90	Baseline, 3.0	High
WIFI	640x360	29.97	1264	64	90	Baseline, 3.1	
WIFI	960x540	29.97	1864	64	90	Main, 3.1	
WIFI	960x540	29.97	2564	64	90	Main, 3.1	
WIFI	1280x720	29.97	4564	64	90	Main, 3.1	
WIFI	1280x720	29.97	6564	64	90	Main, 3.1	
WIFI	1920x1080	29.97	8564	64	90	High, 4.0	

- Optimal strategy for delivering to mobile
- Should you produce a separate set using the High profile for computers and OTT

416x234p Comps

- Nothing dramatic here

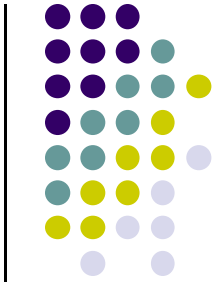
480x270p Comps

- Here either

640x360 @600 kbps Comps

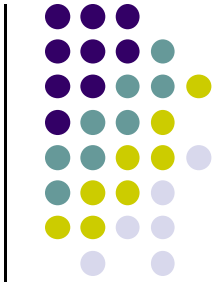
- Here either

Rule Number 3: Don’t assume you need custom groups



- Key takeaway:
  - The quality difference between Baseline and High may be less than you would expect
- Two sets of files for desktop/mobile/OTT?
  - Test quality difference at most extreme rates and see if the quality difference warrants it

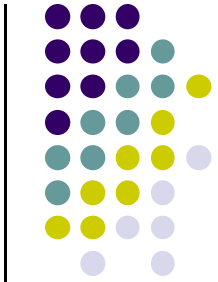
What are H.264 Levels?



- "Constrains key parameters in the bitstream"

Level number	Max video bit rate (VCL) for Baseline, Extended and Main Profiles	Max video bit rate (VCL) for High Profile	Examples for high resolution @ frame rate (max stored frames) in Level
1	64 kbit/s	80 kbit/s	128x96@30.9 (8) 176x144@15.0 (4)
1b	128 kbit/s	160 kbit/s	128x96@30.9 (8) 176x144@15.0 (4)
1.1	192 kbit/s	240 kbit/s	176x144@30.3 (9) 320x240@10.0 (3) 352x288@7.5 (2)
1.2	384 kbit/s	480 kbit/s	320x240@20.0 (7) 352x288@15.2 (6)

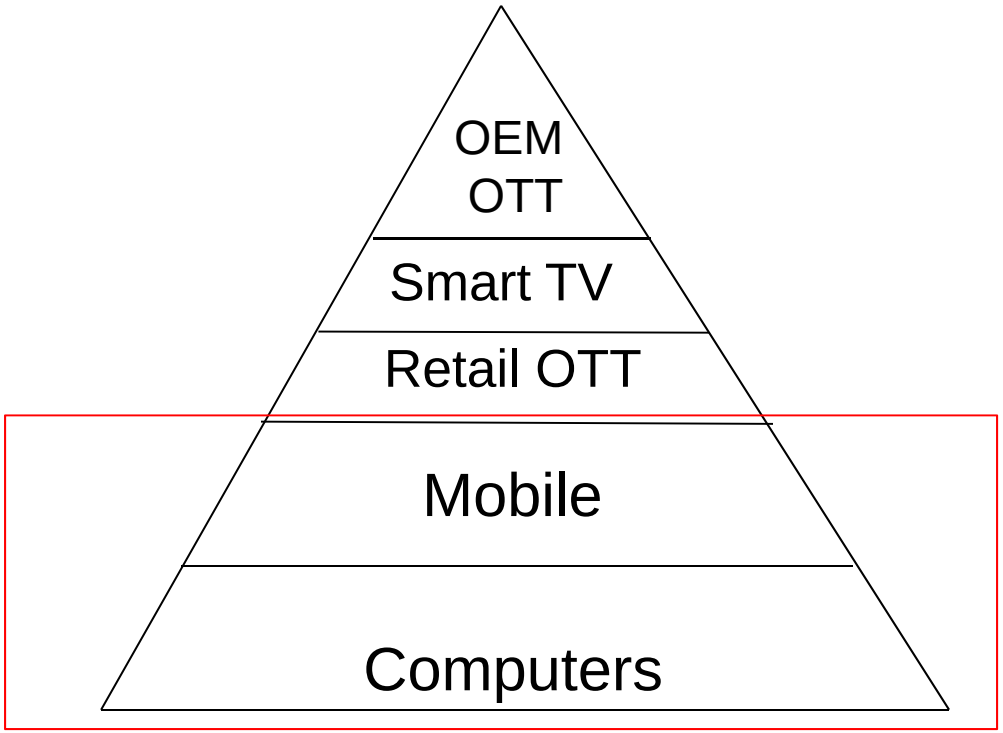
H.264 Levels-Devices



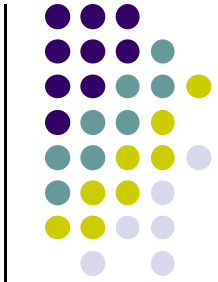
- Critical not to exceed level of target device
  - All devices have level limitations
  - Computers/OTT do not, so not a concern
  - Bottom line:
    - Mobile targeted streams must not exceed supported profile/level

Single File Streaming to Desktop and Mobile

- Use HTML5 with Flash Fallback
  - H.264 or H.264 + WebM
- Reach
  - All computers/all mobile
- Cost
  - Free (FFMP/HandBrake, code your own HTML5)
  - $549 – Squeeze – encode/create HTML for you
- Complexity: Minimal

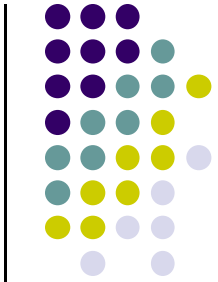


About Flash Fallback



- Why Flash Fallback
  - To reach 10%-15% of viewers who aren't running HTML5 compatible browsers
- What is Flash Fallback
  - Video tag presented first; if browser understands (and supports codec), video plays
  - If browser doesn't understand, Flash playback coding is provided
    - Plays H.264 file

What's It Look Like



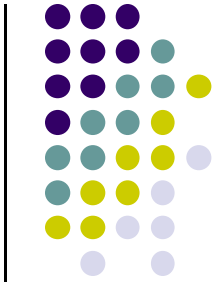
H.264

WebM

Flash

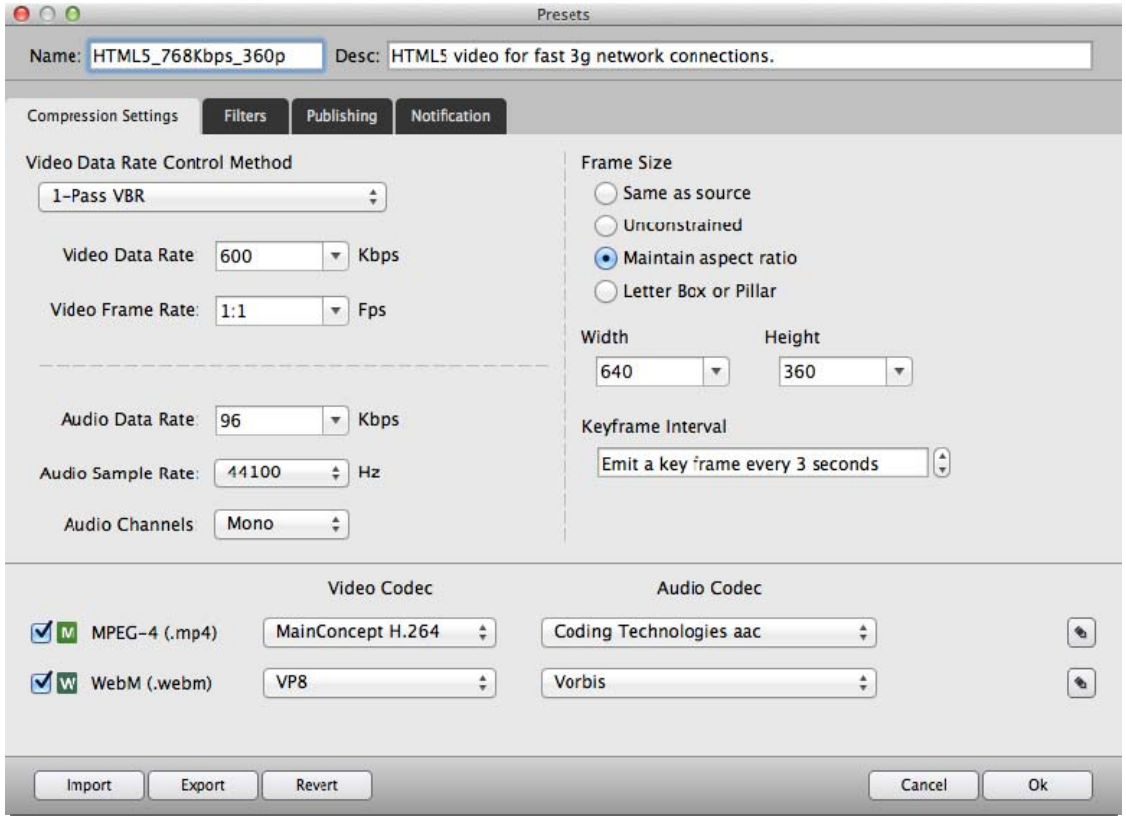
```
<video preload="auto" width="640"
       height="360" poster="Disney_html.vs-1.jpg" controls>
  <!-- mp4 is for I E 9, I E 10, S a f a r i -->
  <source src="Disney.html.mp4" type="video/mp4"/>
  <!-- webm for F i r e f o x, C h r o m e -->
  <source src="Disney.html.webm" type="video/webm"/>
  <!-- swf and mp4 is for older browsers that support F l a s h 9 -->
  <div>
    <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
            width="658" height="411"
            id="id-Disney">
      <param name="movie" value="Disney.html.swf"/>
      <param name="play" value="false" />
      <param name="flashvars" value="flvUrl=Disney.html.mp4" />
      <!--[if !IE]>-->
      <object type="application/x-shockwave-flash"
              width="658" height="411"
              data="Disney.html.swf">
        <param name="play" value="false" />
        <param name="flashvars" value="flvUrl=Disney.html.mp4" />
        <!--<![endif]-->
        <div>
          <a href="Disney.html.mp4">
            <img src="Disney_html.vs-1.jpg" alt="Disney" title="Disney" />
          </a>
        </div>
        <div>
          <a href="http://www.adobe.com/go/getflashplayer">
            <img src="http://www.adobe.com/images/shared/download_buttons/get_flash_player.gif"
                 alt="Get Adobe Flash player" />
```

Implementing HTML5 with Flash Fallback



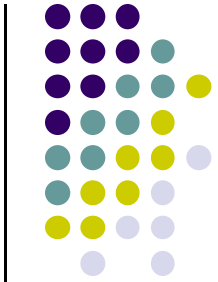
- Manual coding
  - Absolutely can encode files separately and create HTML code manually
- Sorenson Squeeze 9
  - Can produce both files and HTML code automatically
    - Squeeze makes it very simple
    - See tutorial up on Streaming Media website
      - bit.ly/squeeze_html5

Best Implementation – Sorenson Squeeze 9



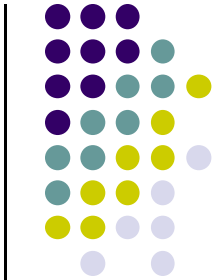
- Presets create both WebM and H.264 files
- All necessary HTML code
- Just encode/upload
- Test – bit.ly/what_dis

What You Don’t Get



- Not true streaming (progressive download)
  - Problem if distributing massive files
- Adaptive
- Captions
- DRM
- Reach

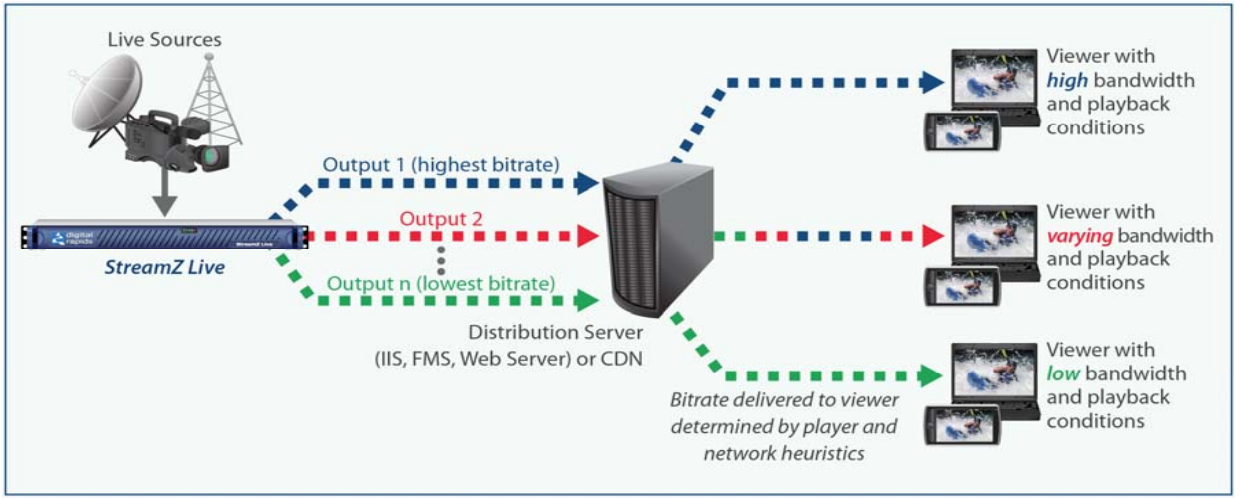
Introduction to Adaptive Streaming



- What it is
- Alternatives
- Implementing
  - HLS
  - DASH

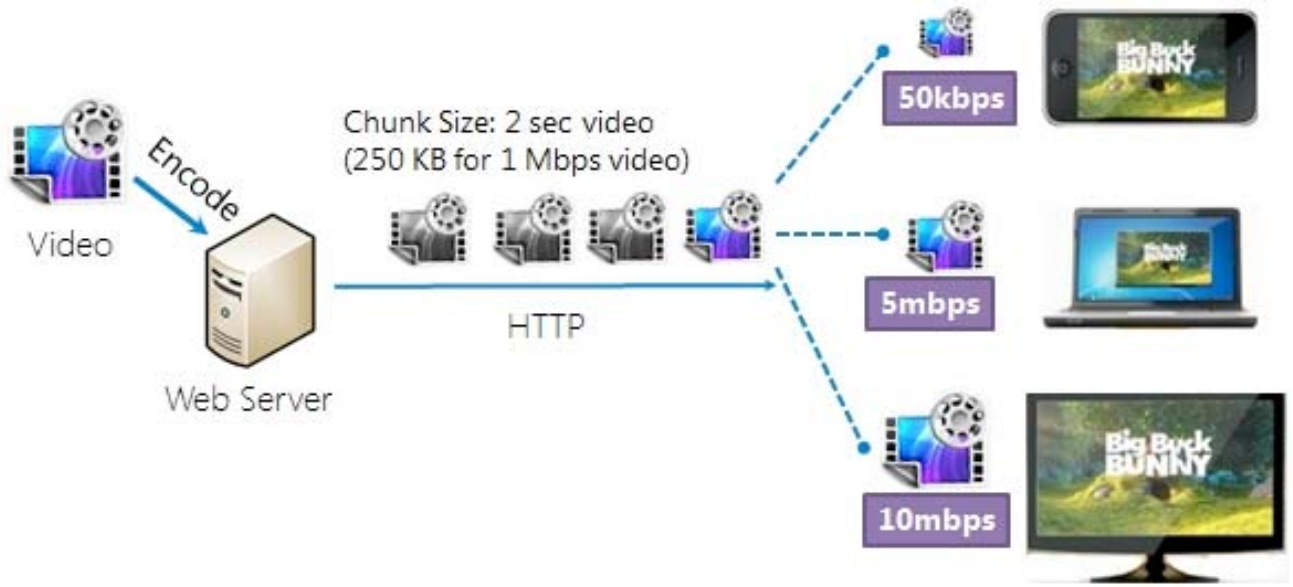
What is Adaptive Streaming?

- One stream in, multiple streams out
- Streams switched transparently to adapt to factors like:
  - Changing delivery bandwidths (avoid hard stops)
  - CPU utilization at client (avoid frame drops)



Key Benefits

- Match stream quality to connection and playback platform used by viewer

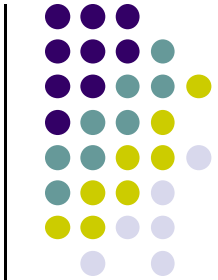


Mobile

Desktop

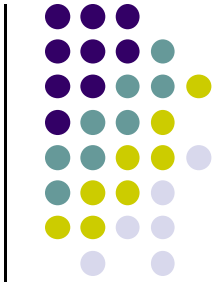
OTT

Technology Overview



- Two types of systems
  - Server-based (Flash, RTMP)
  - HTTP
    - HTTP Live Streaming (HLS)
    - Smooth Streaming
    - HTTP-based Dynamic Streaming (HDS)
    - Dynamic Adaptive Streaming over HTTP (DASH)

Deploying Adaptive Streaming Technologies

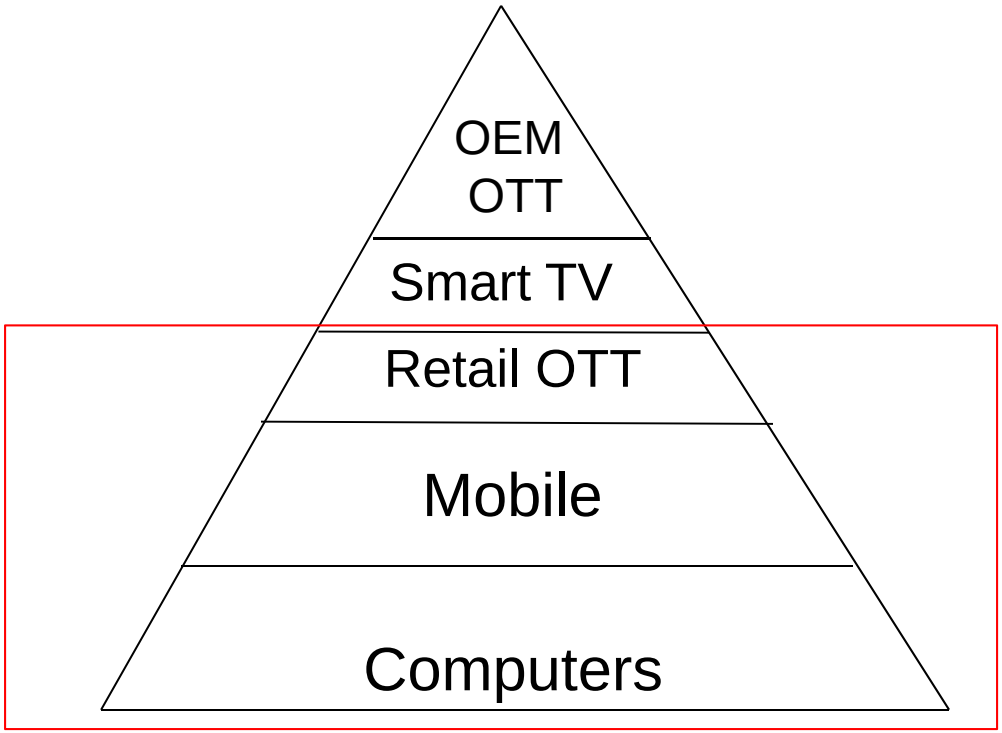


	Desktop	iOS	Android	Win Phone	Retail OTT
Flash – RTMP	With Flash	App	App	App	App?
HLS	With OTS Player	Native	Native/App/ Fallback	App	Lots Native
Smooth	Silverlight of Flash Plugin	App	App	App	Some native
Flash HDS	With Flash	App	App	App	App?
DASH	Custom development	App*	App**	App	App?

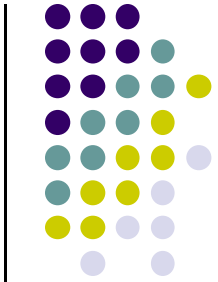
* Some question if Apple will approve app with DASH
** Android 4.4 (8.5%) supports DASH natively

Adaptive Streaming to Desktop and Mobile

- Use HLS with OTS Player
  - JW Player is one I've used
- Reach
  - All computers/all mobile/files compatible with retail OTT
- Cost
  - $49 - Compressor
  - $300 – JW Player
- Complexity: Minimal

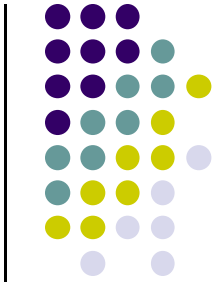


What You Don’t Get



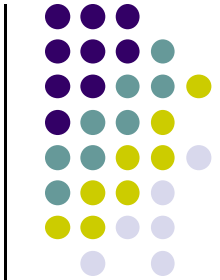
- Reach – tippy top tiers
- The ability to customize a group of streams for each target

Other Alternatives



- Create custom encoding groups
  - Approach taken by most very large streamers
  - Most expensive costwise

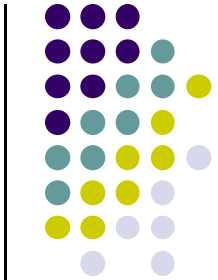
What We Know



- Most producers customize
  - Turner Broadcasting – NBA League Pass

Web	Type	Format	Resolution	Vid BR	Frame Rate	Aud Codec	Sample Rate	Aud BR	Channels
mosaic	Flash	h264	480x270	452 kbps	29.97	HE-AAC	44.1 kHz	48 kbps	Stereo
low	Flash	h264	768x432	836 kbps	29.97	HE-AAC	44.1 kHz	64 kbps	Stereo
med	Flash	h264	896x504	1436 kbps	29.97	HE-AAC	44.1 kHz	64 kbps	Stereo
high	Flash	h264	960x540	2436 kbps	29.97	HE-AAC	44.1 kHz	64 kbps	Stereo
full	Flash	h264	1280x720	3436 kbps	29.97	HE-AAC	44.1 kHz	64 kbps	Stereo
Mobile	**Type**	**Format**	**Resolution**	**Vid BR**	**Frame Rate**	**Aud Codec**	**Sample Rate**	**Aud BR**	**Channels**
Audio	HLS	---	---	---	---	HE-AAC	44.1 kHz	40 kbps	Stereo
iPhone	HLS	h264	416x234	110 kbps	10	HE-AAC	44.1 kHz	40 kbps	Stereo
iPhone	HLS	h264	416x234	200 kbps	15	HE-AAC	44.1 kHz	40 kbps	Stereo
iPhone	HLS	h264	416x234	400 kbps	29.97	HE-AAC	44.1 kHz	40 kbps	Stereo
iPhone	HLS	h264	640x360	600 kbps	29.97	HE-AAC	44.1 kHz	40 kbps	Stereo
iPad	HLS	h264	640x360	1200 kbps	29.97	HE-AAC	44.1 kHz	40 kbps	Stereo
iPad	HLS	h264	960x540	1800 kbps	29.97	HE-AAC	44.1 kHz	40 kbps	Stereo
Connect Devices	**Type**	**Format**	**Resolution**	**Vid BR**	**Frame Rate**	**Aud Codec**	**Sample Rate**	**Aud BR**	**Channels**
OTT	HLS	h264	1024x576	1200 kbps	29.97	HE-AAC	44.1 kHz	96 kbps	Stereo
OTT	HLS	h264	1024x576	2500 kbps	29.97	HE-AAC	44.1 kHz	96 kbps	Stereo
OTT	HLS	h264	1280x720	3500 kbps	29.97	HE-AAC	44.1 kHz	96 kbps	Stereo

Other Alternatives

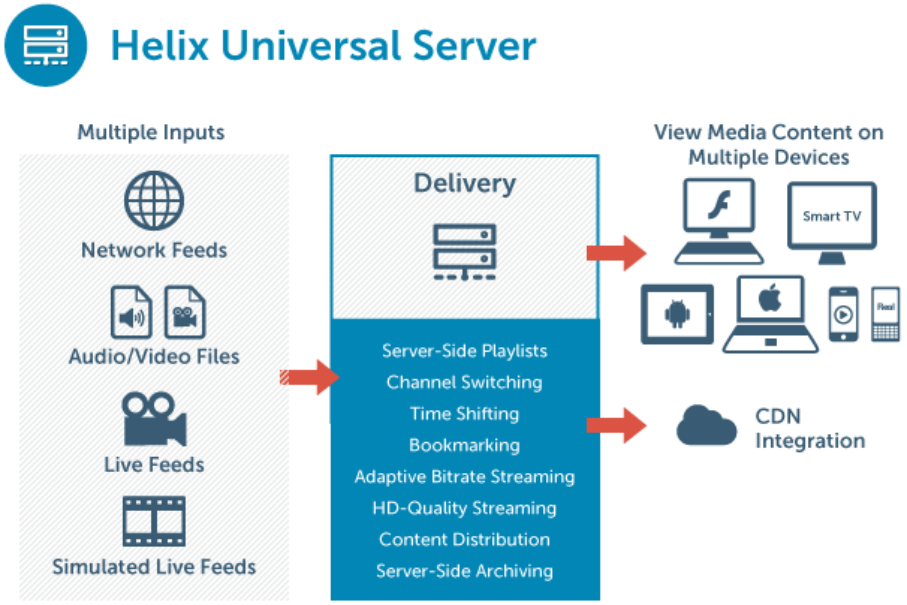


- Create custom encoding groups
  - Approach taken by most very large streamers
  - Most expensive costwise
- Create single set of profiles and transmux

Transmuxing Simplifies Multiple Platform Support

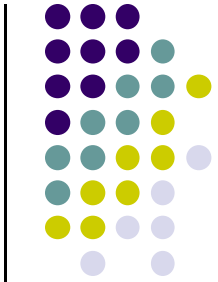
- Transmuxing-dynamic repackaging of streams for different platforms

One stream (or group of streams) in



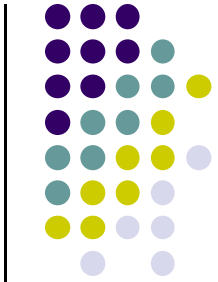
- Flash
- Apple
- Android
- OTT out

Transmuxing



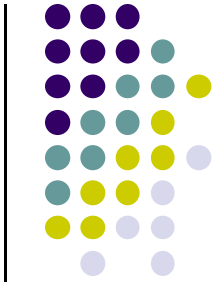
- Not re-encoding
  - “re-wrapping” file into a different container format
  - Creating necessary manifest files
  - Delivering files to HTTP server
  - All in real time

Transmux Providers



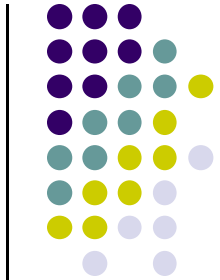
- Media Servers
  - Adobe Media Server
  - RealNetworks Helix
  - Wowza Media Server
- Cloud Services
  - Azure Media Services
- CDNs
  - Akamai
- Many service providers transmux inhouse
  - iStreamPlanet

Perspective



- Key point: If you want to transmux, need to use single set of files for all targets
  - Must be all baseline for streams targeted for mobile
  - If so, can use one set of files to deliver to all three groups
- Bottom line: If serving multiple targets, must consider transmuxing

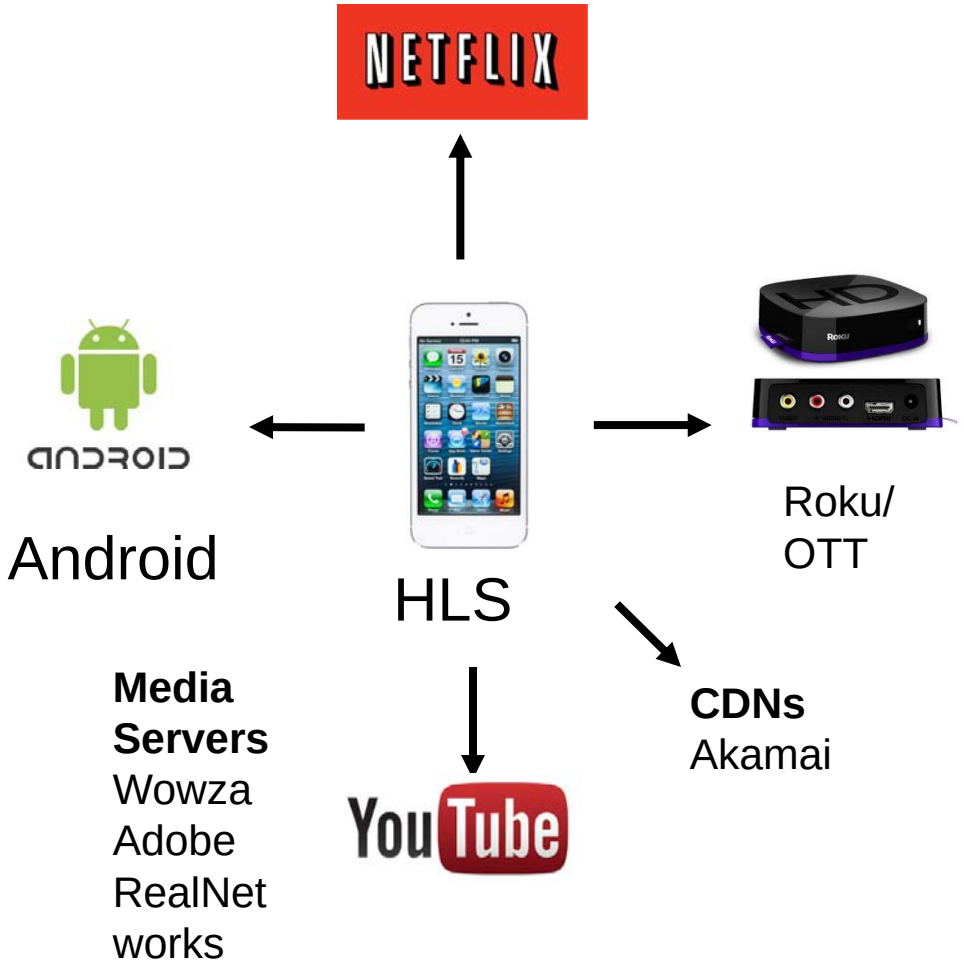
DASH



- Clearly the future
- What's holding it up?
  - Lack of general-purpose support
    - Flash
    - HTML5 (coming)
    - Android
    - iOS

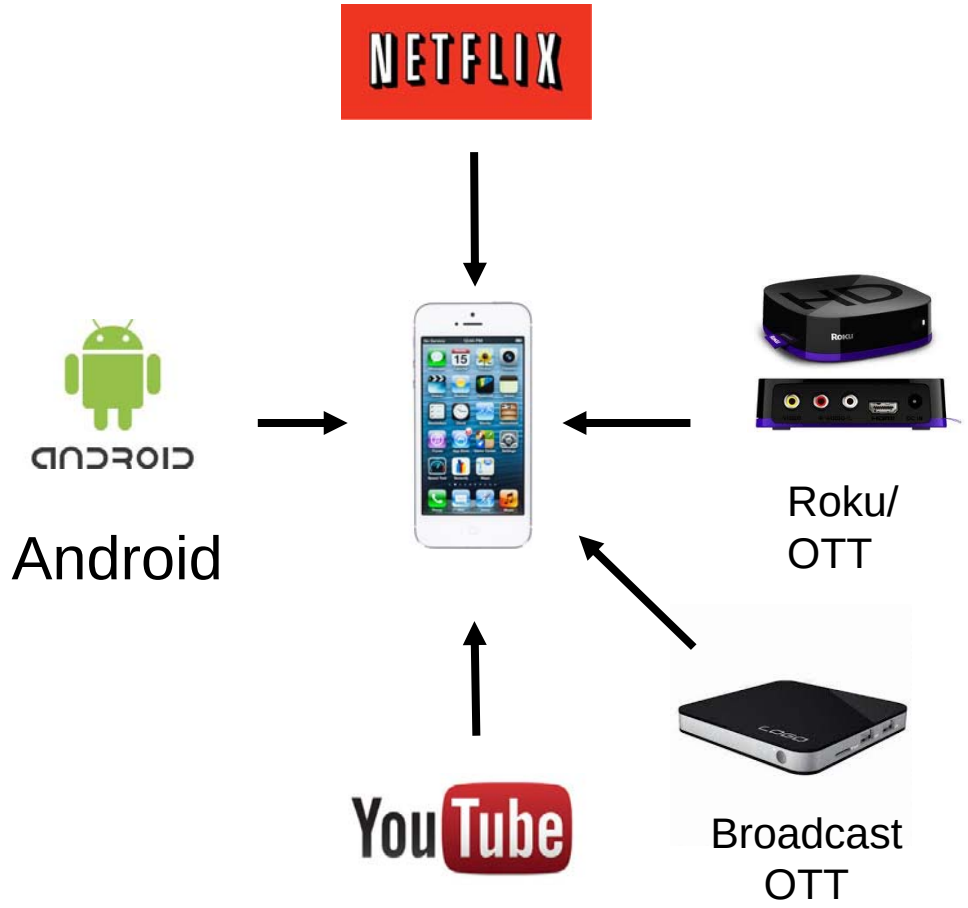
Evolution of HLS support

- iOS so successful that video distributors and other platforms had to support
- HLS still only technology to natively reach iOS
- Tons of content in HLS format
- Apple gained format control they may not want to give up

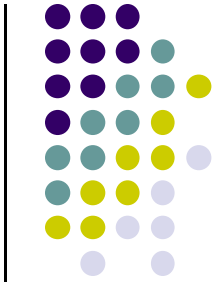


DASH will have to Forced on Apple

- Bottom Line:
  - No content publisher will abandon HLS in foreseeable future
  - Content in DASH format will begin to accumulate
  - Risk that iOS experience may degrade because HLS content not cached
    - Seems down the road



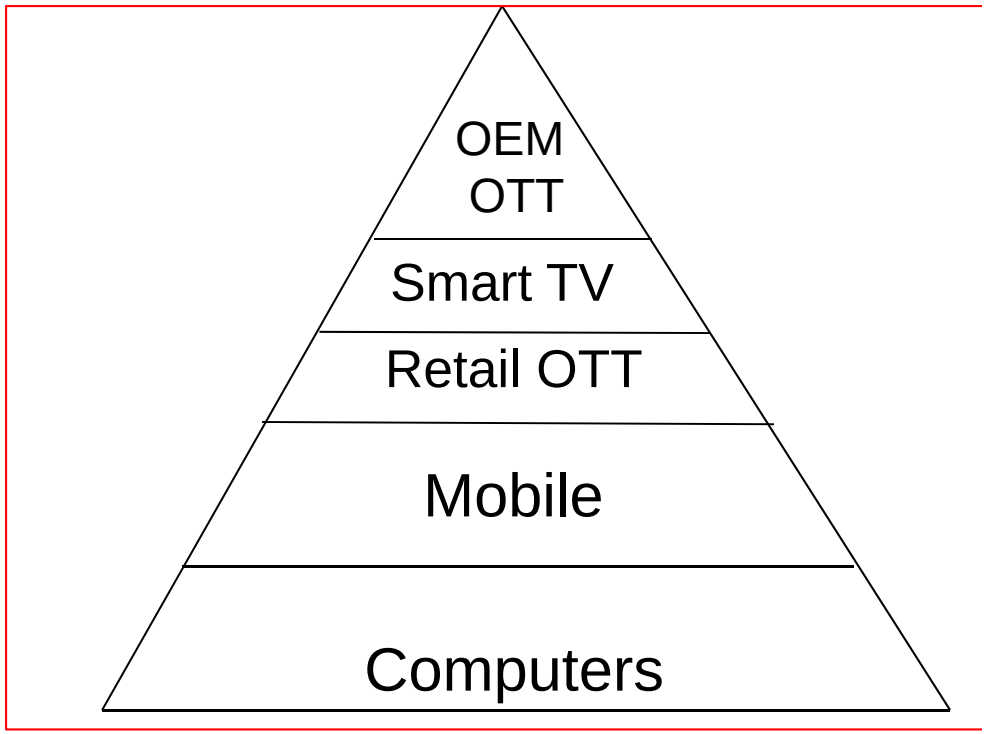
Who's Using DASH Today?



- Companies with R&D budgets larger than yours
  - YouTube, Netflix, Hulu – directly
  - Soon – Primetime/Azure customers
- Why?
  - Greater reach – tippy top of pyramid
  - One common format for a range of deployments
  - Access to common encryption formats

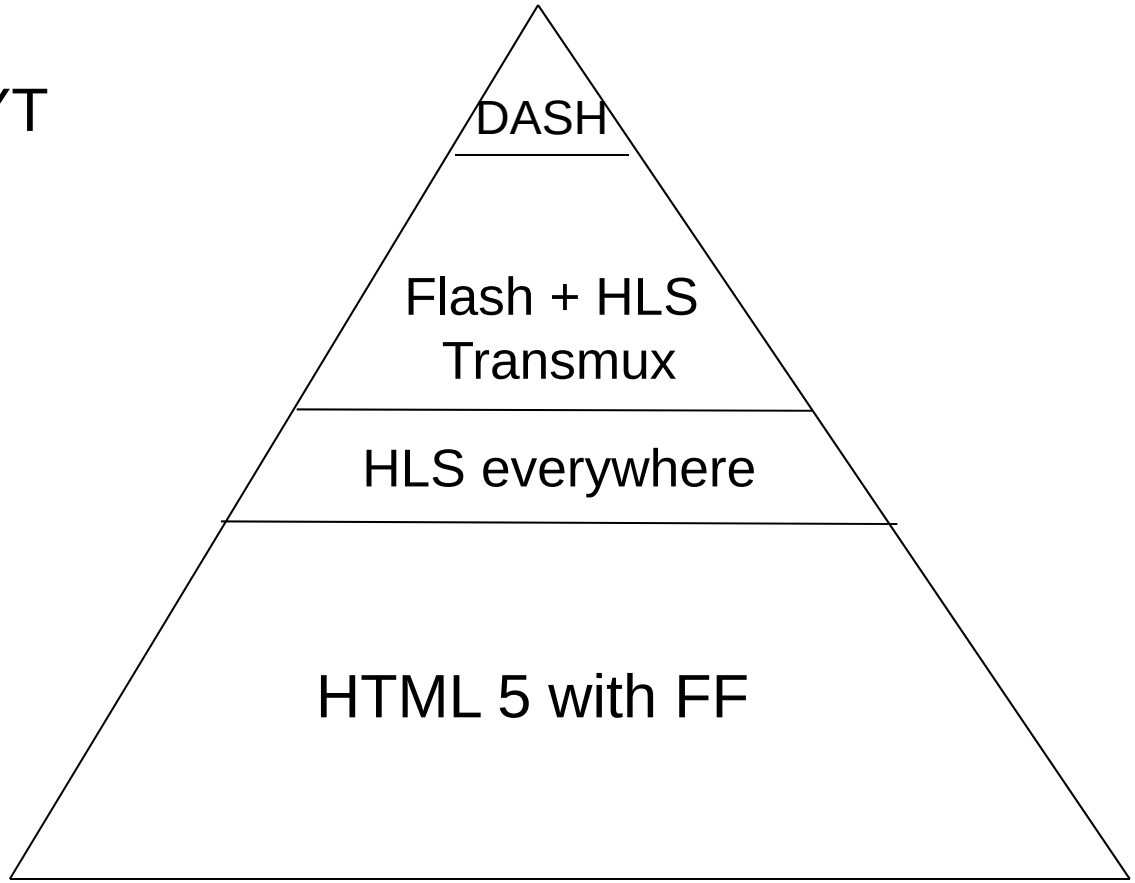
DASH

- Combination of custom desktop players and apps
  - If rolling your own
  - Use Primetime/Azure
- Reach
  - Everywhere
- Cost
  - If you have to ask
- Complexity: significant



Bottom Line

Netflix, YT



Mom & Pop websites

Questions?