W4: KEY ENCODING SKILLS, TECHNOLOGIES TECHNIQUES
STREAMING MEDIA EAST - 2019

Jan Ozer
www.streaminglearningcenter.com
jozer@mindspring.com/
276-235-8542
@janozer

Agenda

- Introduction
- Lesson 1: Delivering to Computers, Mobile, OTT, and Smart TVs
- Lesson 2: Codec review
- Lesson 3: Delivering HEVC over HLS
- Lesson 4: Per-title encoding
- Lesson 5: How to build encoding ladder with objective quality metrics
- Lesson 6: Current status of CMAF
- Lesson 7: Delivering with dynamic and static packaging

Lesson 1: Delivering to Computers, Mobile, OTT, and Smart TVs

- Computers
- Mobile
- OTT
- Smart TVs

Choosing an ABR Format for Computers

- Can be DASH or HLS
- Factors
  - Off-the-shelf player vendor (JW Player, Bitmovin, THEOPlayer, etc.)
  - Encoding/transcoding vendor

Choosing an ABR Format for iOS

- Native support (playback in the browser)
  - HTTP Live Streaming
- Playback via an app
  - Any, including DASH, Smooth, HDS or RTMP Dynamic Streaming

iOS Media Support

	Native	App
Codecs	H.264 (High, Level 4.2), HEVC (Main10, Level 5 high)	Any
ABR formats	HLS	Any
DRM	FairPlay	Any
Captions	CEA-608/708, WebVTT, IMSC1	Any
HDR	HDR10, DolbyVision	?

http://bit.ly/hls_spec_2017

iOS Encoding Ladders

H.264

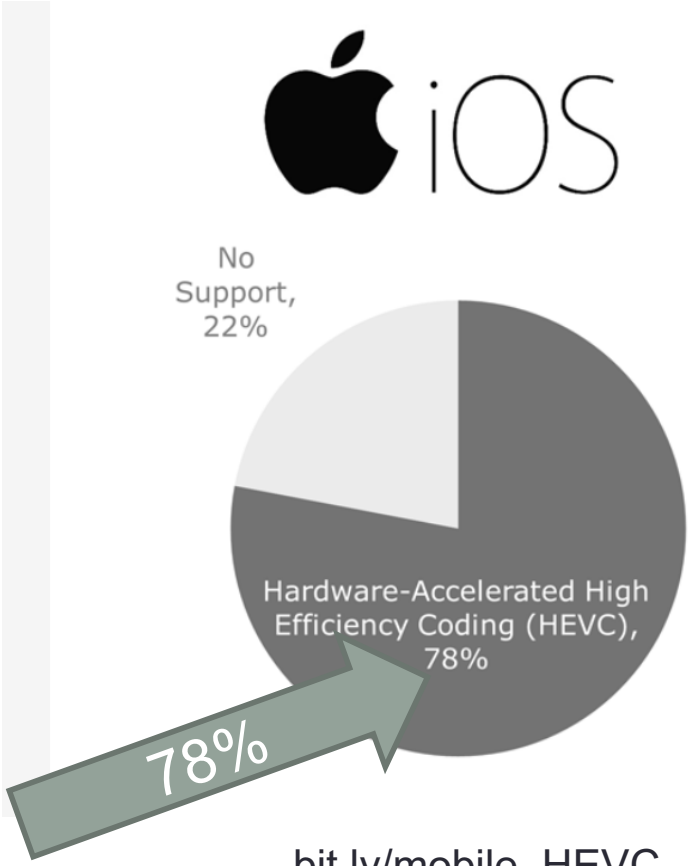
16:9 aspect ratio	H.264/AVC	Frame rate
416 x 234	145	≤ 30 fps
640 x 360	365	≤ 30 fps
768 x 432	730	≤ 30 fps
768 x 432	1100	≤ 30 fps
960 x 540	2000	same as source
1280 x 720	3000	same as source
1280 x 720	4500	same as source
1920 x 1080	6000	same as source
1920 x 1080	7800	same as source

HEVC

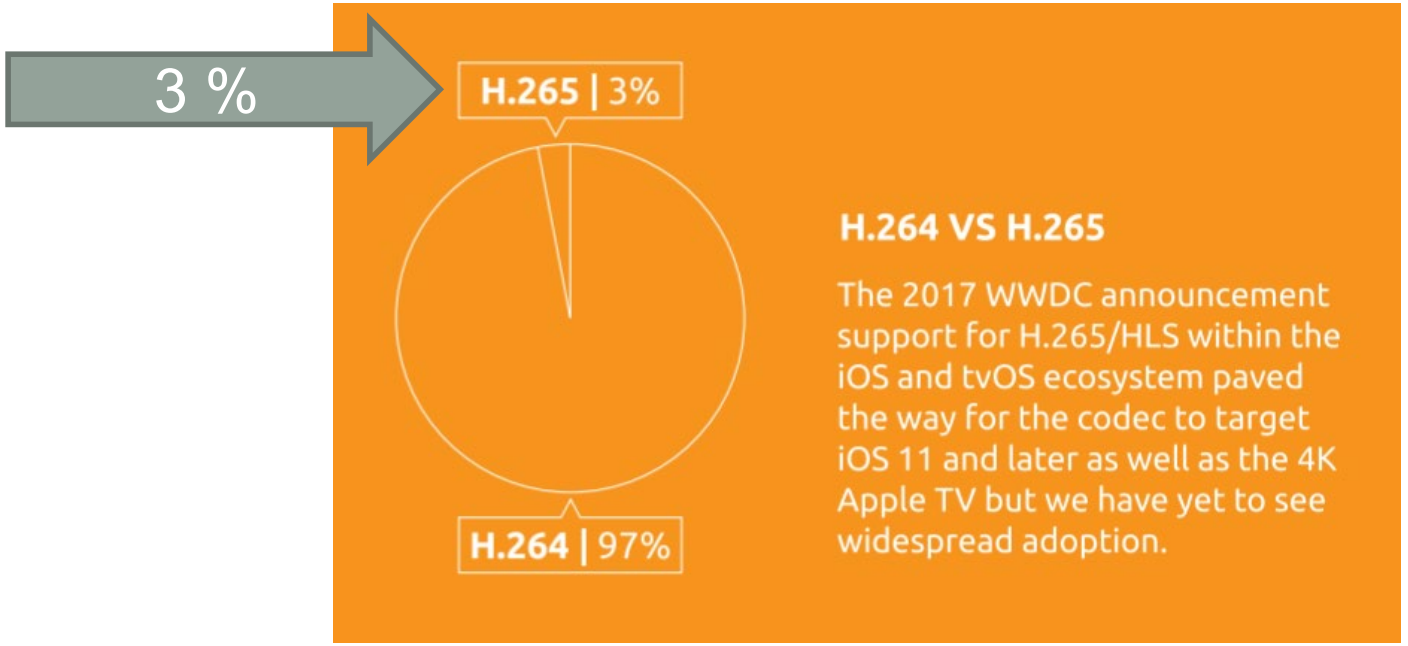
16:9 aspect ratio	HEVC/H.265 30 fps	HDR (HEVC) 30 fps	Frame rate
640 x 360	145	160	≤ 30 fps
768 x 432	300	360	≤ 30 fps
960 x 540	600	730	≤ 30 fps
960 x 540	900	1090	≤ 30 fps
960 x 540	1600	1930	same as source
1280 x 720	2400	2900	same as source
1280 x 720	3400	3850	same as source
1920 x 1080	4500	5400	same as source
1920 x 1080	5800	7000	same as source
2560 x 1440	8100	9700	same as source
3840 x 2160	11600	13900	same as source
3840 x 2160	16800	20000	same as source

http://bit.ly/hls_spec_2017

HEVC Hardware Support - iOS



bit.ly/mobile_HEVC



http://bit.ly/glob_med_2019

Android: Codec and ABR Format Support

Version	Codename	API	Distribution
2.3.3 - 2.3.7	Gingerbread	10	0.2%
4.0.3 - 4.0.4	Ice Cream Sandwich	15	0.3%
4.1.x	Jelly Bean	16	1.1%
4.2.x		17	1.5%
4.3		18	0.4%
4.4	KitKat	19	7.6%
5.0	Lollipop	21	3.5%
5.1		22	14.4%
6.0	Marshmallow	23	21.3%
7.0	Nougat	24	18.1%
7.1		25	10.1%
8.0	Oreo	26	14.0%
8.1		27	7.5%

http://bit.ly/And_ver
(from 10/26/2018 – not updated)

Codecs	ABR
VP8 (2.3+) ↓	
H.264 (3+) ↓	HLS (3+) ↓
VP9 (4.4+) ↓	DASH 4.4+ Via MSE in Chrome ↓
HEVC (5+) ↓	

- Multiple codecs and ABR technologies
  - Serious cautions about HLS
  - DASH now close to 97%
- HEVC
  - Main Profile Level 3 – mobile
    - 960×540@30.0
    - Hardware support probably exceeds this
  - Main Profile – Level 4.1 – Android TV
    - 2,048×1,080@60.0

http://bit.ly/androidvideospecs

Android Media Support

	Native	App
Codecs	H.264, VP8, VP9, HEVC	Any
ABR formats	DASH, HLS	Any
DRM	Widevine	Any
Captions	Embedded 608/607 SRT	Any
HDR	Dolby-Vision, HDR10, VP9-HLG, VP9-PQ	?

https://developer.android.com/guide/topics/media/media-formats

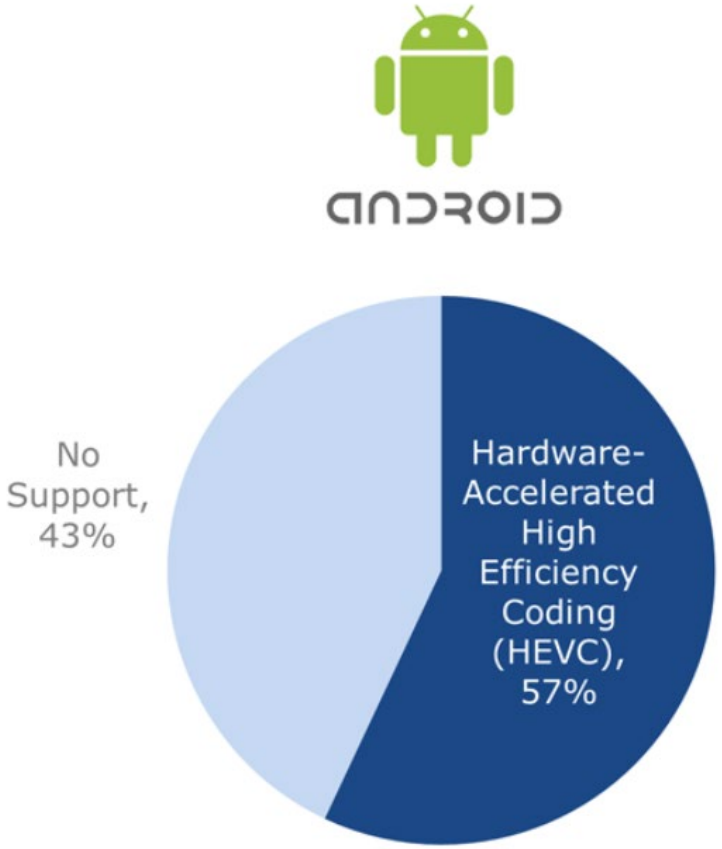
Android Recommended Encoding Ladders

	SD (Low quality)	SD (High quality)	HD 720p (N/A on all devices)
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

- Software only – most devices support hardware
- Most encoding ladders much like iOS

https://developer.android.com/guide/topics/media/media-formats

HEVC Hardware Support - Android



bit.ly/mobile_HEVC

- iOS playback more extensive but little penetration
- Seems that VP9 is more favored than HEVC for Android delivery

Adaptive Streaming to OTT

- Who matters
- Roku
- Apple TV
- Chromecast
- Amazon Fire TV

Who Matters?

STREAMING MEDIA DEVICE US MARKET SHARE

Source : 2018 PARKS ASSOCIATES

http://bit.ly/glob_med_2019

OTT Platform-Format Support

OTT Platforms	Smooth Streaming	HLS	DASH
Roku (bit.ly/roku_vid)	Yes	Yes	Yes
Amazon Fire TV (https://amzn.to/2L8dCdp)	Yes	Yes	Yes (?)
ChromeCast (http://bit.ly/GCast_Media)	Yes	Yes	Yes
Apple TV (bit.ly/AppleTV_recs)	No	Yes	No

Notes:

- Roku 4 and Roku4 TVs supports HEVC and VP9
- Fire TV Gen 2 supports HEVC
- Fire TV Supports VP9
- Most recent Apple TV specs do support CMAF

OTT Platform Codec Support

OTT Platforms	H264	HEVC	VP9	Other
Roku (bit.ly/roku_vid)	Yes	Yes	Yes	None
Amazon Fire TVInsignia HD (https://amzn.to/2L8dCdp)	Yes	Yes	Yes	VP8, H.263, MPEG-2/4
ChromeCast Ultra (http://bit.ly/GCast_Media)	Yes	Yes	Yes	VP8, HDR10, DolbyVision
Apple TV (bit.ly/AppleTV_recs)	Yes	Yes	No	None

OTT Platform DRM Support

OTT Platforms	PlayReady	Widevine	FairPlay	Other
Roku (bit.ly/roku_vid)	Smooth/ DASH	DASH (Beta)	No	Adobe, Verimatrix, AES-128
Amazon Fire TV Insignia HD (https://amzn.to/2L8dCdp)	Yes	Yes	No	HDCP 2.2
ChromeCast (http://bit.ly/GCast_Media)	(DASH/ Smooth)	DASH/HLS	No	AES128, SAMPLE AES
Apple TV (bit.ly/AppleTV_recs)	No	No	Yes	SAMPLE-AES

OTT Platform HDR Support

OTT Platforms	Dolby Vision	HDR 10/10+	HLG	Other
Roku (bit.ly/roku_vid)	No?	Yes/No	No	No
Amazon Fire TV Stick 4K (https://amzn.to/2L8dCdp)	Yes	Yes/Yes	Yes	No
ChromeCast (http://bit.ly/GCast_Media)	Yes	Yes/No	No	No
Apple TV (bit.ly/AppleTV_recs)	Yes	Yes	No	No

Roku Video Specs

- Highlights
  - Codecs – H.264, HEVC, VP9
  - ABR – HLS, DASH, Smooth
  - CMAF – Yes!
  - DRM – PlayReady, Widevine, AES-128
  - Bitrate max
    - H.264 – 10 Mbps
    - HEVC/VP9 – 40 mbps
  - Peak video bitrate (VBR)
    - 1.5x average (so 150% constrained VBR)

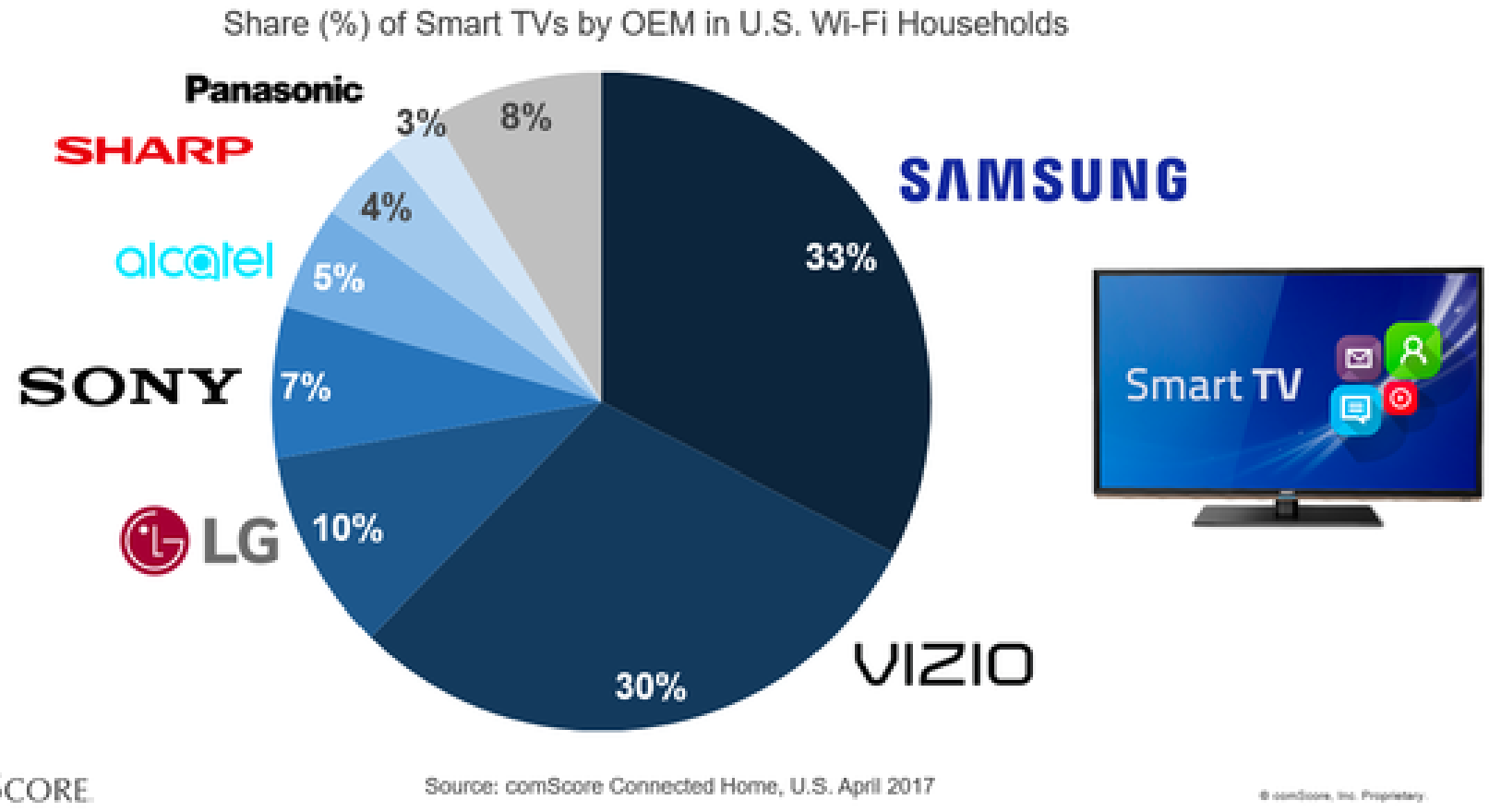
bit.ly/roku_vid

	H.264	HEVC (H.265) [1]	VP9 [1]
Aspect Ratio [2]	Various	Various	Various
Dimension	Various up to 1920x1080	Various up to 3840x2160	Various up to 3840x2160
Video File Format	`.mp4`, `.mov`, `.m4v`, `.mkv`	`.mp4`, `.mov`, `.m4v`, `.mkv`	`.webm`, `.mkv`
Manifest File Format	HLS: `.m3u8` Smooth: `.ism` Dash: `.mpd`	HLS: `.m3u8` Smooth: `.ism` Dash: `.mpd`	Dash: `.mpd`
Stream Format [3]	HLS: `"hls"` Smooth: `"ism"` Dash: `"dash"`	HLS: "hls" Smooth: `"ism"` Dash: `"dash"`	Dash: `"dash"`
Input Frame Rate [4]	24p, 25p, 30p, 50p, 60p	24p, 25p, 30p, 50p, 60p	24p, 25p, 30p, 60p[7]
Color Space	Rec.709	Rec.709, Rec.2020	Rec.709, Rec.2020
Profile	main, high	main, main 10	profile 0, profile 2[7]
Level	4.1, 4.2	4.1, 5.0, 5.1	
Video Mode	Constrained VBR	Constrained VBR	Constrained VBR
Average Streaming Video Bit rate	Up to 10Mbps	Up to 40Mbps	Up to 40Mbps
Average USB Video Bit rate	384Kbps – 10Mbps	Up to 40Mbps	Up to 40Mbps
Peak Video Bit rate	1.5x average	1.5x average	1.5x average
Key Frame Interval [5]	< 10s	< 10s	< 10s
DRM	PlayReady for Smooth/DASH Streaming AES-128 bit encryption for HLS Widevine with HLS/CMAF	PlayReady for Smooth/DASH Streaming AES-128 bit encryption for HLS Widevine with HLS/CMAF	PlayReady for DASH Widevine with HLS/CMAF

Adaptive Streaming to Smart TVs

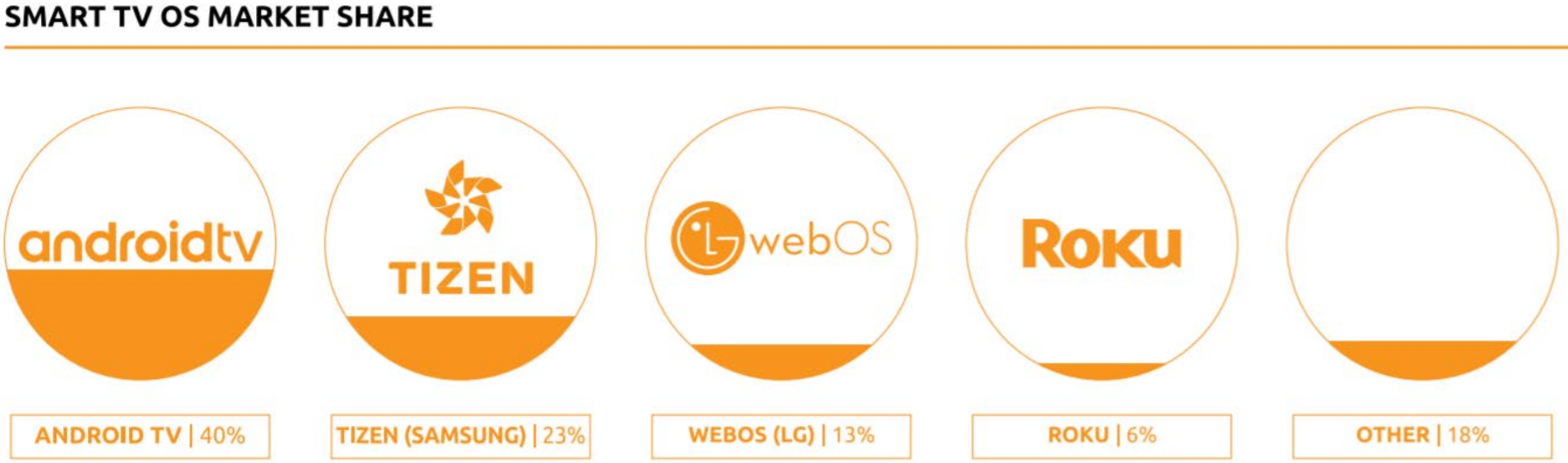
- Format support – general
- Samsung
- Vizio
- Sharp
- Panasonic
- LG
- Smart TV Alliance
- HbbTV

Who Matters – Smart TVs?



http://bit.ly/comscore_SMTV

Who Matters – Smart TV OS Market Share?



http://bit.ly/glob_med_2019

Android TV – Same as Android

	Native
Codecs	H.264, VP8, VP9, HEVC
ABR formats	DASH, HLS
DRM	Widevine
Captions	Embedded 608/607 SRT
	Dolby-Vision, HDR10, VP9-HLG, VP9-PQ

https://developer.android.com/guide/topics/media/media-formats

Samsung Format Support (Tizen)

- Very well defined - bit.ly/tizen_media

	TV 2019	TV 2018
codecs	H.264, HEVC, WMV, VP9	H.264, HEVC, WMV, VP9
ABR formats	DASH, HLS, Smooth	DASH, HLS, Smooth
DRM	Widevine, AES-128, Verimatrix WebClient	Widevine, AES-128, Verimatrix WebClient
Captions	SMI, SRT, SMPTE-TT, WebVTT, 608/708	SMI, SRT, SMPTE-TT, WebVTT, 608/708
HDR		

Vizio Format Support - ?

- Data not publicly available

Sharp Format Support -?

- Data not publicly available

Smart TV Alliance

- Members
  - Panasonic, LG, Toshiba
- Spec – 5.0 (9/2015)
- Codecs
  - H.264, HEVC
- ABR formats (M=mandatory)
  - MPEG DASH, Smooth Streaming, HLS
- DRM
  - PlayReady, Widevine
- Captions
  - W3C TTML

Function	Detail	A/V content
General	HTTP 1.1 with Range request	M
	HTTPS streaming over SSL	M
Adaptive	HTTP Live Streaming	M
	Microsoft Smooth Streaming	M
	MPEG-DASH (ISOBMFF & CENC) according to HbbTV version 1.2.1 profile [26]	M

http://www.smarttv-alliance.org/specification.html

HbbTV 2.01 – 4/16/2016

- Codecs
  - H.264, HEVC
- ABR formats
  - DASH
- DRM
  - CENC
- Captions
  - W3C TTML

HTTP adaptive streaming shall be supported using MPEG DASH as defined in annex E.

bit.ly/HbbTV_201

Questions?

- Questions
-

Should be 2:00

Lesson 2: Codec Review

- Choosing a codec
  - Heritage/cost
  - Playback
  - Quality
  - Encoding time
  - Playback performance

My Focus

- Content publishers (not hardware developers)
  - Primary concern is content royalties decode royalties
- Mainstream codecs
  - There are business cases for V-Nova PERSEUS and RealMedia HD, but I won't discuss them here

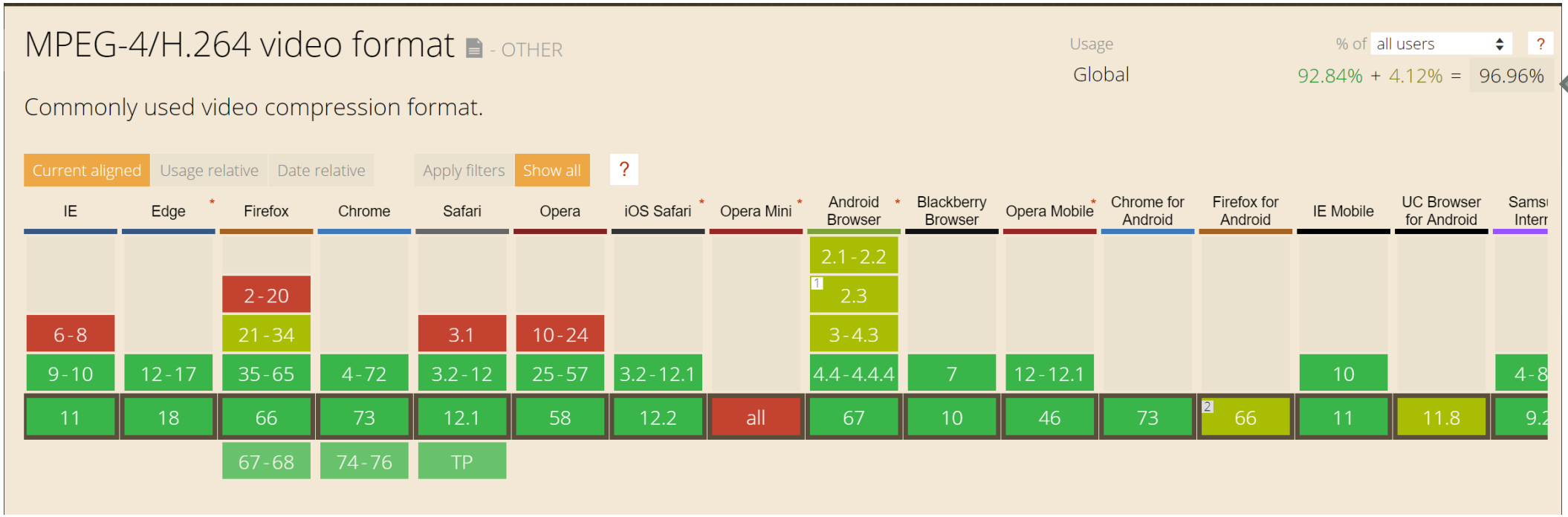
Agenda

- Why do we change codecs?
- To make money
  - Enter new markets (HDR, ultra-low bandwidth)
  - Improve QoE in existing markets
- Factors in decision
  - Codec reach
  - Codec features
  - Codec quality/bitrate
- To save money
  - Reduce bandwidth costs
- Factors:
  - Codec reach
  - Quality
  - Encoding cost
  - Other factors

H.264

- Reach
- Features
- Cost side

H.264: Computers and Mobile



96.96%

- Near ubiquitous reach is H.264's strongest feature

https://caniuse.com/#search=H.264

H.264 OTT/Smart TV

OTT	**H.264**
Roku	Yes
Chromecast	Yes
FireTV	Yes
Apple TV	Yes
Smart TV	
Samsung	Yes
HbbTV	Yes
Smart TV Alliance	Yes

H.264:Live and Live Transcoding

- Ubiquitous live encoders in all shapes and forms
- Transcoding available from Wowza (right), Nimble Streamer, and many cloud, software, and appliance-based encoding vendors
- Very little 4K deployed in live, making H.264 a great option for most live events

Video Output	H.264
Adobe RTMP	✓
RTSP/RTP	✓
MPEG-TS	✓
Apple HLS	✓
MPEG-DASH	✓
Adobe HDS	✓
Microsoft Smooth Streaming	✓
WebRTC(Preview)	✓

H.264 and High Dynamic Range

Technically feasible, but:

- Virtually all HDR devices support HEVC
- Dolby Vision support is 8-bit, not 10-bit
- 4K delivery costs would be excessive

Table 4: Constraints on codec level

Profile ID	Profile Name	BL/EL codec	BL:EL	Dolby Vision level (maximum)	BL/EL codec profile	BL codec level (maximum)	EL codec level (maximum)
4	dvhe.04	10-bit HEVC	1:¼	uhd60	H.265 main10	5.1	4.1
5	dvhe.05	10-bit HEVC	NA	uhd60	H.265 main10	5.1	NA
7	dvhe.07	10-bit HEVC	1:1	fhd60	H.265 main10	High Tier 5.1	High Tier 5.1
			1:¼	uhd60	H.265 main10	High Tier 5.1	High Tier 5.1
8	dvhe.08	10-bit HEVC	NA	uhd60	H.265 main10	5.1	NA
9	dvav.09	8-bit AVC	NA	fhd60	H.264 high	4.2	NA

Note: Profiles 0–3 and 6 are not supported for new applications.

H.264 Cost Side

- Quality – lowest
- Encoding time/cost – least expensive
- Storage – most expensive

H.264: Royalties

- Subscription
  - 100,000 or fewer subscribers/yr = no royalty;
  - 100,000 to 250,000 subscribers/yr = $25,000;
  - 250,000 to 500,000 subscribers/yr = $50,000;
  - 500,000 to 1M subscribers/yr = $75,000;
  - 1M subscribers/yr = $100,000
- Title-by-Title - 12 minutes or less = no royalty;
  - 12 minutes in length = lower of (a) 2% or (b) $0.02 per title
    -

H.264: What's it Cost You?

- Capacity - if delivering over fixed capacity infrastructures
  - According to Netflix: x265 and VP9 up to 40% more efficient, especially at higher resolutions. http://bit.ly/nf_codec
  - So: supporting either VP9, HEVC, or both will expand your capacity and potentially shave bandwidth costs

H.264: What's it Cost You? - QoE

	H.264		HEVC		
Data Rate	Rez	VMAF	Rez	VMAF	Delta
145	234p	21.50	432p	26.56	5.06
365	270p	52.52	540p	65.12	12.61
730	360p	69.10	720p	78.45	9.34
1100	432p	80.61	720p	87.32	6.72
2000	540p	88.02	1080p	92.94	4.92
3000	720p	92.89	1080p	95.86	2.97
4500	720p	95.06	1080p *	97.53	2.47
6000	1080p	96.99	1080p *	97.53	0.54
7800	1080p	97.71	1080p *	97.53	-0.18

1.1 Mbps H.264 is 432p (80.61 VMAF)

4.5 Mbps H.264 is 720p (95.06 VMAF)

7.8 Mbps rung is same quality as 4.5 mbps HEVC

1.1 Mbps HEVC is 720p (87.32 VMAF)

4.5 Mbps HEVC is 1080p (97.53 VMAF) (and is top rung)

- Mobile
  - 1.1 mbps - H.264 - 432p - 80.61 VMAF
  - 1.1 mbps stream HEVC/VP9 - 720p – 87.32 VMAF (noticeably higher quality)
- Living room
  - 4.5 mbps - H.264 - 720p ~ 95.06 VMAF
  - 4.5 mbps - HEVC/VP9 - 1080p – 97.41 VMAF (may be noticeable)
  - Can save 42% bitrate at same quality level

H.264 Scorecard

- Great for reach and features
  - Clearly best codec for legacy viewers
  - Not optimal for HDR
- Cost side
  - Low quality means:
    - high bandwidth costs
    - Limited access to low-bandwidth markets
  - Content royalties an accepted reality

	H.264
Revenue Side	
Reach	
Computers	100%
Mobile with hardware	100%
OTT/Smart TV	100%
Features	
Live	100%
Live transcode	100%
Low latency	100%
HDR	Not optimal (reach of 10-bit AVC unknown)
Cost Side	
Quality	1 - lowest of the bunch
Encoding time	1
Content royalty cost	PPV/Subscripion
FUD Factor	Nokia/Motorola

H.264: The Big Question

- How much longer will you be encoding H.264?

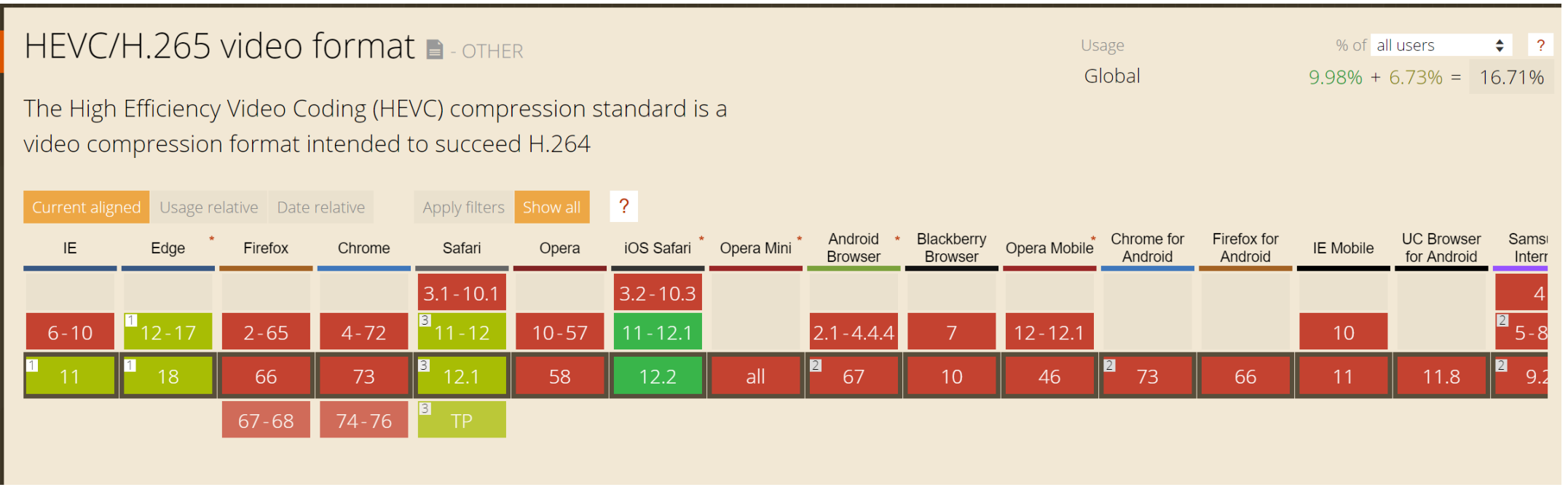
H.264: The Big Question

- How much longer will you be encoding H.264?
  - Forever

HEVC

- Reach
- Features
- Cost side

HEVC: Computers and Mobile



16.71%

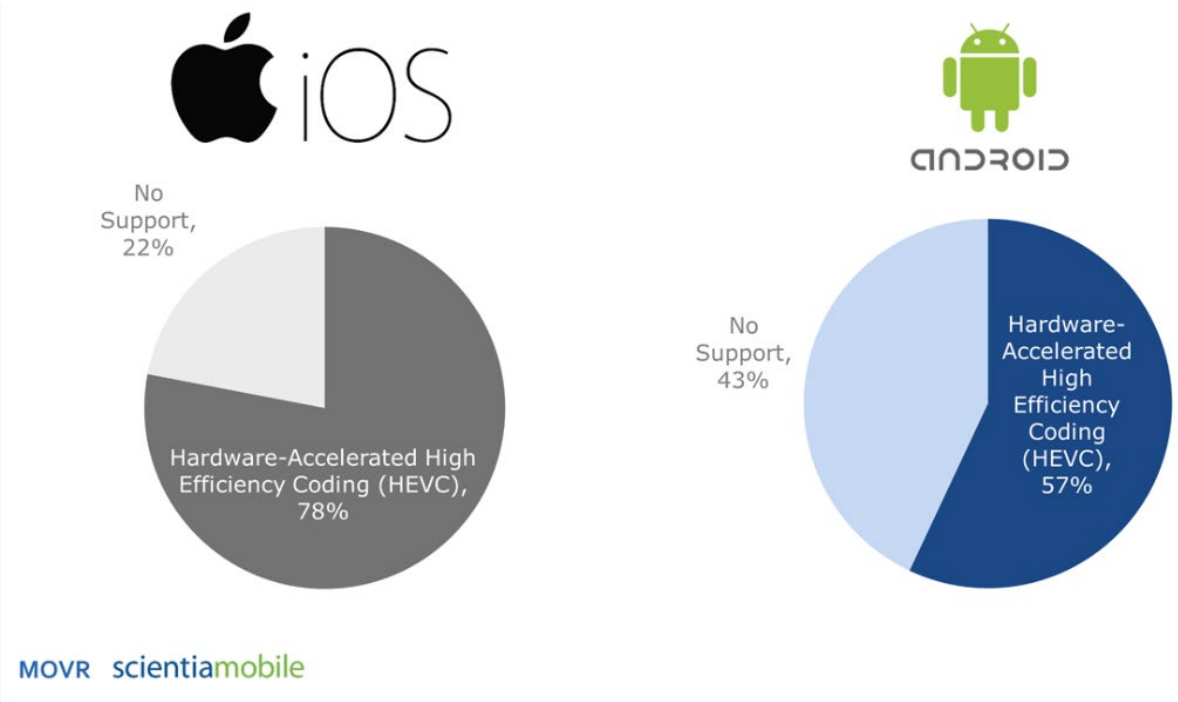
https://caniuse.com/#search=HEVC

- Windows 10
  - Computers with HEVC hardware decode
  - Not HLS
- MacOS - High Sierra
- iOS – 11/Android 5.0 (not HLS)

- No support
  - Computer - Chrome and Firefox
  - Pre MacOS/iOS 11
  - Pre-Windows 10

http://bit.ly/mobile_HEVC

Mobile Hardware Support



- iOS can access HEVC via browser and apps
- Android appears to be app only

HEVC OTT/Smart TV

OTT	**HEVC**	**HEVC**
Roku	Yes	4K capable
Chromecast	Yes	Ultra
FireTV	Yes	2nd Gen
Apple TV	Yes	4K
Smart TV		
Samsung	Yes	2015+
HbbTV	Yes	Yes
Smart TV Alliance	Yes	Yes

HEVC:Live and Live Transcoding

- Some live encoders in all shapes and forms
- Transcoding available from Wowza (right), Nimble Streamer, and many cloud, software, and appliance-based encoding vendors
- HEVC becoming codec of choice for mobile origination according to LiveU

Video Output	H.265	H.264
Adobe RTMP	✔	✔
RTSP/RTP	✔	✔
MPEG-TS	✔	✔
Apple HLS	✔	✔
MPEG-DASH	✔	✔
Adobe HDS		✔
Microsoft Smooth Streaming		✔
WebRTC(Preview)		✔

HEVC and High Dynamic Range

- Technology of choice at this point

Table 4: Constraints on codec level

Profile ID	Profile Name	BL/EL codec	BL:EL	Dolby Vision level (maximum)	BL/EL codec profile	BL codec level (maximum)	EL codec level (maximum)
4	dvhe.04	10-bit HEVC	1:¼	uhd60	H.265 main10	5.1	4.1
5	dvhe.05	10-bit HEVC	NA	uhd60	H.265 main10	5.1	NA
7	dvhe.07	10-bit HEVC	1:1	fhd60	H.265 main10	High Tier 5.1	High Tier 5.1
			1:¼	uhd60	H.265 main10	High Tier 5.1	High Tier 5.1
8	dvhe.08	10-bit HEVC	NA	uhd60	H.265 main10	5.1	NA
9	dvav.09	8-bit AVC	NA	fhd60	H.264 high	4.2	NA

Note: Profiles 0–3 and 6 are not supported for new applications.

HEVC Cost Side

- Quality
  - Assume same quality as H.264 at 60% of the data rate (save 40%)
- Encoding time/cost
  - Assume 2x H.264, 4x for UHD streams
    - Much less if running your own encoding farm
    - More if you're paying retail by the GB or minute
- Storage – assume 60% the cost of H.264 quality

HEVC: Content Royalties

- Three patent groups
  - MPEG LA
  - HEVC Advance
  - Velos – hasn't announced
- Non-affiliated

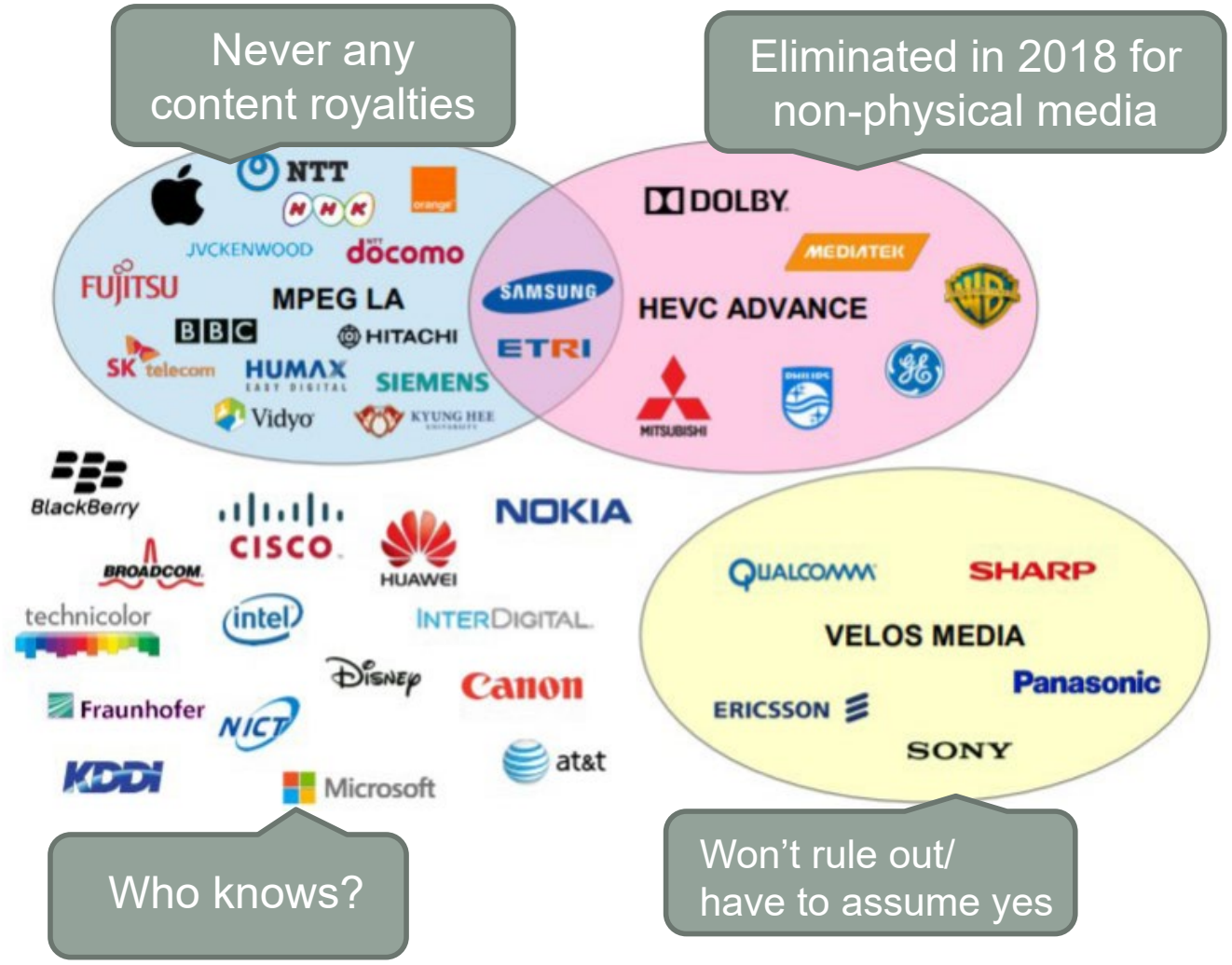


Image by Jonatan Samuelsson, Divideon

What do Steely-Eyed CFOs Hate More than Anything?

HEVC in HLS?

- Two reasons
  - Improved QoE
  - Bandwidth savings
- Bandwidth savings
  - Explored before – higher quality at all bitrates
  - Most noticeable for mobile

	H.264		HEVC		
Data Rate	Rez	VMAF	Rez	VMAF	Delta
145	234p	21.50	432p	26.56	5.06
365	270p	52.52	540p	65.12	12.61
730	360p	69.10	720p	78.45	9.34
1100	432p	80.61	720p	87.32	6.72
2000	540p	88.02	1080p	92.94	4.92
3000	720p	92.89	1080p	95.86	2.97
4500	720p	95.06	1080p *	97.53	2.47
6000	1080p	96.99	1080p *	97.53	0.54
7800	1080p	97.71	1080p *	97.53	-0.18

Bandwidth Savings Breakeven Analysis

- Cost –
  - Fixed – player cost – should be minimal
  - Variable – additional encoding/storage
- Savings
  - Reduced bandwidth costs
  - What are bandwidth savings?
    - Just because HEVC enables same quality as H.264 at 60% of data rate doesn't mean you save 40% of bandwidth

$$\frac{\text{Fixed + Variable Costs}}{\text{Savings/hour}} = \text{Viewing hours to Breakeven}$$

Netflix ISP Index

- Averages 4.06 Mbps during primetime viewing

ISP LEADERBOARD - MARCH 2018

RANK	ISP	SPEED Mbps		PREVIOUS Mbps
1	Comcast	4.06		4.00
2	Verizon - FiOS	4.04		3.97
3	Cox	4.02		3.95
4	Optimum	3.98		3.92
5	Spectrum	3.98		3.90
6	Mediacom	3.94		3.87
7	AT&T - U-verse	3.92		3.87
8	Suddenlink	3.84		3.76
9	Frontier	3.36		3.31
10	CenturyLink	3.36		3.29
11	Windstream	3.30		3.21
12	Verizon - DSL	3.16		3.22
13	AT&T - DSL	2.86		2.87

https://ispspeedindex.netflix.com/country/us/

How ABR Works

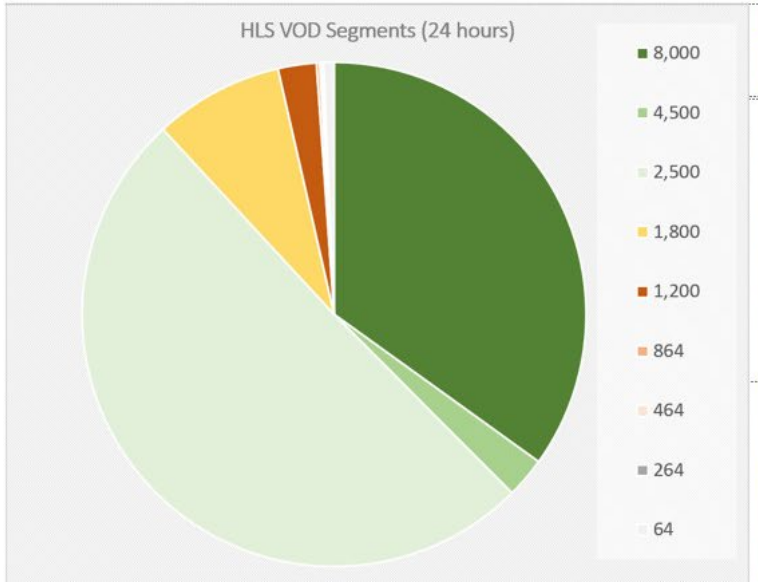
- Netflix averages 4 Mbps in the US during primetime hours
- Assume you can push through 4.5 Mbps
- H.264 would be this stream
- HEVC would be this stream
- No bandwidth savings
- Clearly – just because HEVC cuts bitrates by 40% doesn't mean you cut bandwidth costs by 40%

	H.264	HEVC
Data Rate	Rez	Rez
145	234p	432p
365	270p	540p
730	360p	720p
1100	432p	720p
[illegible]	540p	1080p
[illegible]	720p	1080p
4500	720p	1080p *
6000	1080p	1080p *
7800	1080p	1080p *

H.264

HEVC

How Can You Assess Potential Savings?



HLS VOD (usage over 24 hours)		
Bitrate	HLS VOD Segments (24 hours)	Segment usage
8.000	1.349.530	34,93%
4.500	100.328	2,60%
2.500	1.954.312	50,59%
1.800	320.749	8,30%
1.200	95.102	2,46%
864	8.337	0,22%
464	3.904	0,10%
264	5.023	0,13%
64	25.982	0,67%
Total	**3.863.267**	
Sum of "excessive" bitrates		**6,18%**
Sum of "relevant" bitrates		**93,82%**

- Is your average 4 Mbps because of bandwidth restrictions or because of a mix of SD and HD and UHD?
- Check your log files

- 85% of this client's distribution were the highest quality SD &HD streams
- HEVC/VP9/AV-1 should lead to very significant bandwidth savings

- https://www.linkedin.com/pulse/check-your-dang-log-files-jan-ozer/

HEVC in HLS?

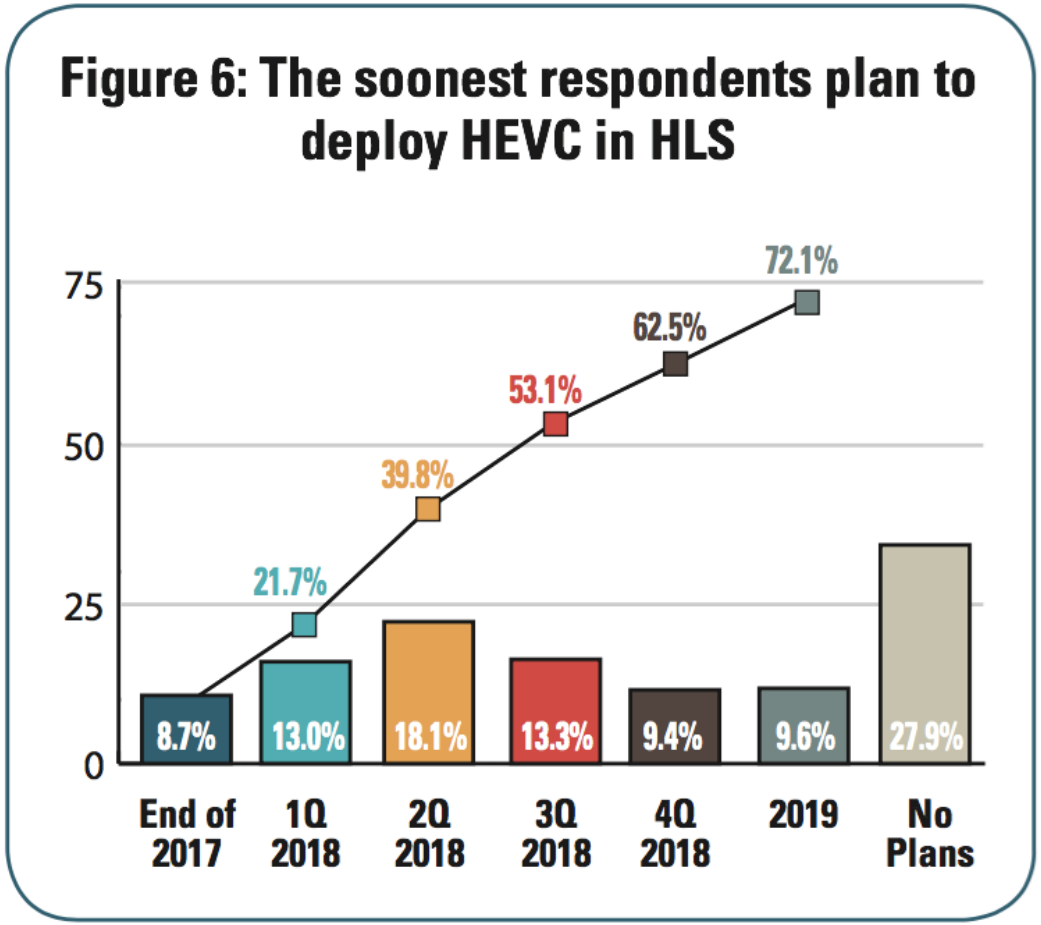
	Bandwidth Cost (per GB)							
Encoding Cost/video hour	**$0.085**	**$0.08**	**$0.06**	**$0.04**	**$0.03**	**$0.03**	**$0.02**	**$0.01**
$3	49	52	69	104	139	167	208	417
$5	82	87	116	174	231	278	347	694
$10	163	174	231	347	463	556	694	1,389
$15	245	260	347	521	694	833	1,042	2,083
$20	327	347	463	694	926	1,111	1,389	2,778
$25	408	434	579	868	1,157	1,389	1,736	3,472
$30	490	521	694	1,042	1,389	1,667	2,083	4,167

- Assumptions
  - H.264 @ 4 Mbps/HEVC @ 2.4 MB
  - Storage not included (~ $1.20/hr for 5 years)
  - No player dev cost (native playback)

- Simple math exercise

Seems Logical; on the Radar Screen

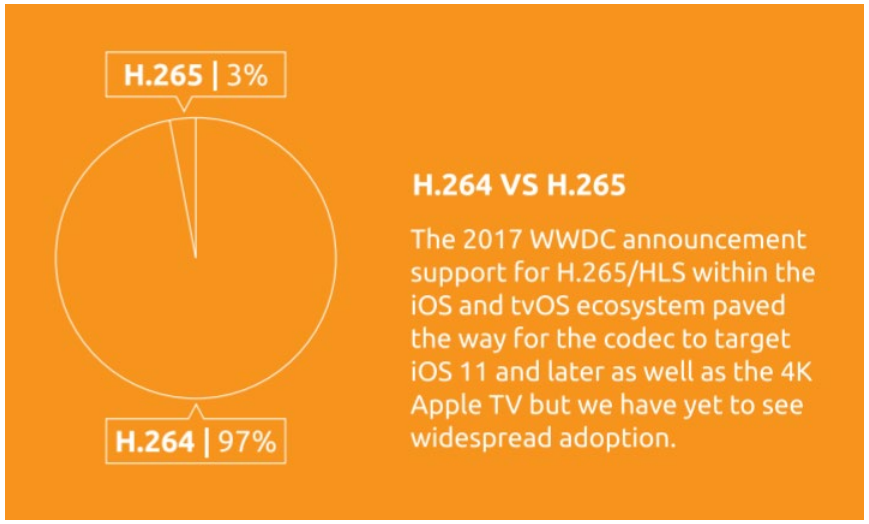
- Streaming Media survey sponsored by Harmonic; published in January 2018
  - 437 *Streaming Media* readers responded
- Substantial interest in supporting HEVC in HLS, particularly in mid-to-late 2018.



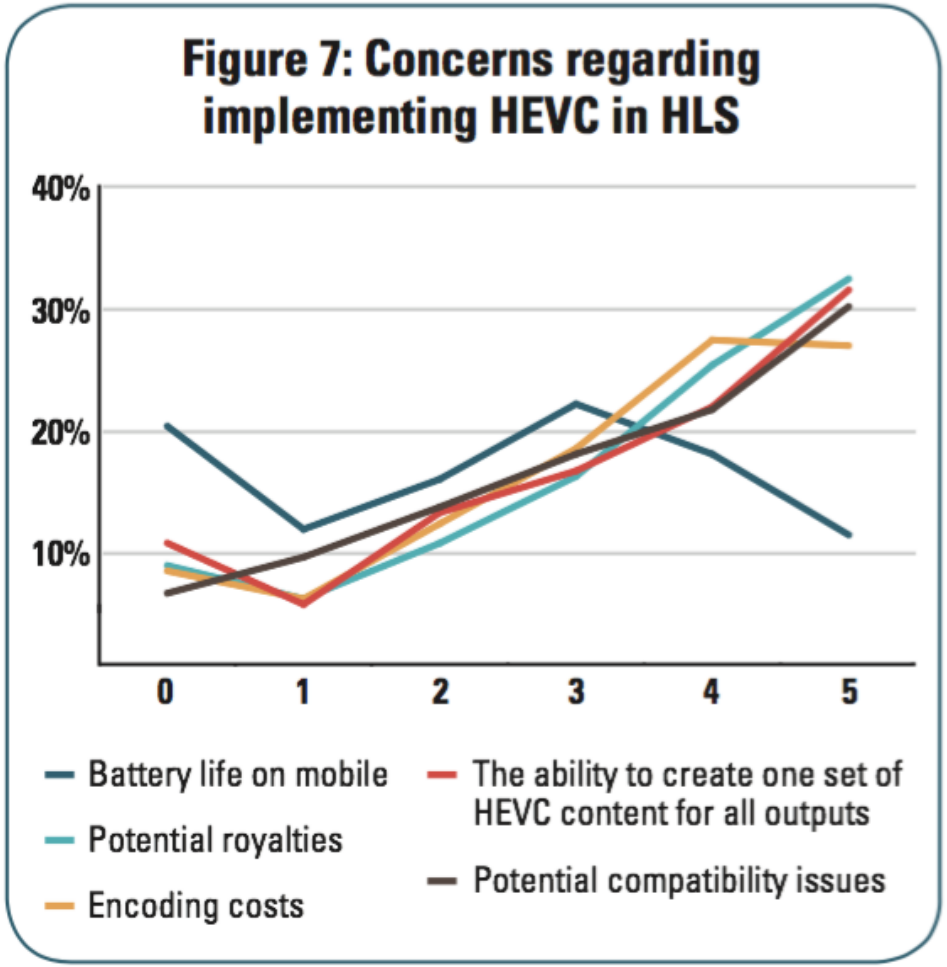
http://bit.ly/HEVC_HLS_Survey

Seems Logical; on the Radar Screen

- Not surprisingly, royalties were the most significant concern
  - 30% rated this concern a 5, which was the strongest



http://bit.ly/glob_med_2019



http://bit.ly/HEVC_HLS_Survey

HEVC Scorecard

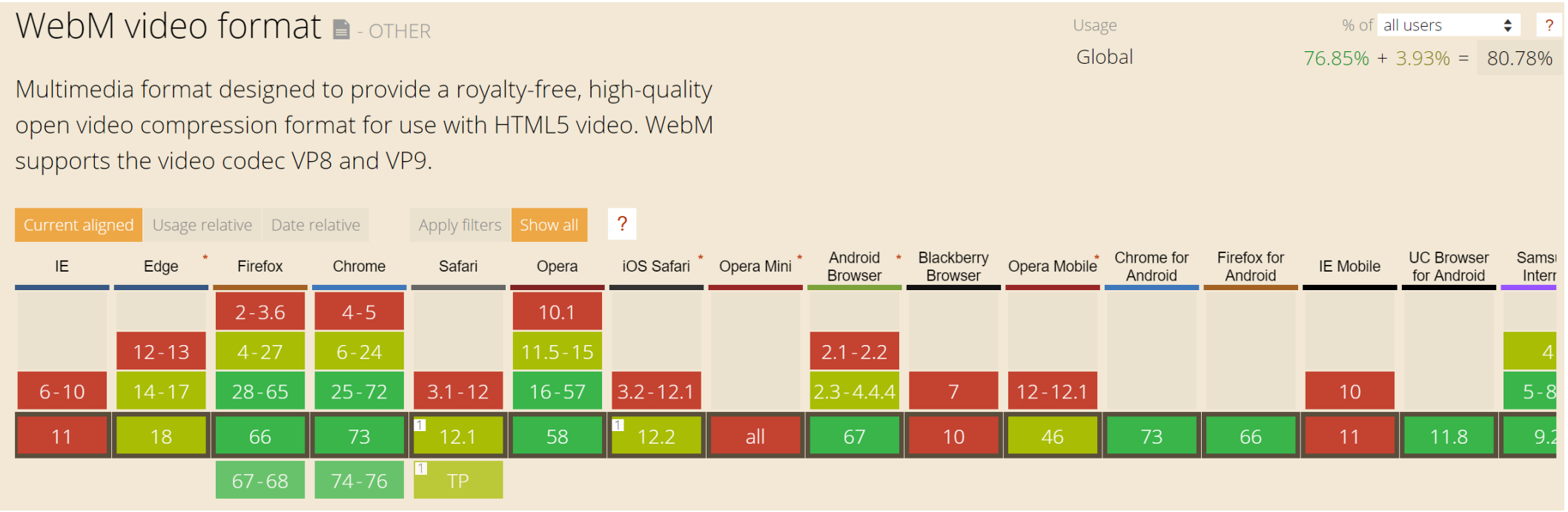
- Potential for content royalty continues to be a huge wet blanket
- Non-starter for browser-based distribution on computers
- Best option for HDR to Smart TVs
  - No brainer
- What about HEVC/HLS?
  - Let's take a look

	H.264	HEVC
Revenue Side		
Reach		
Computers	100%	Safari/Edge
Mobile with hardware	100%	100%
OTT/Smart TV	100%	100%
Features		
Live	100%	Many options
Live transcode	100%	Many options
Low latency	100%	Some options
HDR	Not optimal (reach of 10-bit AVC unknown)	100%
Cost Side		
Quality	1 - lowest of the bunch	H.264 @ 60%
Encoding time	1	~ 4x H.264
Content royalty cost	PPV/Subscripion	Velos?
FUD Factor	Nokia/Motorola	Others not in pool

VP9

- Reach
- Features
- Cost side

VP9 Browser and Mobile (www.caniuse.com)



WebM video format - OTHER

Multimedia format designed to provide a royalty-free, high-quality open video compression format for use with HTML5 video. WebM supports the video codec VP8 and VP9.

Usage: Global — % of all users: 76.85% + 3.93% = 80.78%

Current aligned | Usage relative | Date relative | Apply filters | Show all | ?

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Blackberry Browser	Opera Mobile *	Chrome for Android	Firefox for Android	IE Mobile	UC Browser for Android	Samsung Internet
		2-3.6	4-5		10.1										
	12-13	4-27	6-24		11.5-15			2.1-2.2							4
6-10	14-17	28-65	25-72	3.1-12	16-57	3.2-12.1		2.3-4.4.4	7	12-12.1			10		5-8
11	18	66	73	1 12.1	58	1 12.2	all	67	10	46	73	66	11	11.8	9.2
		67-68	74-76	1 TP											

80.78%

- Strong browser support
  - Chrome, Firefox, Edge
  - Android support
- No support:
  - MacOS/iOS (support above is VP8 for WebRTC)

VP9 Compatibility Matrix

OTT	**H.264**	**HEVC**	**VP9**
Roku	Yes	4K capable	4K capable
Chromecast	Yes	Ultra	Ultra
FireTV	Yes	2nd Gen	2nd Gen
Apple TV	Yes	4K	**No**
Smart TV			
Samsung	Yes	**2015+**	2015+
HbbTV	Yes	**Yes**	**No**
Smart TV Alliance	Yes	**Yes**	**No**

VP9: Live Transcoding - Available for DASH

- Very few live encoding devices (most H.264/HEVC)
- Decent transcoding support from services
  - Growing support for hardware-based transcode (NG Codec)
  - Accelerated software (Intel SVT-VP9)
- Not so much from traditional encoders

Video Output	H.263	VP9
Adobe RTMP		
RTSP/RTP		
MPEG-TS		
Apple HLS		
MPEG-DASH		✔
Adobe HDS		
Microsoft Smooth Streaming		
WebRTC(Preview)		✔

VP9 and High Dynamic Range

- Technically feasible (10-bit+ available)
- Not included in Dolby Vision spec
- Not included in Apple HLS spec
- Is included in HLG spec

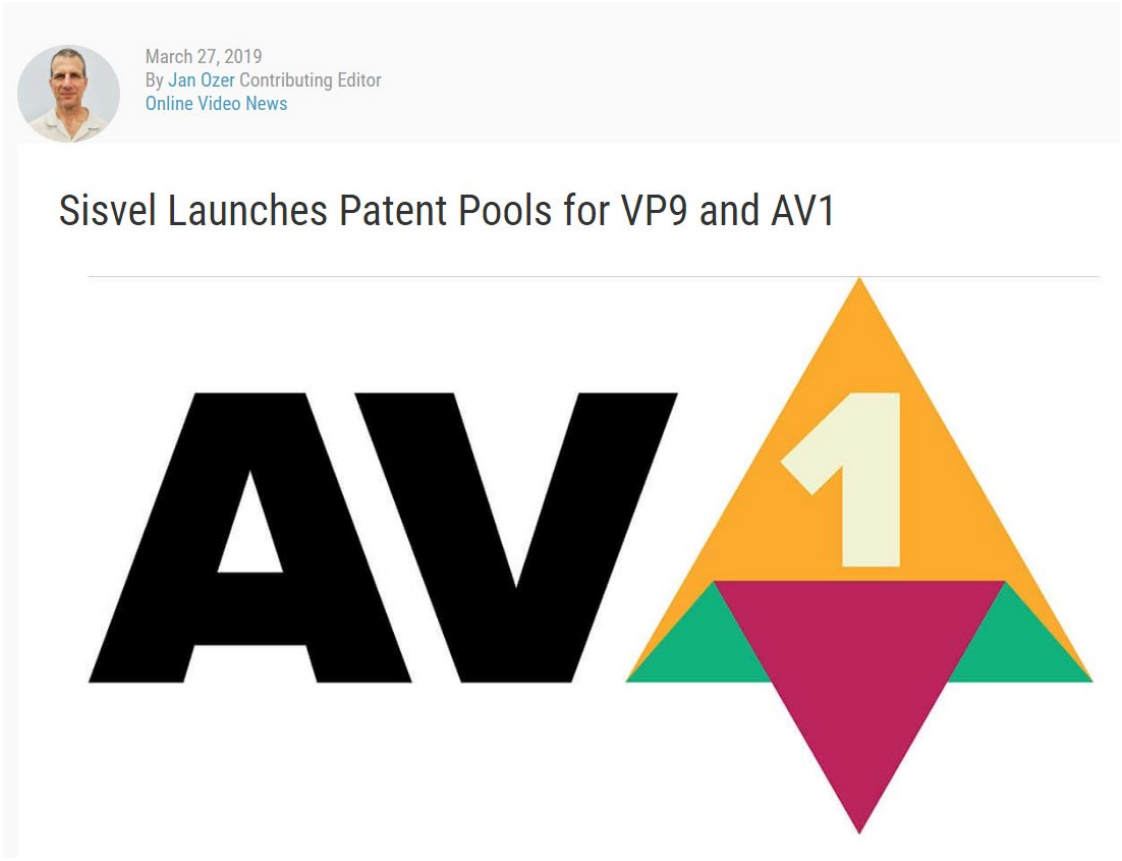
Table 4: Constraints on codec level

Profile ID	Profile Name	BL/EL codec	BL:EL	Dolby Vision level (maximum)	BL/EL codec profile	BL codec level (maximum)	EL codec level (maximum)
4	dvhe.04	10-bit HEVC	1:¼	uhd60	H.265 main10	5.1	4.1
5	dvhe.05	10-bit HEVC	NA	uhd60	H.265 main10	5.1	NA
7	dvhe.07	10-bit HEVC	1:1	fhd60	H.265 main10	High Tier 5.1	High Tier 5.1
			1:¼	uhd60	H.265 main10	High Tier 5.1	High Tier 5.1
8	dvhe.08	10-bit HEVC	NA	uhd60	H.265 main10	5.1	NA
9	dvav.09	8-bit AVC	NA	fhd60	H.264 high	4.2	NA

Note: Profiles 0–3 and 6 are not supported for new applications.

VP9: What's it Cost You?

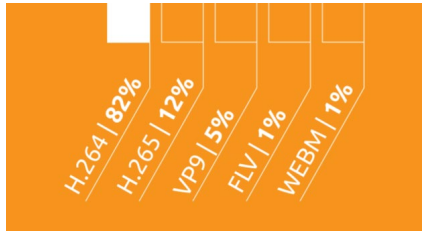
- Royalty free, but no indemnifications from Google
  - Sisvel patent pool for AV1/VP9 and threats from Velos
    - Consumer device only (.24 Euro for VP9/.32 Euro for AV1)
    - No content royalties
    - No cap
    - Software tbd
- Same storage costs at origin as HEVC



bit.ly/sisvel_av1pool

VP9: When to Support VP9 in the Browser?

- Same QoE and bandwidth saving analysis
  - Player development cost needs to be considered
  - Encoding cost should be about 2x H.264
- Penetration to date
  - Some uptake
  - Encoding.com's VP9 production down from 11% in 2016



VP9 STEADY

We reported soft interest in VP9 last year primarily related to finalization of the CMAF spec and forward-looking experimenting with next generation AV1 codec. Interest in VP9 remained steady in 2018, bolstered by a few HLG specific HDR workflows.

http://bit.ly/glob_med_2019

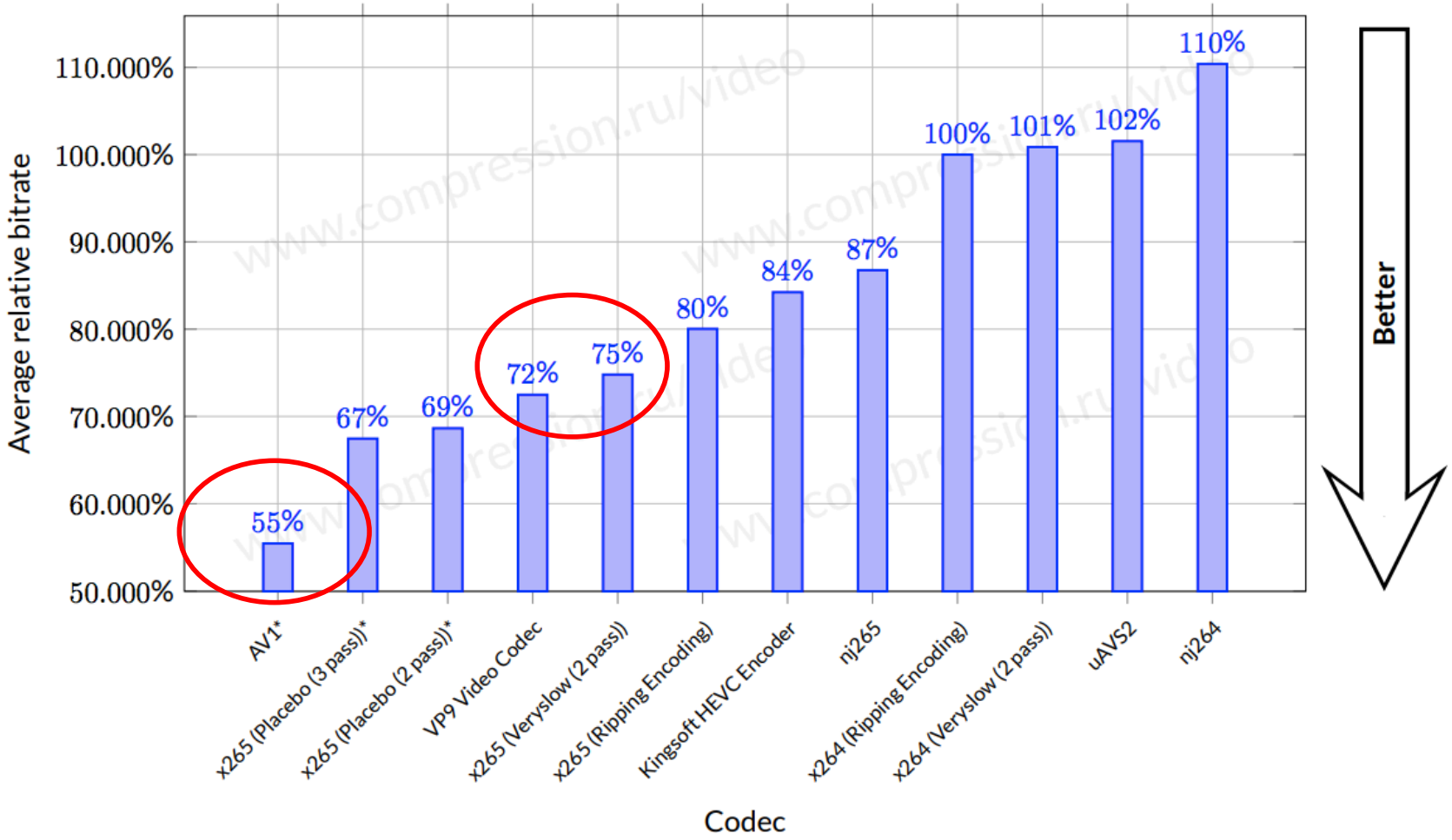
VP9 Scorecard

- Great browser support
- Preferred for Android
- Limited features

	H.264	HEVC	VP9
Revenue Side			
Reach			
Computers	100%	Safari/Edge	95%
Mobile with hardware	100%	100%	50%
OTT/Smart TV	100%	100%	75%
Features			
Live	100%	Many options	Little support
Live transcode	100%	Many options	Wowza & Nimble
Low latency	100%	Some options	unknown
HDR	Not optimal (reach of 10-bit AVC unknown)	100%	No std. support
Cost Side			
Quality	1 - lowest of the bunch	H.264 @ 60%	H.264 @ 60%
Encoding time	1	~ 4x H.264	~2X H.264
Content royalty cost	PPV/Subscripion	Velos?	None
FUD Factor	Nokia/Motorola	Others not in pool	Feels low risk

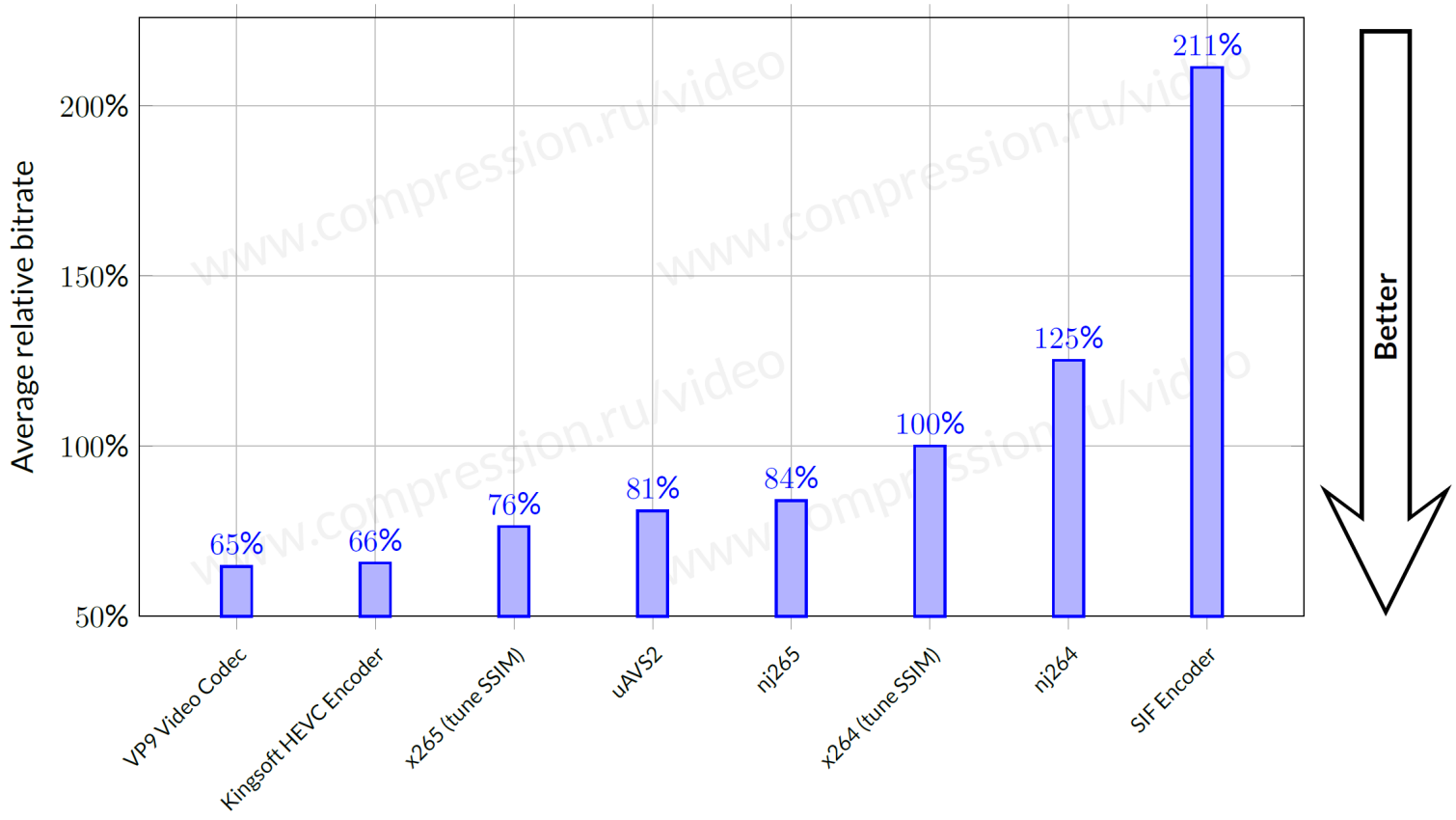
What About HEVC vs. VP9 Quality?

- Moscow State University
- Most likely use case – very close according to SSIM
- AV1 produces same quality as x264 @ 55% of the data rate



New Subjective Comparisons

- Subjectify.us – cloud-sourced viewing:
  - https://youtu.be/ftSbNUM1yMA
- VP9 slightly better than best HEVC alternative



http://www.compression.ru/vi

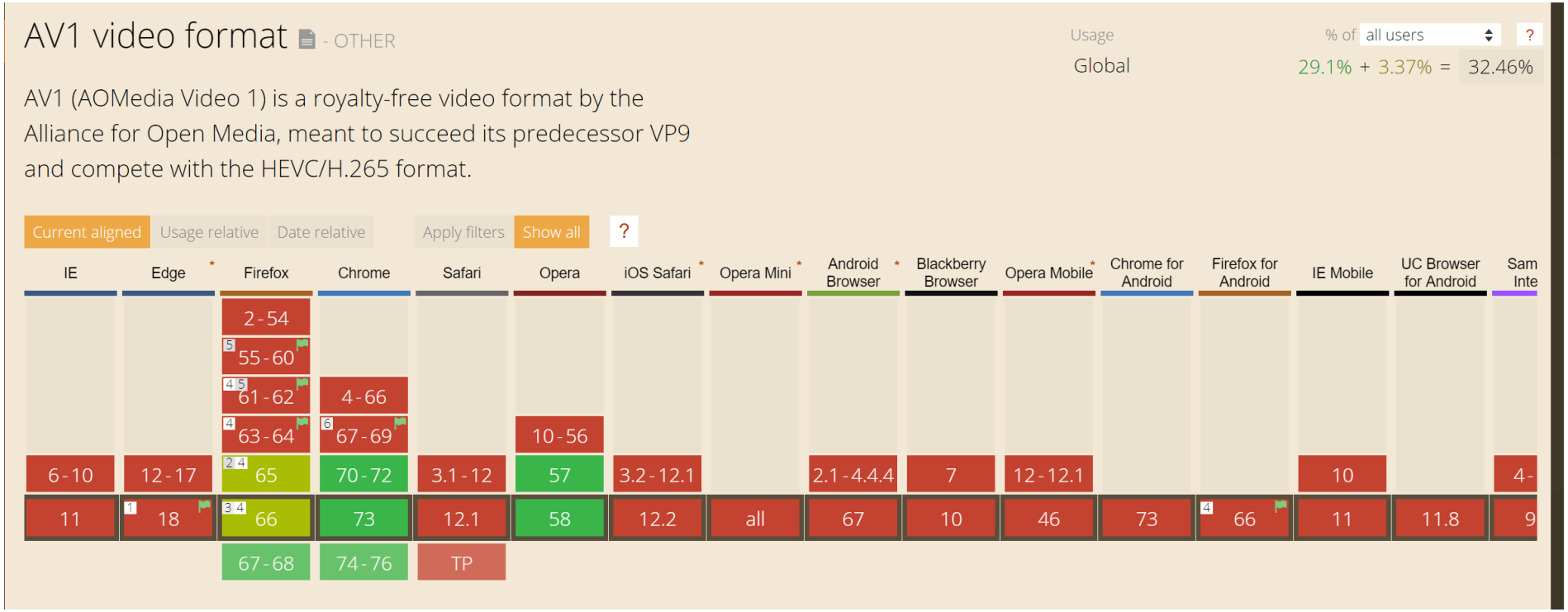
Alliance for Open Media AV1

- What is it?
- Reach
- Features
- Cost side

What is AV1?

- Codec produced by the Alliance for Open Media (AOM)
- Prominent members include:
  - Codec vendors – Google, Microsoft, Mozilla, Cisco
  - Hardware – Intel, NVIDIA, ARM, Broadcom, Ittiam
  - Content – YouTube, Netflix, Amazon, Facebook, Hulu, BBC
  - Infrastructure – Bitmovin, Ateme, IBM
  - Technically sophisticated group (hold that thought)
- Bitstream frozen soon after NAB 2018

AV1 Browser and Mobile (www.caniuse.com)

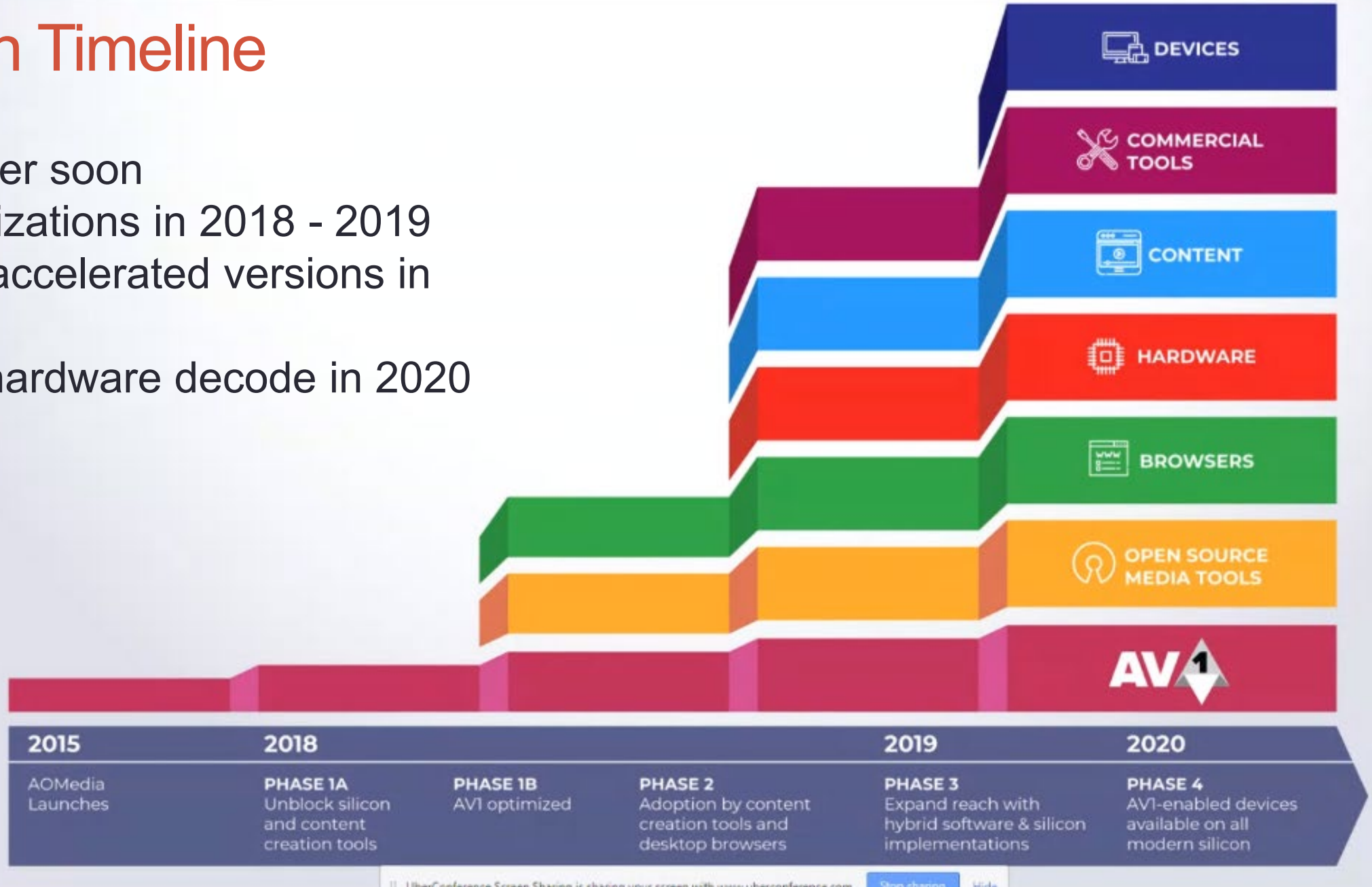


32.46

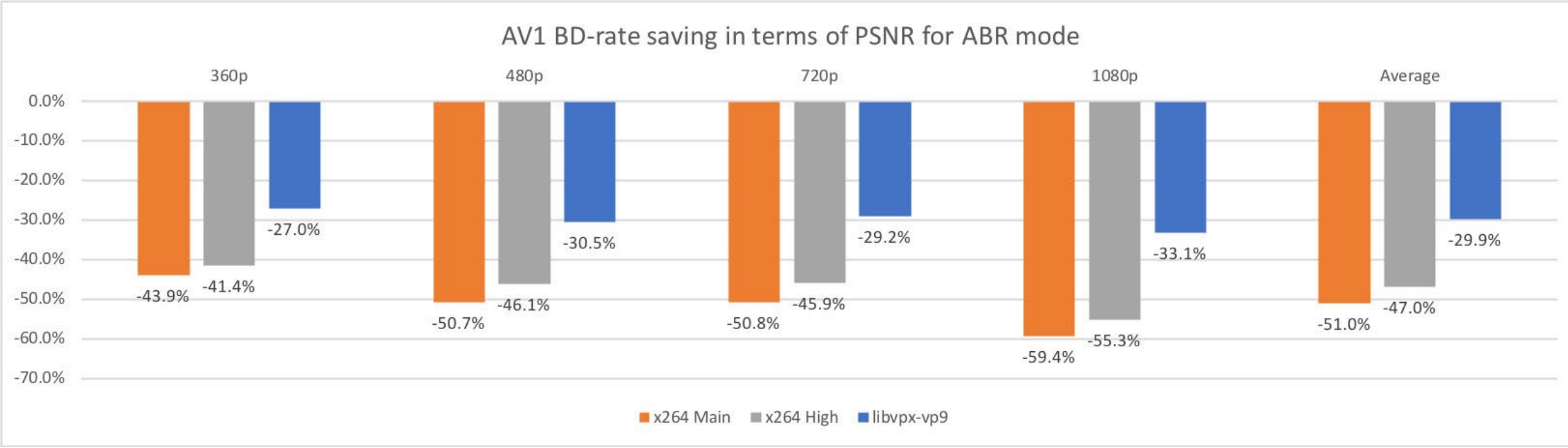
- Strong browser support
  - Chrome, Firefox, Opera
  - HEVC at 16.71; AV1 almost double
- Facebook reportedly supporting via apps in Android and iOS

Adoption Timeline

- Browser soon
- Optimizations in 2018 - 2019
- GPU-accelerated versions in 2019
- Total hardware decode in 2020



AV1: What's it Give You?



- Same quality as VP9 at 70% the bandwidth

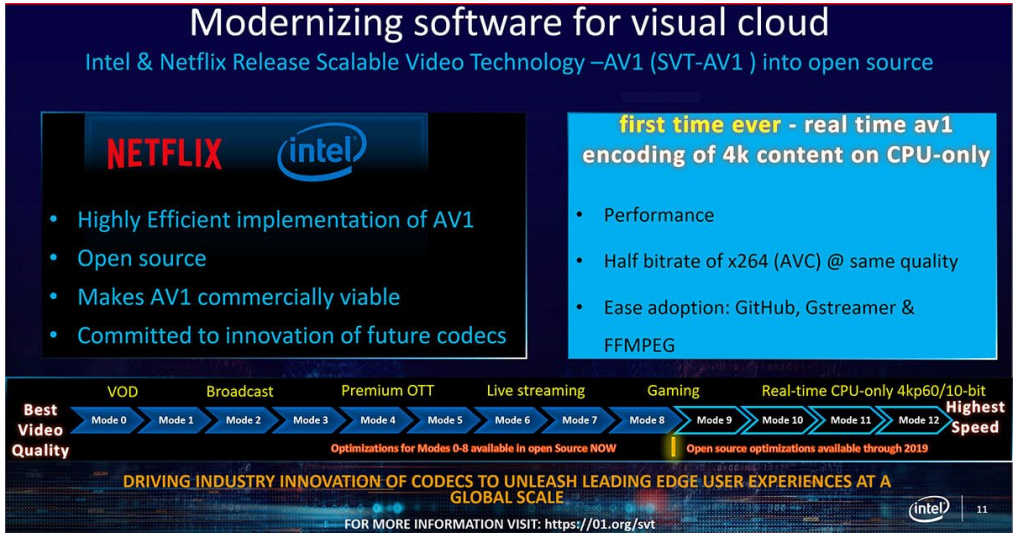
http://bit.ly/2ILAIo7

What's About Encoding Time?

- My tests – 3/4/2019
  - About 16x HEVC
- NAB
  - Intel/Netflix – Real Time 4K/60p
  - Quality tbd

	Encoding Time (Seconds)	Times Real Time	VMAF
AV1 - cpu-used 5	736	147.20	95.55
x265 - slow	38	7.60	94.83
LibVPx - speed 2	35	7.00	93.07
x264 - slow	7	1.40	92.27

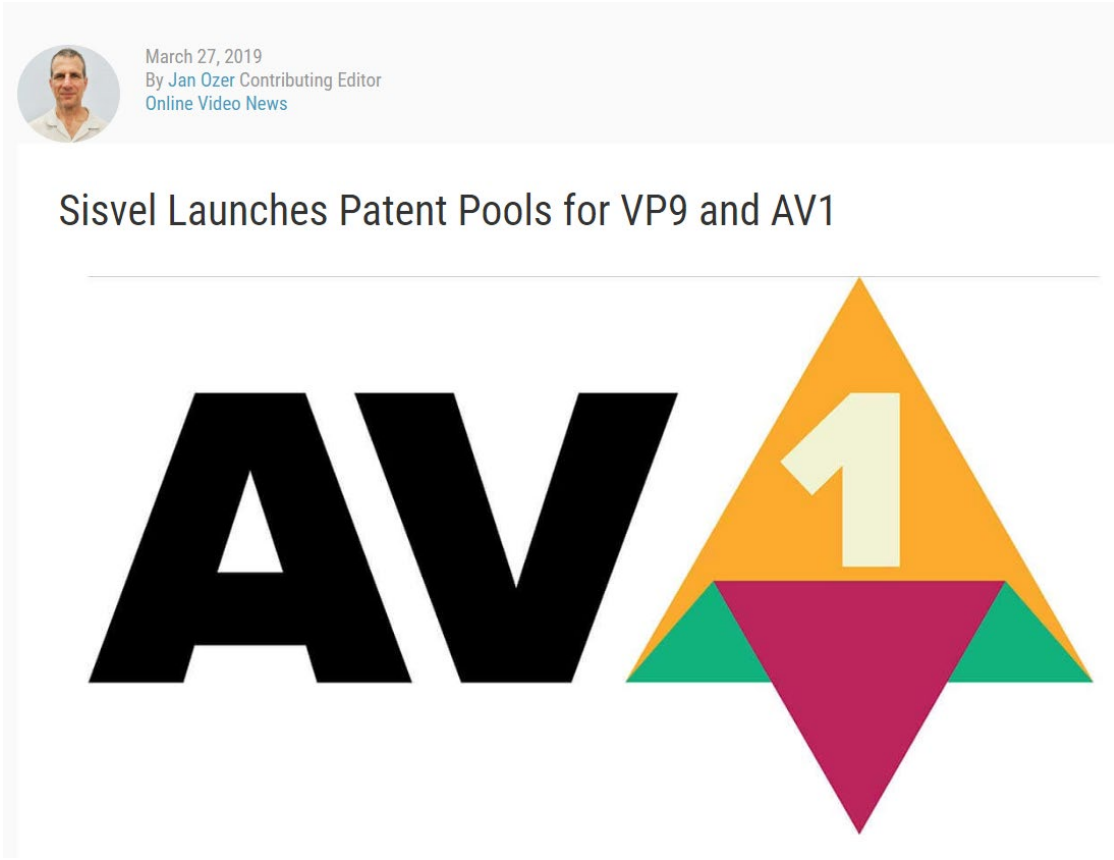
bit.ly/av1_enc_16x



http://bit.ly/av1_re4K60

AV1: What's it Cost You?

- Royalty free, but no indemnifications from AOMedia
  - Sisvel patent pool for AV1/VP9 and threats from Velos
    - Consumer device only (.24 Euro for VP9/.32 Euro for AV1)
    - No content royalties
    - No cap
    - Software tbd



bit.ly/sisvel_av1pool

AV1 Scorecard:

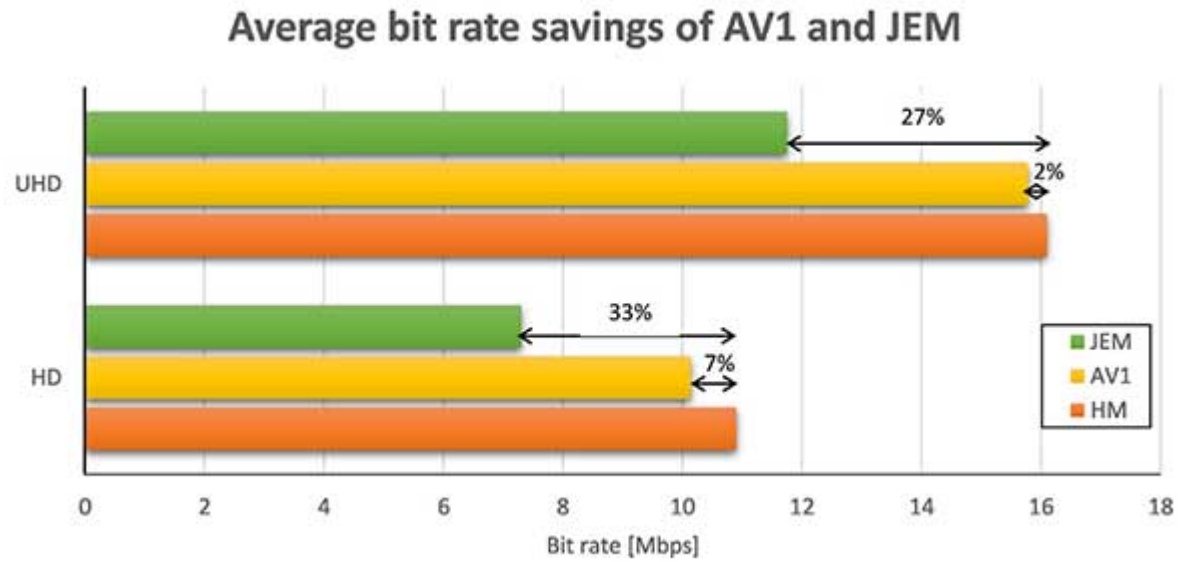
-- If you're not one of the companies on the front panel of most smart TVs
- AV1 won't be relevant through the end of 2018 or later

	H.264	HEVC	VP9	AV1
Revenue Side				
Reach				
Computers	100%	Safari/Edge	95%	80% in 6 months/CPU not yet known
Mobile with hardware	100%	100%	50%	2 years
OTT/Smart TV	100%	100%	75%	2 years
Features				
Live	100%	Many options	Little support	Challenging
Live transcode	100%	Many options	Wowza & Nimble	Challenging
Low latency	100%	Some options	unknown	Challenging
HDR	Not optimal (reach of 10-bit AVC unknown)	100%	No std. support	2 years
Cost Side				
Quality	1 - lowest of the bunch	H.264 @ 60%	H.264 @ 60%	HEVC @ 70%
Encoding time	1	~ 4x H.264	~2X H.264	1000+ x VP9
Content royalty cost	PPV/Subscripion	Velos?	None	None
FUD Factor	Nokia/Motorola	Others not in pool	Feels low risk	Some risk

Bottom Line on Codec Deployments

- Producers deploy new codecs when they open new markets
  - HEVC with 4K/Smart TVs
- Other than the largest producers (Netflix, YouTube, Amazon, Facebook), few producers seem to deploy new codecs to save bandwidth or improve QoE
- Some hope that AV1 will be the long-term H.264 replacement, but there are doubts regarding:
  - Quality at scale
  - Royalty status
  - Encoding cost

VVC in a Nutshell from BBC Report



- HM = HEVC
- AV1 = AV1
- JEM = VVC (don't ask
- Chart shows data rate needed for equivalent quality
  - Shorter is better

- VVC appears to have a significant advantage over AV1 and HEVC
  - But it's two years from being final, about 1.5 years behind AV1, maybe more
- HEVC and AV1 appear about equal
- Will have royalties

Questions?

- Questions
-

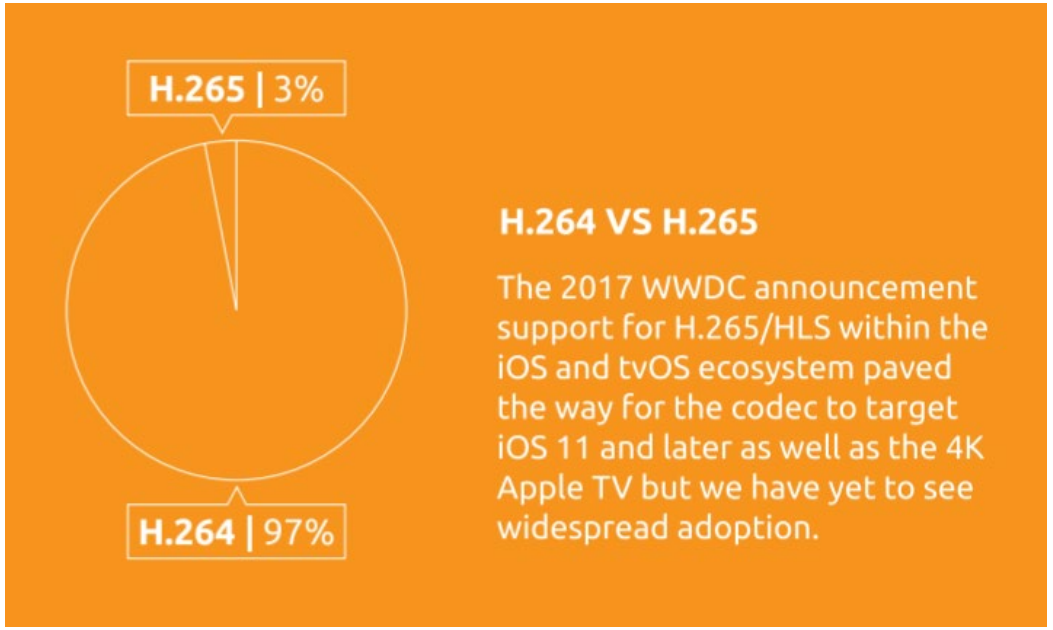
Should be 2:40

Lesson 3 – Delivering HEVC over HLS

- Controlling and sample documents
- Producing HLS streams
  - H264 only
  - H264/HEVC
  - H264/HEVC/HDR

Perspective

- Not a lot of companies doing this
- Very little public information about how to do it



Apple Resources

- HLS Authoring Spec provides
  - Sample encoding ladders
  - Details regarding all aspects of HLS production

http://bit.ly/hls_spec_2017

- HTTP Live Streaming Examples
  - Provides sample streams and manifest files

http://bit.ly/hls_samps

Still labeled
“preliminary example”

- Will reference both during presentation

H.264 Only

Video Streams

- H.264 streams

Trick Play Streams

- i-Frame streams (I-frame playlists (EXT-X-I-FRAME-STREAM-INF) **MUST** be provided to support scrubbing and scanning UI
- SHOULD create one fps “dense” dedicated I-frame renditions
- MAY use I-frames from normal content, but trick play performance is improved with a higher density of I-frames
- Perspective – few producers seem to support trick play requirements

Configuration (h.264)

- Profile and Level MUST be less than or equal to High Profile, Level 4.2.
- SHOULD use High Profile in preference to Main or Baseline Profile

H264 Encoding Ladder - Content

Data Rate	Rez	Frame rate	Profile	Level *	Key Frame	Segment
145	416 x 234	≤ 30 fps	High	4.2	2 second	6 seconds
365	480 x 270	≤ 30 fps	High	4.2	2 second	6 seconds
730	640 x 360	≤ 30 fps	High	4.2	2 second	6 seconds
1100	768 x 432	≤ 30 fps	High	4.2	2 second	6 seconds
2000	960 x 540	source	High	4.2	2 second	6 seconds
3000	1280 x 720	source	High	4.2	2 second	6 seconds
4500	1280 x 720	source	High	4.2	2 second	6 seconds
6000	1920 x 1080	source	High	4.2	2 second	6 seconds
7800	1920 x 1080	source	High	4.2	2 second	6 seconds

* Level: Should not use a higher level than required for content resolution and frame rate

H264 Encoding Ladder – I-Frame/Trick Play

Data Rate	Rez	Frame rate	Profile	Key Frame	Profile	Segment
45	640x360	1 fps	High	1	High	1
90	768x432	1 fps	High	1	High	1
250	960x540	1 fps	High	1	High	1
375	1280x720	1 fps	High	1	High	1
600	1920x1080	1 fps	High	1	High	1

HEVC/H.264

Video Streams

- H.265
- H.264 streams (For backward compatibility some video content **SHOULD** be encoded with H.264)

Trick Play Streams

- H.264
- H.265 (not specified, but Apple has for both)
- Dedicated encodes of iFrame-only streams are preferred, but can use existing file and trick play manifest

Configuration (HEVC)

- Main 10, Level 5, High Tier
  - Level 5 peaks at 30 fps
  - Apple HLS sample stream @ 60 fps (but peak at 1080p)
  - Encoding ladder says 30 fps
- Must be fragmented MP4

HEVC Encoding Ladder - Content

Data Rate	Rez	Frame rate	Profile	Level *	Key Frame	Segment
145	640 x 360	≤ 30 fps	Main 10	5.0	2 second	6 second
300	768 x 432	≤ 30 fps	Main 10	5.0	2 second	6 second
600	960 x 540	≤ 30 fps	Main 10	5.0	2 second	6 second
900	960 x 540	≤ 30 fps	Main 10	5.0	2 second	6 second
1600	1280 x 720	source	Main 10	5.0	2 second	6 second
2400	1280 x 720	source	Main 10	5.0	2 second	6 second
3400	1280 x 720	source	Main 10	5.0	2 second	6 second
4500	1920 x 1080	source	Main 10	5.0	2 second	6 second
5800	1920 x 1080	source	Main 10	5.0	2 second	6 second
8100	2560x1440	source	Main 10	5.0	2 second	6 second
11600	3840x2160	Source/30	Main 10	5.0	2 second	6 second
16800	3840x2160	Source/30	Main 10	5.0	2 second	6 second

* Level: Should not use a higher level than required for content resolution and frame rate

HEVC Encoding Ladder – I-Frame/Trick Play

Data Rate	Rez	Frame rate	Profile	Key Frame	Profile	Segment
40	768x432	1 fps	High	1	High	1
75	960x540	1 fps	High	1	High	1
200	960 x 540	1 fps	High	1	High	1
300	1280 x 720	1 fps	High	1	High	1
525	1920 x 1080	1 fps	High	1	High	1

Note: 6.1 – I-frame playlists MUST be provided to support scrubbing and scanning UI.
No requirement for HEVC

HDR/HEVC/H264

Video Streams

- HDR
- H.265 (SDR streams must be provided – not specified if H.264 content suffices)
- H.264 streams (For backward compatibility some video content **SHOULD** be encoded with H.264)

Trick Play Streams

- H.264
- H.265 (SDR must be provided; not clear if H.264 suffices)
- If HDR provided, should provide at all resolutions

Configuration (HDR)

- MUST be HDR10 or Dolby Vision
  - Dolby Vision – profile 5 (10-bit single layer), level 7
- If HDR provided, SHOULD be provided at all resolutions
- 30 fps or less
- Must be fMP4

HDR Encoding Ladder - Content

Data Rate	Rez	Frame rate	Profile	Level *	Key Frame	Segment
160	640 x 360	≤ 30 fps	Main 10	5.0	2 second	6 second
360	768 x 432	≤ 30 fps	Main 10	5.0	2 second	6 second
730	960 x 540	≤ 30 fps	Main 10	5.0	2 second	6 second
1090	960 x 540	≤ 30 fps	Main 10	5.0	2 second	6 second
1930	1280 x 720	source	Main 10	5.0	2 second	6 second
2900	1280 x 720	source	Main 10	5.0	2 second	6 second
3850	1280 x 720	source	Main 10	5.0	2 second	6 second
5400	1920 x 1080	source	Main 10	5.0	2 second	6 second
7000	1920 x 1080	source	Main 10	5.0	2 second	6 second
9700	2560x1440	source	Main 10	5.0	2 second	6 second
13900	3840x2160	Source/30	Main 10	5.0	2 second	6 second
20000	3840x2160	Source/30	Main 10	5.0	2 second	6 second

* Level: Should not use a higher level than required for content resolution and frame rate

HDR Encoding Ladder – I-Frame/Trick Play

Data Rate	Rez	Frame rate	Profile	Key Frame	Profile	Segment
55	768x432	1 fps	High	1	High	1
94	960x540	1 fps	High	1	High	1
238	960 x 540	1 fps	High	1	High	1
360	1280 x 720	1 fps	High	1	High	1
650	1920 x 1080	1 fps	High	1	High	1

Note: 6.1 – I-frame playlists MUST be provided to support scrubbing and scanning UI.
No requirement for HEVC

All Frame Rate/Bitrate Control

- Frame rates above 60 fps **SHALL NOT** be used.

VOD:

- If progressive use that rate
- You **SHOULD** de-interlace 30i content to 60p instead of 30p (streams above 2 Mbps)

Live:

- Live/linear video from NSTC or ATSC source SHOULD be 60 or 59.94 fps (PAL=50 fps)
- HEVC/HDR – max 30 fps

VOD:

- Average segment bit rate MUST be within 10% of the AVERAGE-BANDWIDTH attribute
- Measured peak bit rate MUST be within 10% of the BANDWIDTH attribute.
- Peak bit rate SHOULD be no more than 200% of the average bit rate.

Live:

- Average segment bit rate over a long (~1 hour) MUST be less than 110% of the AVERAGE-BANDWIDTH attribute
- Measured peak bit rate MUST be less than 125% of the BANDWIDTH attribute.

Apple's HEVC/H264 Encoding Ladder

- Nine HEVC video variants
  - Gear 9 - 1920x1080 @ 5.8 Mbps
  - Gear 8 - 1920x1080 @ 4.5 Mbps
  - Gear 7 - 1920x1080 @ 3.2 Mbps
  - Gear 6 - 1280x720 @ 2.4 Mbps
  - Gear 5 - 960x540 @ 1.7 Mbps
  - Gear 4 - 768x432 @ 990 Mbps
  - Gear 3 - 640x360 @ 660 kbps
  - Gear 2 - 480x270 @ 350 kbps
  - Gear 1 - 416x234 @ 145 kbps
- Nine H.264 video variants
  - Gear 9 - 1920x1080 @ 7.8 Mbps
  - Gear 8 - 1920x1080 @ 6.0 Mbps
  - Gear 7 - 1920x1080 @ 4.5 Mbps
  - Gear 6 - 1280x720 @ 3.0 Mbps
  - Gear 5 - 960x540 @ 2.0 Mbps
  - Gear 4 - 768x432 @ 1.1 Mbps
  - Gear 3 - 640x360 @ 730 kbps
  - Gear 2 - 480x270 @ 365 kbps
  - Gear 1 - 416x234 @ 145 kbps
- I-Frame variants (fast-forward / rewind support)
- 3 audio renditions
  - AAC-LC - 48 kHz stereo @ 160 kbps
  - AC-3 - 48 kHz 5.1 @ 384 kbps
  - EC-3 - 48 kHz 5.1 @ 192 kbps
- 1 subtitle rendition (WebVTT)
  - English

- I-frame variants in HEVC/H264 formats
- Dolby obviously not required

https://developer.apple.com/streaming/examples/

H.264 Adaptive Group (from Master)

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=2190673,BANDWIDTH=2523597,CODECS="avc1.640020,mp4a.40.2",
RESOLUTION=960x540,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v5/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=8052613,BANDWIDTH=9873268,CODECS="avc1.64002a,mp4a.40.2",
RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v9/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=6133114,BANDWIDTH=7318337,CODECS="avc1.64002a,mp4a.40.2",
RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v8/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=4681537,BANDWIDTH=5421720,CODECS="avc1.64002a,mp4a.40.2",
RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v7/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=3183969,BANDWIDTH=3611257,CODECS="avc1.640020,mp4a.40.2",
RESOLUTION=1280x720,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v6/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=1277747,BANDWIDTH=1475903,CODECS="avc1.64001f,mp4a.40.2",
RESOLUTION=768x432,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v4/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=890848,BANDWIDTH=1017705,CODECS="avc1.64001f,mp4a.40.2",
RESOLUTION=640x360,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v3/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=533420,BANDWIDTH=582820,CODECS="avc1.64001f,mp4a.40.2",
RESOLUTION=480x270,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v2/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=303898,BANDWIDTH=339404,CODECS="avc1.64001f,mp4a.40.2",
RESOLUTION=416x234,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v1/prog_index.m3u8

H.264 I-Frame Group

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=928091,BANDWIDTH=1015727,CODECS="avc1.640028",
RESOLUTION=1920x1080,URI="tp5/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=731514,BANDWIDTH=760174,CODECS="avc1.64001f",
RESOLUTION=1280x720,URI="tp4/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=509153,BANDWIDTH=520162,CODECS="avc1.64001f",
RESOLUTION=960x540,URI="tp3/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=176942,BANDWIDTH=186651,CODECS="avc1.64001f",
RESOLUTION=640x360,URI="tp2/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=90796,BANDWIDTH=95410,CODECS="avc1.64001f",
RESOLUTION=480x270,URI="tp1/iframe_index.m3u8"

H.265 Adaptive Group (from Master)

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=1966314,BANDWIDTH=2164328,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=960x540,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v14/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=6105163,BANDWIDTH=6664228,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v18/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=4801073,BANDWIDTH=5427899,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v17/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=3441312,BANDWIDTH=4079770,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v16/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=2635933,BANDWIDTH=2764701,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=1280x720,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v15/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=1138612,BANDWIDTH=1226255,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=768x432,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v13/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=829339,BANDWIDTH=901770,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=640x360,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v12/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=522229,BANDWIDTH=548927,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",
RESOLUTION=480x270,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v11/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=314941,BANDWIDTH=340713,CODECS="hvc1.2.4.L123.B0,mp4a.40.2”
,RESOLUTION=416x234,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v10/prog_index.m3u8

HEVC I-Frame Group

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=287207,BANDWIDTH=328352,CODECS="hvc1.2.4.L123.B0",
RESOLUTION=1920x1080,URI="tp10/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=216605,BANDWIDTH=226274,CODECS="hvc1.2.4.L123.B0",
RESOLUTION=1280x720,URI="tp9/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=154000,BANDWIDTH=159037,CODECS="hvc1.2.4.L123.B0",
RESOLUTION=960x540,URI="tp8/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=90882,BANDWIDTH=92800,CODECS="hvc1.2.4.L123.B0",
RESOLUTION=640x360,URI="tp7/iframe_index.m3u8”

#EXT-X-I-FRAME-STREAM-INF:AVERAGE-BANDWIDTH=50569,BANDWIDTH=51760,CODECS="hvc1.2.4.L123.B0",
RESOLUTION=480x270,URI="tp6/iframe_index.m3u8"

IV: Playback Performance and Ladder Creation

- Hybrid and low hybrid
- Tests and results
- Conclusions

Created Two Encoding Ladders for Testing

- “Hybrid”
  - Contained all rungs of recommended H.264 and HEVC ladders

16:9 aspect ratio	H.264/AVC
416 x 234	145
640 x 360	365
768 x 432	730
768 x 432	1100
960 x 540	2000
1280 x 720	3000
1280 x 720	4500
1920 x 1080	6000
1920 x 1080	7800

16:9 aspect ratio	HEVC/H.265 30 fps
640 x 360	145
768 x 432	300
960 x 540	600
960 x 540	900
960 x 540	1600
1280 x 720	2400
1280 x 720	3400
1920 x 1080	4500
1920 x 1080	5800
2560 x 1440	8100
3840 x 2160	11600
3840 x 2160	16800

Created Two Encoding Ladders for Testing

- "Hybrid"
  - Contained all rungs of recommended H.264 and HEVC ladders
- "Low-Hybrid"
  - Sub 720p rungs in H.264
  - 720p and higher rungs in HEVC

16:9 aspect ratio	H.264/AVC
416 x 234	145
640 x 360	365
768 x 432	730
768 x 432	1100
960 x 540	2000
1280 x 720	3000
1280 x 720	4500
1920 x 1080	6000
1920 x 1080	7800

16:9 aspect ratio	HEVC/H.265 30 fps
640 x 360	145
768 x 432	300
960 x 540	600
960 x 540	900
960 x 540	1600
1280 x 720	2400
1280 x 720	3400
1920 x 1080	4500
1920 x 1080	5800
2560 x 1440	8100
3840 x 2160	11600
3840 x 2160	16800

Burned File Configuration into Files

- Used FFmpeg text filter to burn rez/codec/data rate info into file

Asked for Volunteer Testers on LinkedIn

Please Help Me Test HEVC Playback in HLS

Published on April 30, 2018 Edit article | View stats

Jan Ozer
Consultant and Author
26 articles

893 24 15 0

Results

- 43 desktop
- 19 mobile

What Did We Learn

- Generally good performance and compatibility
  - H.264 streams played on older devices without problem
  - Very few quality issues
  - No disruption when switching between H.264 and HEVC

What Did We Learn

- Playback
  - Apple typically won't retrieve higher resolution file than display resolution
    - One instance where MacBookPro with 1800 vertical rez retrieved 4K file
    - Otherwise, followed this rule
  - 4K doesn't get retrieved all that often
    - Average bandwidth when retrieving 4K was 580 Mbps
    - Lowest was 64 Mbps for 16.8 Mbps stream
    - Many devices with very high bandwidth and necessary resolution could not play
    - Apple looking into this as potential "bug"

Does Ladder Composition Make a Difference?

- Maybe
- There were several instances where the result between hybrid and low hybrid differed
  - In all but one instance, the low-hybrid experience was worse
    - Either H.264 instead of HEVC
    - Lower data rate/resolution
- Safest approach appears to be two complete ladders
  - Obviously, also the most expensive

What Know About Switching?

- Ask Apple – two streams in ladder; which does player select?

1080p – HEVC – 2 Mbps
720p – H.264 – 2.5 Mbps

- Their switching logic is in transition but it “knows” that H.265 should be higher quality than H.264 at similar data rates
  - So don’t need to game the system (create artificially high data rate for H.265 streams so
- Typically won’t switch between H.264 and H.265 when both available
- Apple recommends full H.264/H.265 ladders in all cases

Questions?

- Questions
-

Should be 3:00

Lesson 4: Per-Title Encoding

- What is per-title Encoding
- Evolution of per-title encoding
- Shot-based encoding
- Per-context encoding

What is Per-Title Encoding

- Customizing encoding for each file
- First implemented by Netflix and YouTube
- First encoder implementation – Capella Systems Cambria Encoder
- Can be implemented vis capped CRF

Why is Optimizing the Bitrate Critical?

Consumer Side

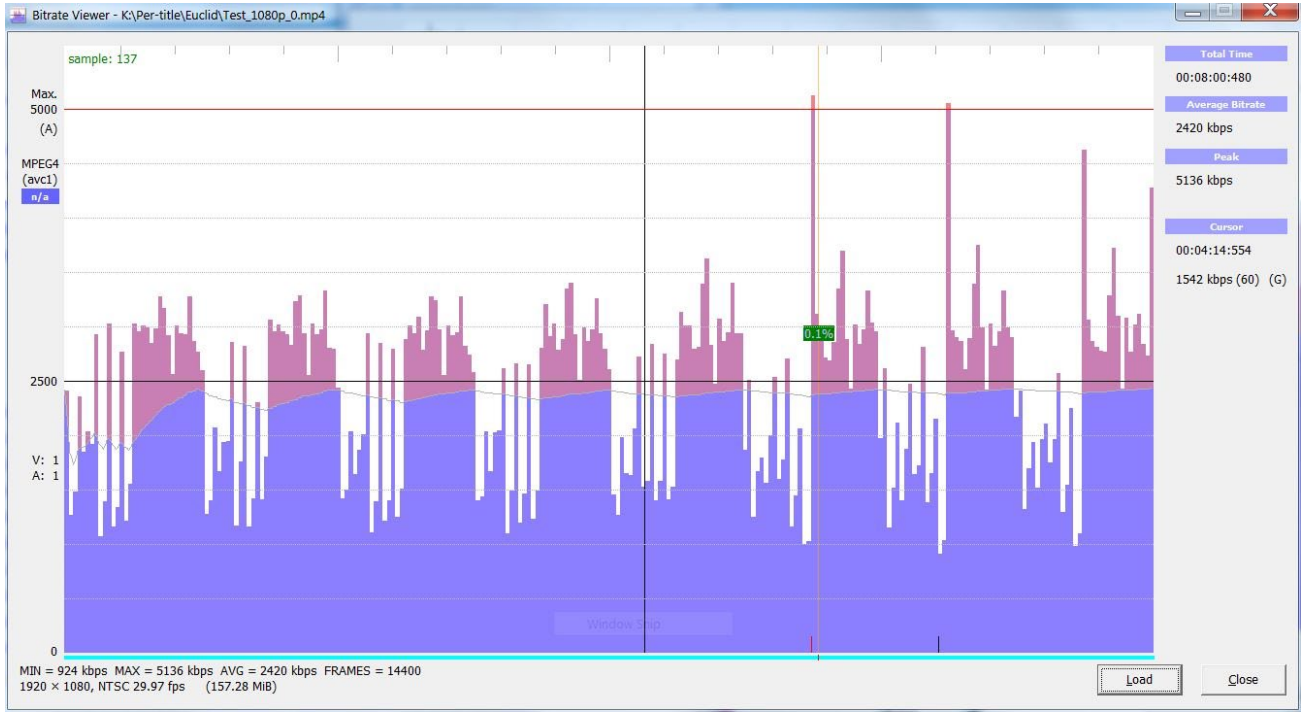
- Reduced bandwidth cost (consumer/corporate)
  - Home
  - Mobile
- More efficient on networks
- Better quality of experience
  - Higher rez stream to mobile

Producer Side

- Lower bandwidth costs
- Lower storage costs
- Lower encoding costs
- More video through fixed pipes
- Better reach to consumers on edge of networks
- More competitive because of consumer-side benefits
- More competitive because a feature in competitive products and services

Evolution of Per-Title/Optimization

When	Prior to 2015
What	Optimization
Who	Beamr/Euclid/ CRF
Operation	Frame by frame
Overall bitrate control	No
Change GOP/Segment	No

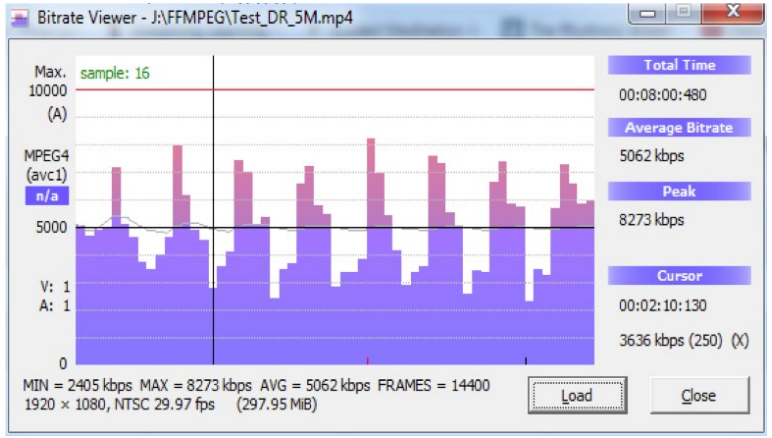


Video – 30 seconds talking head/
30 seconds ballet – repeat 8x

No bitrate control except cap

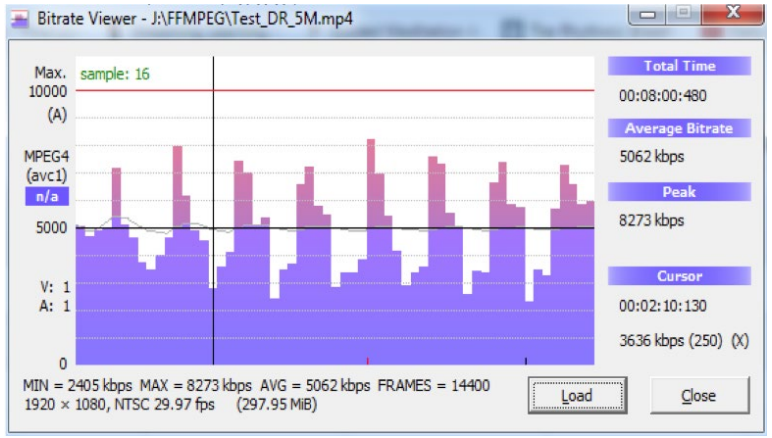
Evolution of Per-Title/Optimization

When	Prior to 2015	Late 2015
What	Optimization	Per-Title Encoding
Who	Beamr/Euclid	Netflix
Operation	Frame by frame	Gauge complexity/ Choose bitrate ladder
Overall bitrate control	No	Yes; CBR/VBR
Change GOP/Segment	No	No



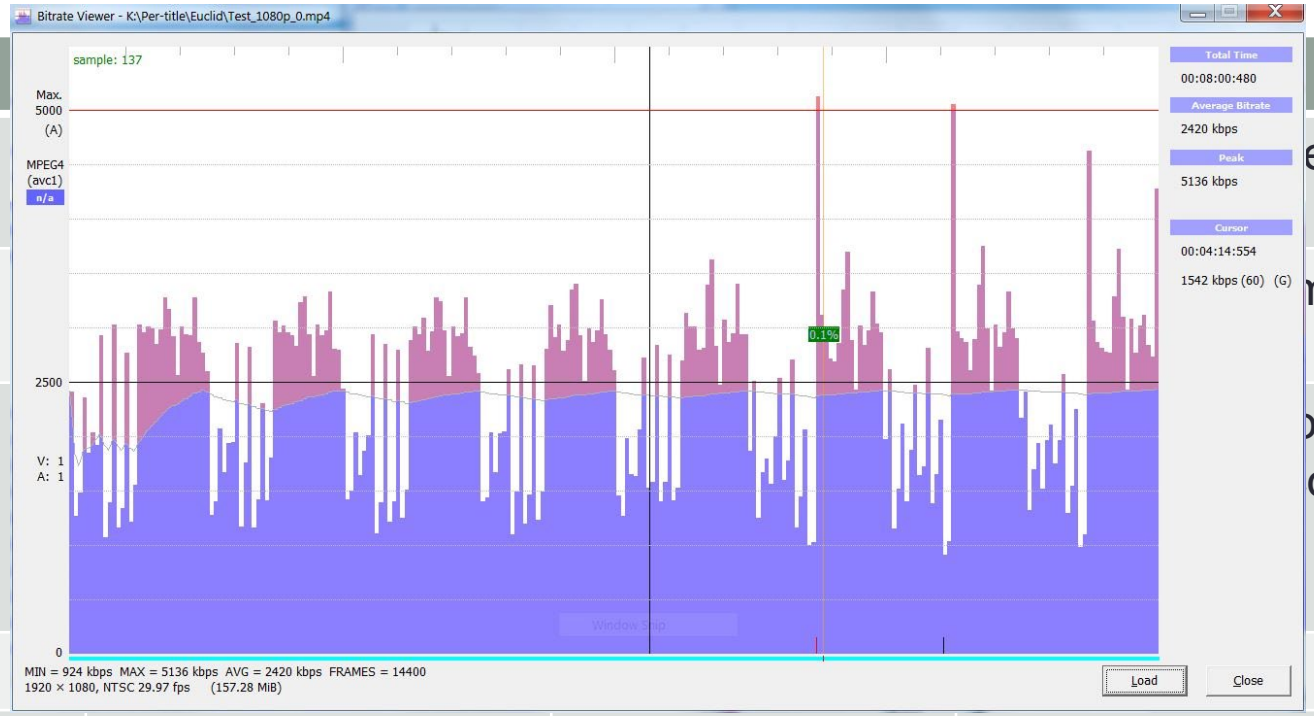
Evolution of Per-Title/Optimization

When	Prior to 2015	Late 2015	2016-2017
What	Optimization	Per-Title Encoding	Commercial Per-title
Who	Beamr/Euclid/ Capped CRF	Netflix	Capella Systems
Operation	Frame by frame	Gauge video complexity/ encode traditionally	Gauge video complexity/ encode traditionally
Overall bitrate control	No	Yes	CBR/VBR
Change GOP/Segment	No	No	No



Evolution of Per-Title/Optimization

When				Late 2017
What			er-	Segment-based encoding
Who			ms,	Euclid, others
Operation			ode	Gauge complexity for each segment; encode segment
Overall bitrate control				Cap, but no CBR
Change GOP/Segment	No	No	No	No

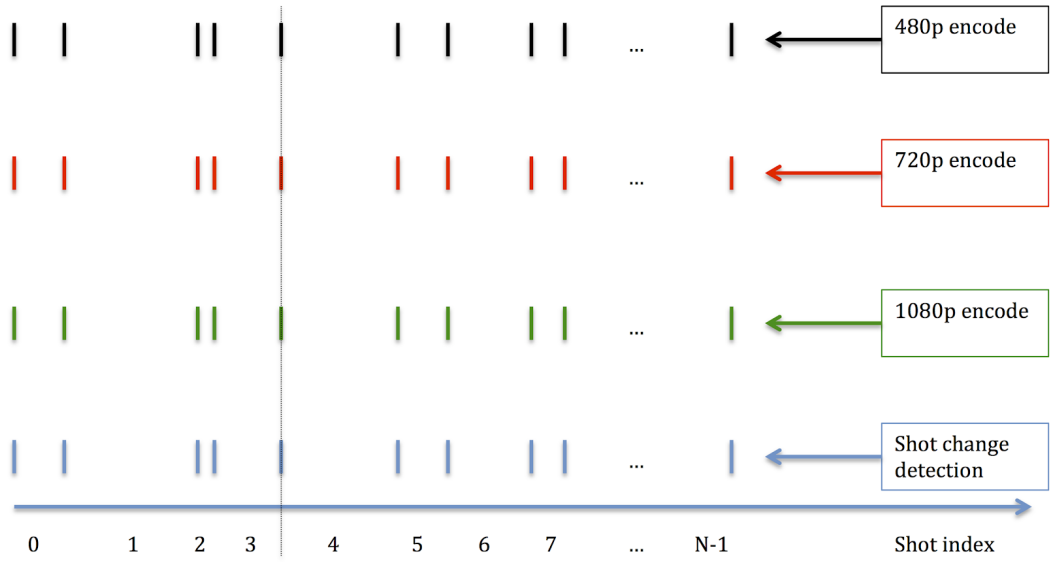


Evolution of Per-Title/Optimization

When	Prior to 2015	Late 2015	2016-2017	Late 2017	2018
What	Optimization	Per-Title Encoding	Commercial Per-title	Segment-based encoding	Shot-based encoding
Who	Beamr/Euclid/ Capped CRF	Netflix	Capella Systems, BC, others	Euclid, others	Netflix
Operation	Frame by frame	Gauge video complexity/ encode traditionally	Gauge video complexity/ encode traditionally	Gauge complexity for each segment; encode segment	Divide each video into shots; encode separately
Overall bitrate control	No	Yes	CBR/VBR	Cap, but no CBR	Probably cap only
Change GOP/Segment	No	No	No	No	Yes

Why Shot-Based Encoding Make Sense

- Key frames at scene changes and not at regular intervals
  - Switching preserved because all iterations encoded the same way
- Major encoding changes up and down at scene changes (so not noticeable)
- Rate control not critical because most scenes are relatively homogenous (minimal capping which can degrade quality)
- Seeking via I-frames are all at scene changes



http://bit.ly/nf_shot

Why Shot-Based Encoding Make Sense

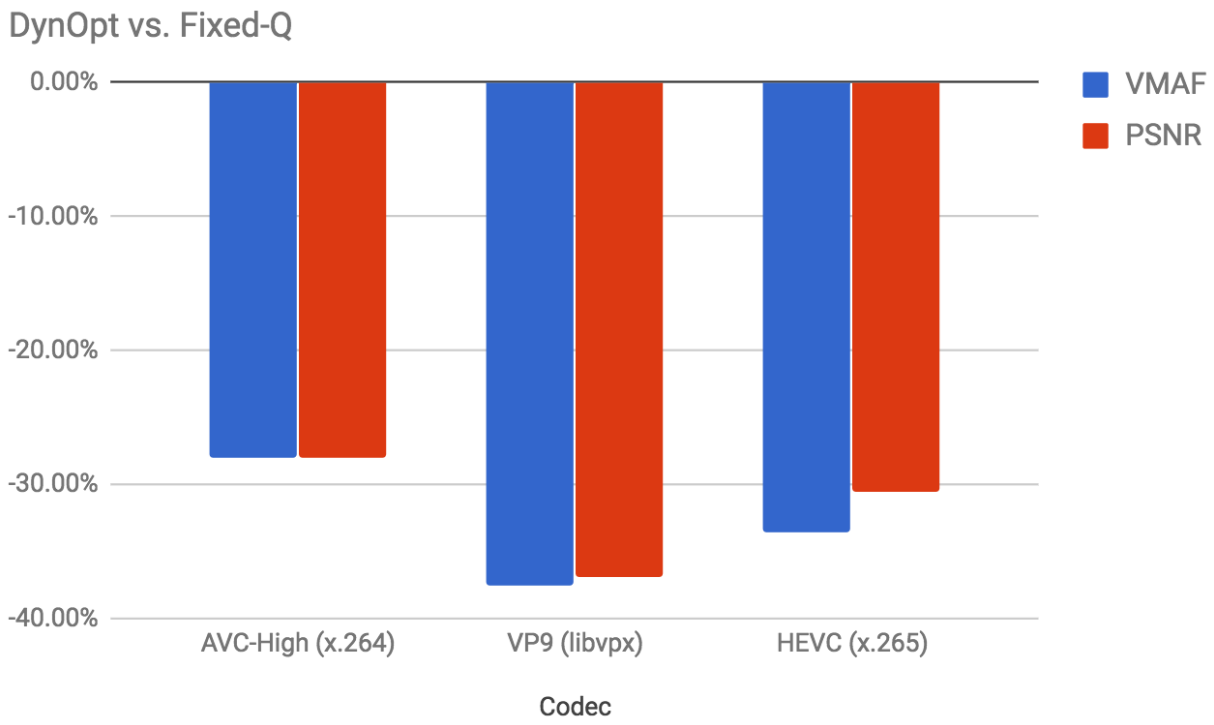
- Significant data rate reductions

TITLE	Best fixed-QP encoding		Dynamic Optimizer @ **same quality**	Dynamic Optimizer @ **same bitrate**
	Bitrate (kbps)	HVMAF (0-100)	Bitrate savings (%)	Delta HVMAF (0-100)
Bloodline	245	86.6	-15%	+2.0
BoJack	230	95.5	-14%	+1.1
Breaking Bad	251	91.8	-16%	+1.7
Marvel's Daredevil	247	92.0	-21%	+1.9
El Fuente	262	36.1	-38%	+21.8
House of Cards	213	92.3	-17%	+1.3
Meridian	259	96.0	-13%	+0.6
Orange is the new black	256	86.2	-11%	+1.6
The Avengers	278	82.0	-18%	+3.4
Wet Hot American Summer	231	78.7	-8%	+1.6
AVERAGE	**247**	**83.7**	**-17.1%**	**+3.7**

http://bit.ly/nf_shot

Why Shot-Based Encoding Make Sense

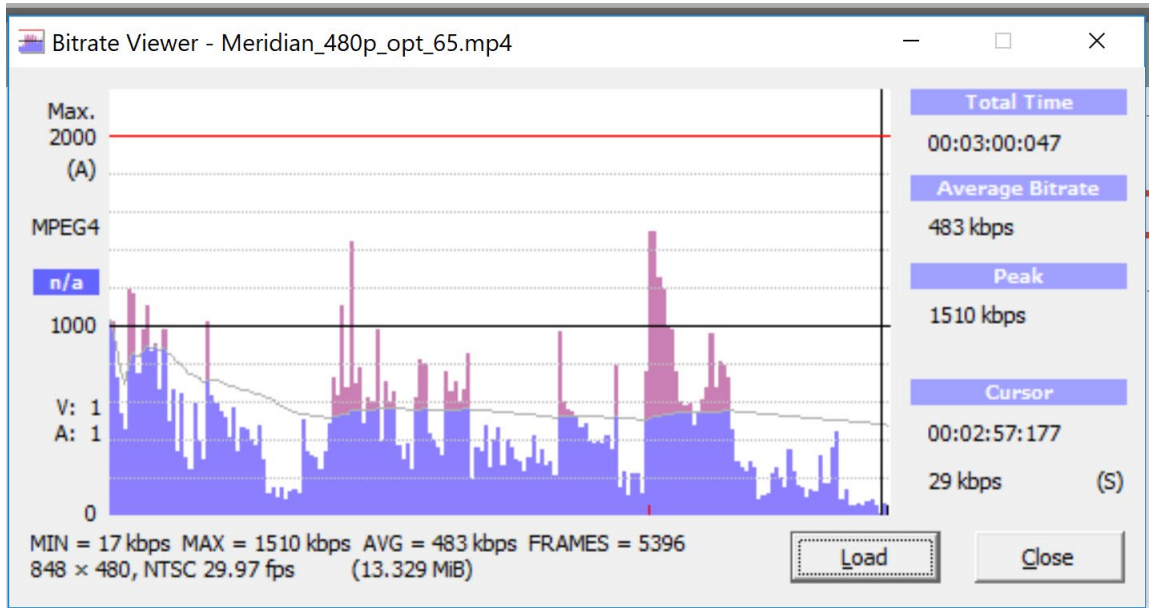
- Benefits are very significant
- Not codec-dependent



http://bit.ly/nf_shot

Issues:

- Traditional rate control may not be available
  - Assume capping
  - But, if this bitrate pattern gives you nightmares, per-shot encoding is probably not for you
- You can’t have it
  - Closest I looked at was segment-based optimization (from Euclid)
  - Assume it’s coming from some third party vendors, but it is technically complex



Evolution of Per-Title/Optimization

When	Prior to 2015	Late 2015	2016-2017	Late 2017	2018	2018-2019
What	Optimization	Per-Title Encoding	Commercial Per-title	Segment-based encoding	Shot-based encoding	Context-aware encoding
Who	Beamr/Euclid / Capped CRF	Netflix	Capella Systems, BC, others	Euclid, others	Netflix	Brightcove, Epic Labs, Mux
Operation	Frame by frame	Gauge video complexity/ encode traditionally	Gauge video complexity/ encode traditionally	Gauge complexity for each segment; encode segment	Divide each video into shots; encode separately	Incorporate bandwidth and device data into encoding ladder
Overall bitrate control	No	Yes	CBR/VBR	Cap, but no CBR	Probably cap only	Can
Change GOP/Segment	No	No	No	No	Yes	Can

How it Works

Usage Pattern

Device type	Usage [%]	Average bandw
PC	0.004	7.5654
Mobile	94.321	3.2916
Tablet	5.514	3.8922
TV	0.161	5.4374
All devices	**100**	**3.3283**

TABLE 2: USAGE AND AVERAGE BANDWIDTH STATISTICS FO

Mostly mobile with low bandwidth

Device type	Usage [%]	Average bandwidt
PC	63.49	14.720
Mobile	6.186	10.609
Tablet	9.165	12.055
TV	21.15	24.986
All devices	100	16.393

TABLE 3: USAGE AND AVERAGE BANDWIDTH STATISTICS FOR OPE

Mostly PC/TV at high bitrates

Device type	Usage [%]	Average bandwidth [Mb
PC	0.0	N/A
Mobile	0.0	N/A
Tablet	0.0	N/A
TV	100	35.7736
All devices	100	35.7736

TABLE 4: USAGE AND AVERAGE BANDWIDTH STATISTICS FOR OPERATOR 3.

All TV at very high bandwidth

Encoding Ladder

Concentrate on lower rungs

Ren	Profile	Resolution	Framerate	Bitrate	SSIM
1	seline	320x180	30	125	0.93369
2	Baseli	480x270	30	223.08	0.93793
3	Main	640x360	30	398.11	0.94636
4	Main	960x540	30	774.78	0.94953
5	Main	1280x720	30	1549.5	0.95637
6	High	1600x900	30	2765.3	0.96105
7	High	1920x1080	30	4935.1	0.96576
Storage				10771	

TABLE 9: CAE-GENERATED ENCODING LADDER FOR OPERATOR 1.

Higher in the middle

Rendition	Profile	Resolution	Framerate	Bitrate	SSIM
1	Baseline	320x180	30	125	0.93338
2	Baseline	480x270	30	239.71	0.94122
3	Main	640x360	30	469.54	0.95202
4	Main	1024x576	30	939.08	0.95221
5	Main	1280x720	30	1568.8	0.95658
6	High	1600x900	30	2765.3	0.96105
7	High	1920x1080	30	4935.1	0.96576
Storage				11026	

TABLE 10: CAE-GENERATED ENCODING LADDER FOR OPERATOR 2.

Higher end, fewer lower rungs

Rendition	Profile	Resolution	Framerate	Bitrate	SSIM
1	Baseline	320x180	30	125	0.93447
2	Baseline	512x288	30	307.42	0.94855
3	Main	960x540	30	803.59	0.95050
4	Main	1280x720	30	1727.8	0.95864
5	High	1920x1080	30	5050.7	0.96599
Storage				8014.6	

TABLE 11: CAE-GENERATED ENCODING LADDER FOR OPERATOR 3.

bit.ly/bc_contextaware

Much More at Tuesday's Session

T103. A Survey Of Per-Title Encoding Technologies

Tuesday, May 7: 1:45 p.m. - 2:30 p.m.

Per-title encoding techniques customize the encoding ladder to match the encoding complexity of the source, saving bandwidth on easy-to-compress videos and ensuring the quality of more complex footage. Codec specialist Jan Ozer compares the efficiency, implementation issues, and costs of multiple commercially available and open-source alternatives for live and VOD per-title encoding. Learn what per-title encoding is, how the various options work, and which is the best option for you.

Speaker:
Jan Ozer, Principal, Streaming Learning Center and Contributing Editor, *Streaming Media*

Contestants

- Technologies
  - Capped CRF (DIY)
  - Capella Systems Cambria
    - Source Adaptive Bitrate Ladder
  - Elemental Technologies QVBR
  - Epic Labs Lightflow
  - QEncode

Key Observations

Questions?

- Questions
-

Should be 3:30

Lesson 5: Build Encoding Ladder with Objective Quality Metrics

- Objective quality metrics:
  - What they are and why we need them
  - Meet VMAF
- Building your ladder with metrics
  - Simple case
  - HEVC and advanced codecs

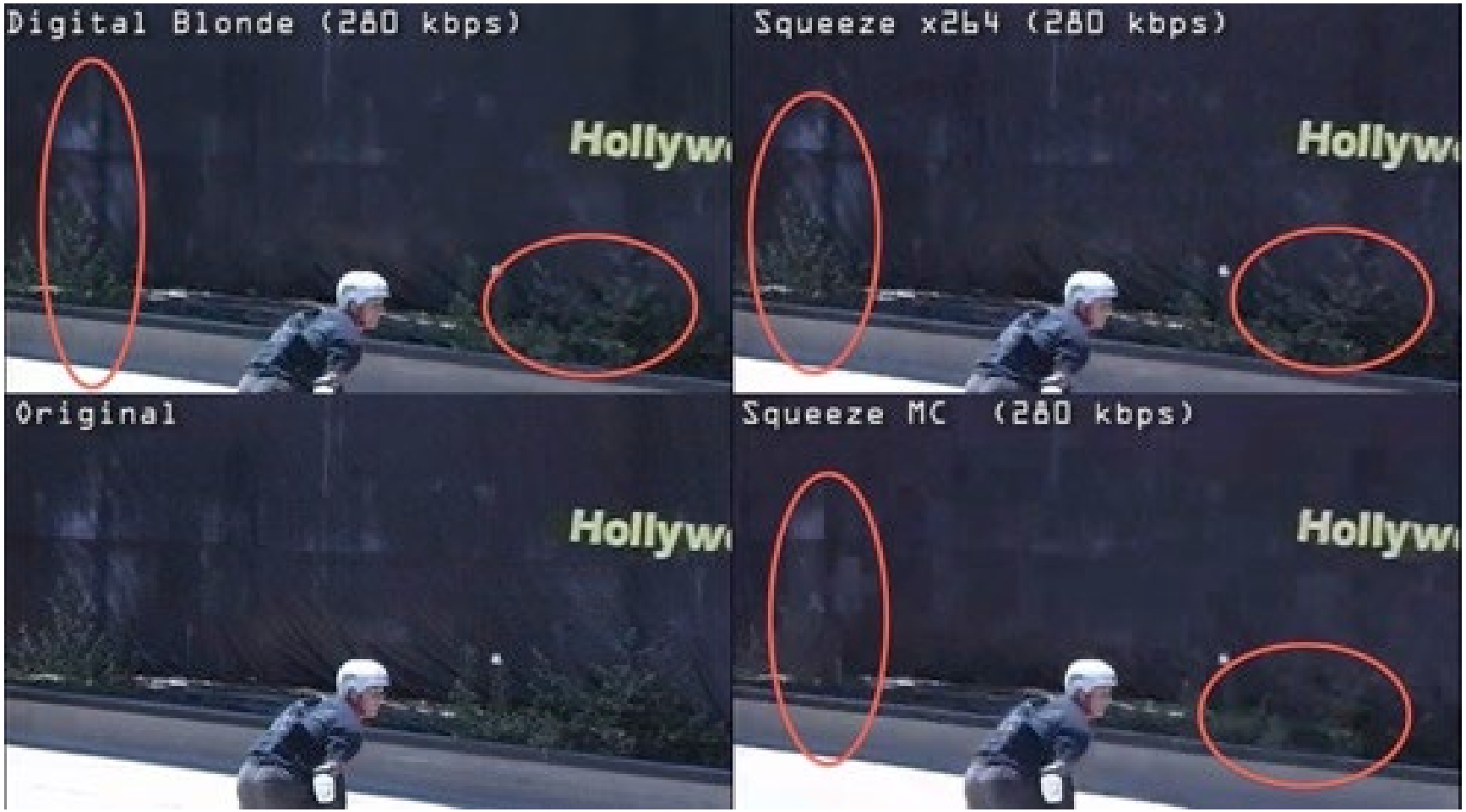
What Are Objective Quality Metrics

- Mathematical formulas that (attempt to) predict how human eyes would rate the videos
  - Faster and less expensive
  - Automatable
- Examples
  - Peak Signal to Noise Ratio (PSNR)
  - Structural Similarity Index (SSIM)
  - SSIMPlus
  - VMAF (Video Multimethod Assessment Fusion)

Why Do We Need Them?

- So many encoding decisions
  - Data rate
  - Keyframe interval
  - B-frame interval
  - Bitrate control technique (VBR vs. CBR)
  - Choice of codec
  - Profile
  - Preset
- All have tradeoffs (quality vs. encoding time
- Objective quality metrics allow us to mathematically measure quality
- Uses
  - Drive many per-title encoding technologies (Netflix)
  - Useful for many critical encoding decisions

Took Me From Here

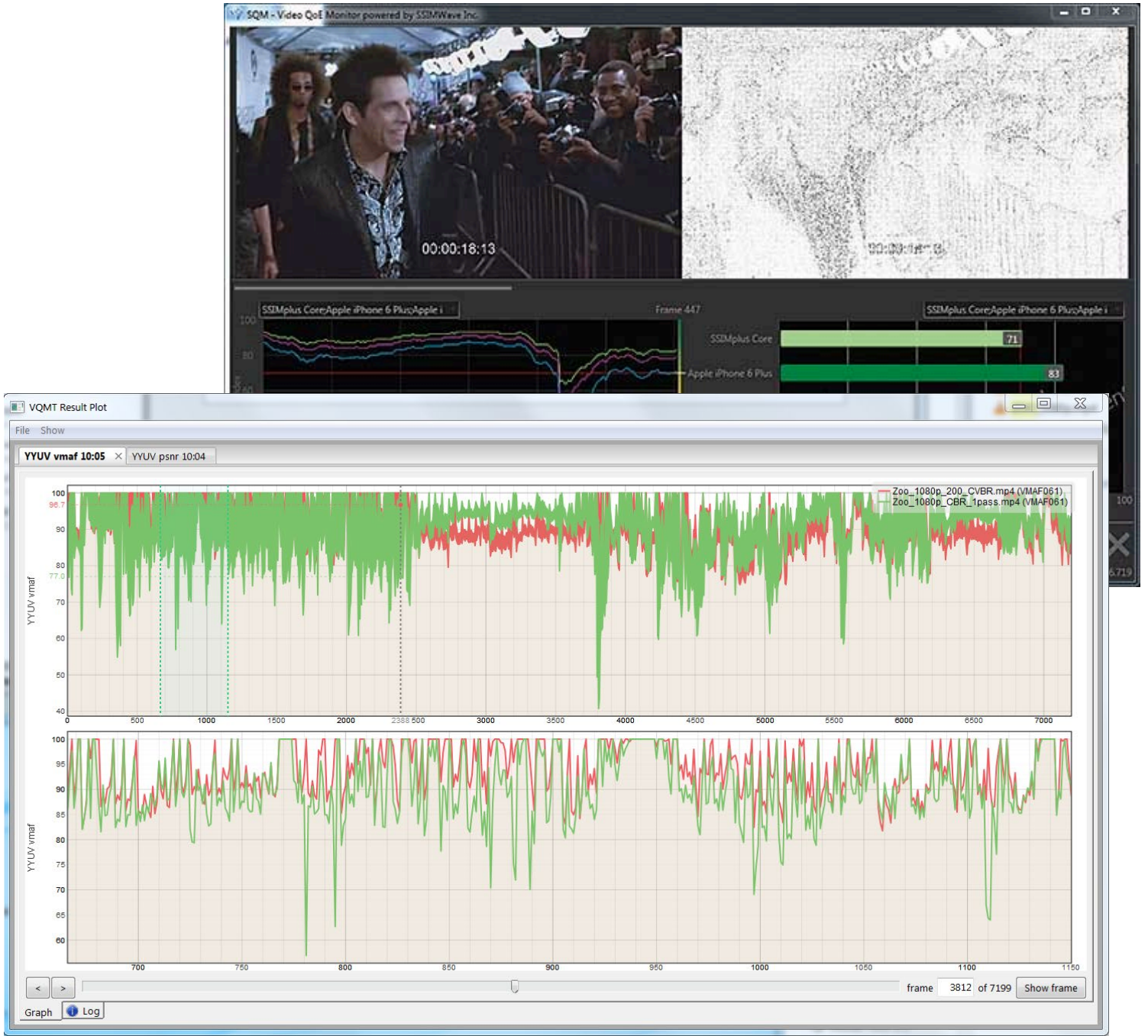


Time consuming and error prone
Subjective comparisons

To Here

VQM (lower is better)					
	Codec A	Codec B	Codec C	High > Low	Codec A > Codec B
Office 1	0.36	0.36	0.37	-3.54%	0.61%
Office 2	0.69	0.61	0.70	-13.51%	12.32%
Office 3	0.28	0.28	0.32	-14.74%	1.32%
Office 4	0.87	0.79	0.87	-9.63%	9.63%
Parking 1	0.68	0.61	0.74	-21.23%	10.90%
Parking 2	0.57	0.55	0.64	-15.47%	3.04%
Parking 3	1.86	1.58	1.76	-17.88%	17.88%
Parking 4	0.47	0.49	0.51	-8.86%	-3.81%
Retail 1	0.56	0.54	0.56	-4.27%	4.27%
Retail 2	0.68	0.66	0.69	-4.45%	3.39%
Retail 3	0.78	0.72	0.76	-8.64%	8.64%
Retail 4	0.73	0.67	0.88	-32.16%	8.52%
Traffic 1	0.55	0.50	0.58	-15.89%	9.14%
Traffic 2	0.34	0.32	0.38	-17.79%	6.39%
Traffic 3	0.52	0.49	0.55	-11.42%	5.29%
Traffic 4	0.68	0.61	0.66	-11.56%	11.56%
Total	**10.61**	**9.78**	**10.96**		
7.84%	Difference between Codec A and Codec B				
-3.34%	Difference between Codec A and Codec C				
-12.13%	Difference between Codec B and Codec C				
	0.61				
	Green equals best in category				
	Orange means worst in category				
	Difference greater than 7.5%				

Statistically meaningful comparisons



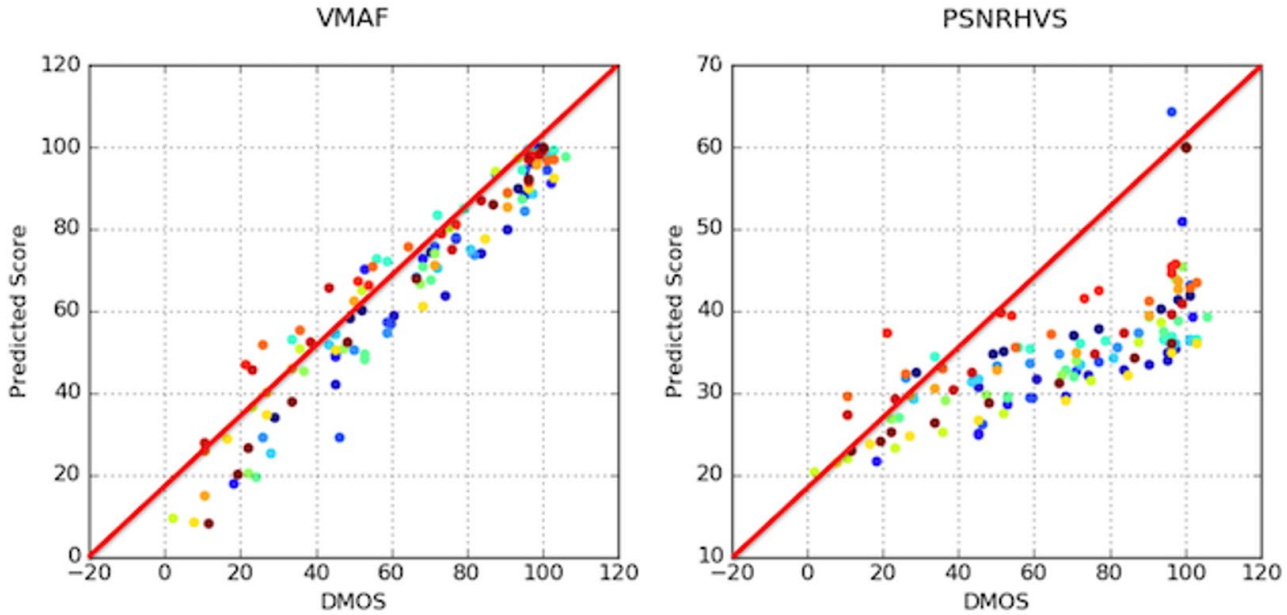
With Objective Quality Metrics You Get

- More data
  - Can run many more tests in much less time
- Better data
  - Mathematical models can detect smaller changes than your eye can easily discern

What is VMAF?

- VMAF is “trainable”
  - Compute VMAF
  - Measure human subjective ratings
  - Feed those results back into VMAF to make the algorithm “smarter”
- Uses
  - Train for different types of content (animation, sports)
  - Train for different viewing conditions

VMAF is a Good Predictor of Subjective Ratings



- Horizonal axis is DMOS rating (human scores)
- Vertical is metric (VMAF on left, PSNR on right)
- Red line is perfect score – metric exactly matches subjective evaluation

- VMAF is more tightly clumped around red line, which means it's more accurate
  - Machine learning means it can get more accurate over time
- PSRN is much more scattered, and as a fixed algorithm, will never improve

Computing VMAF

Source

4K Source

Encode

4K output

2K output

1080p output

720p output

480p output

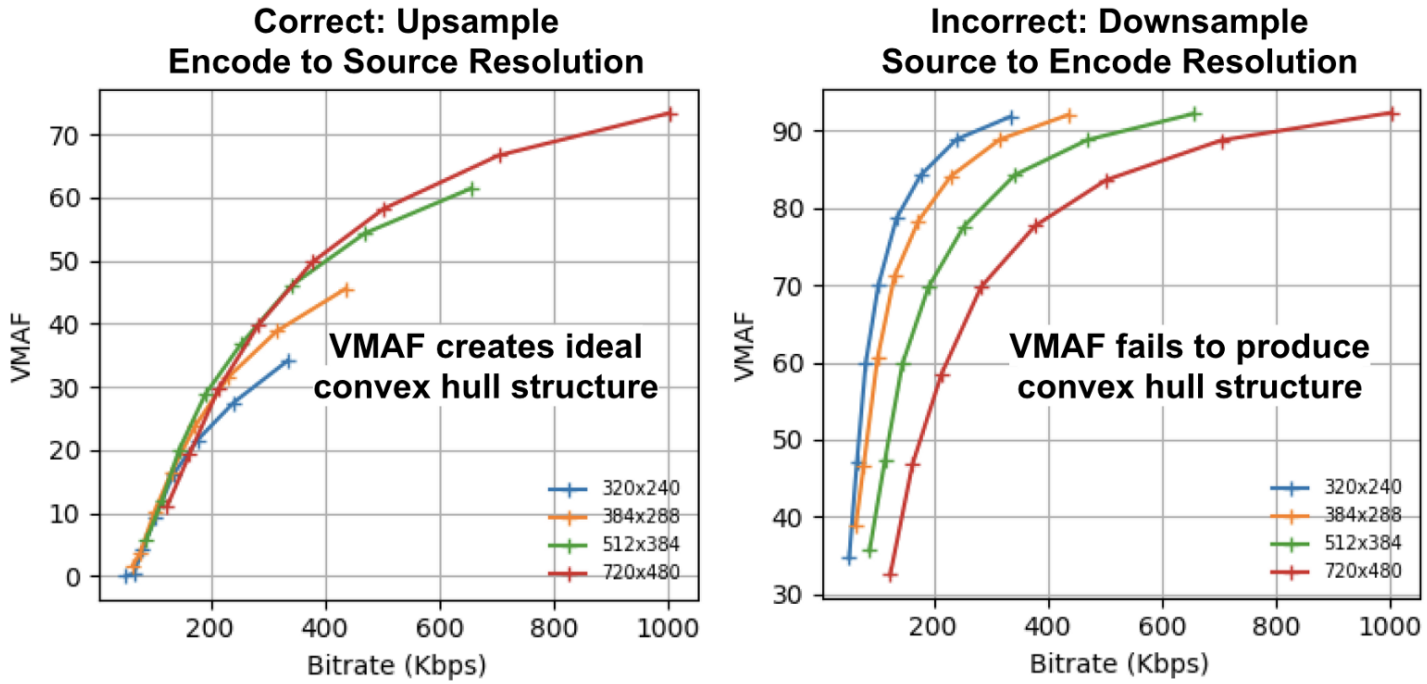
Compare to:

4K Source

http://bit.ly/VMAF_journey

What's This Mean

- Lower resolution rungs necessarily lose detail, yet get compared to 4K
  - Appropriate – assuming viewed on 4K TV
- Ensures that scores will drop at lower resolutions
- Scores range from 0 – 100
  - 80 -100 – excellent
  - 60 – 80 – good
  - 40 – 60 – fair
  - 20 – 40 – poor
  - Below 20 - bad



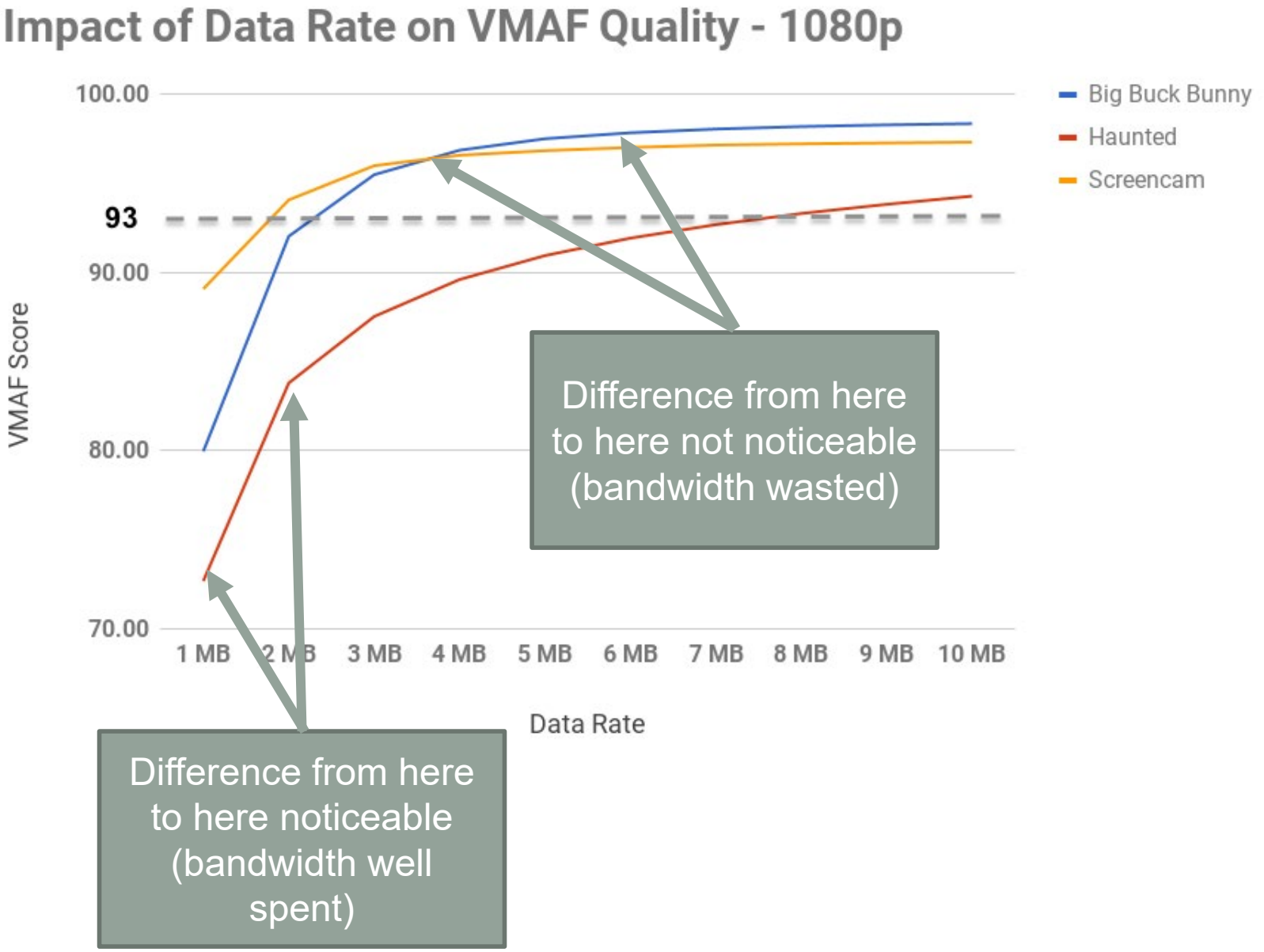
http://bit.ly/VMAF_journey

VMAF Verification – 93 is the Number

- Real Networks White Paper - VMAF Reproducibility: Validating a Perceptual Practical Video Quality Metric
  - 4K 2D videos
- The results indicate that if a video service operator were to encode video to achieve a ***VMAF score of about 93*** then they would be confident of optimally serving the vast majority of their audience with content that ***is either indistinguishable from original or with noticeable but not annoying distortion.***
  - http://bit.ly/vrqm_5

Working With VMAF

- Range – 0 – 100
- Top rung target – typically 93 – 95
  - Higher is a waste
- Scores map to subjective
  - 0-20 bad - 20 – 40 poor
  - 40 – 60 fair - 60 – 80 good
  - 80 – 100 excellent
- 6 VMAF points = Just noticeable difference



VMAF Models

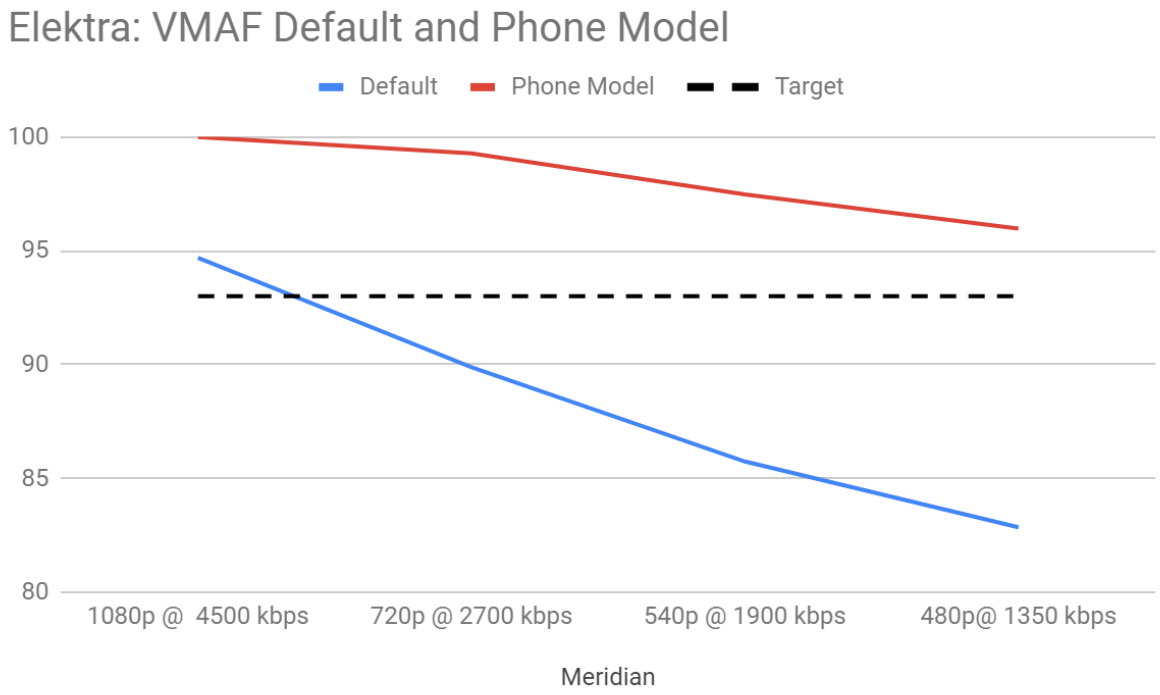
- Original (Default) model
  - Assumed that viewers watch a 1080p display with the viewing distance of 3x the screen height (3H).
- Phone model
  - Assume viewers watch on a mobile phone
- 4K Model
  - Video displayed on a 4K TV and viewed from a distance of 1.5H

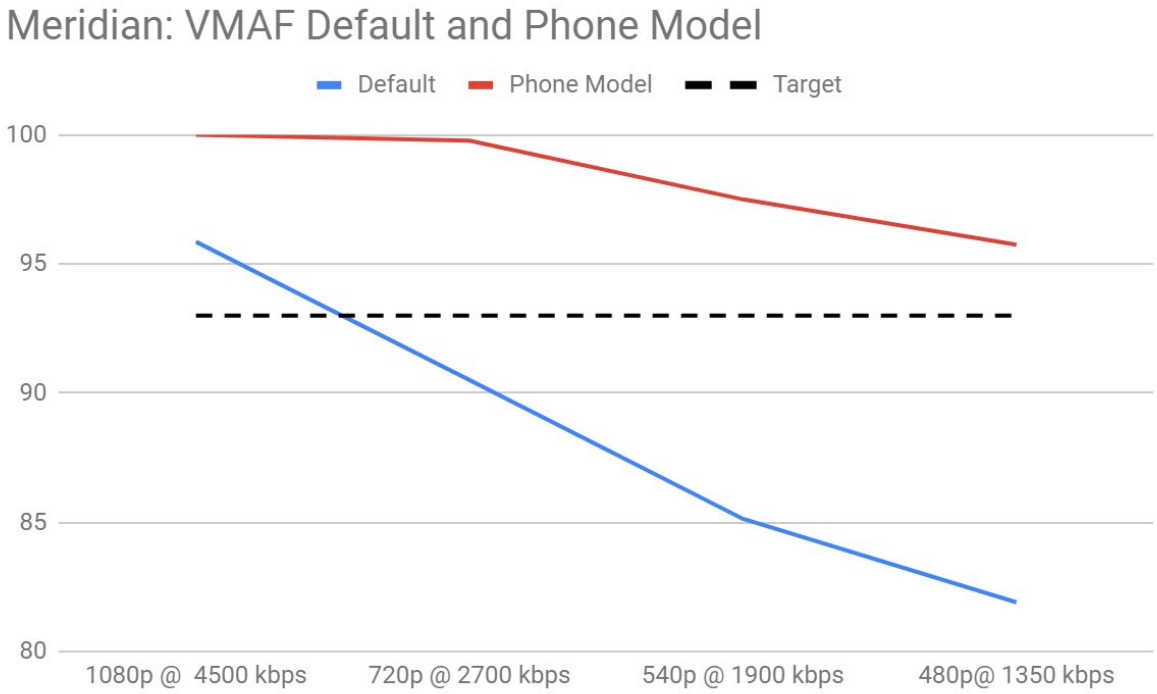
1080p display

Mobile Phone

4K display

Phone vs. Default Model





- 4 encodes, 1080p, 720p, 540p, 480p
- Phone and default VMAF models; 93 target
- With phone model, 480p is above the 93 target in both videos
  - Any reason to transmit 540p+ rungs to mobile phones?

- Only 1080p file is above 93 using default model
  - Need 1080p video in your encoding ladder to achieve 93 score on 1080p displays
- Certainly: Should run both models on 1080p footage targeted at mobile phones and larger displays

Computing VMAF

- Moscow State University VQMT - $995
- Hybrik Cloud – at least $1,000/month
- VMAF Master – Free
- Elecard Video Quality Estimator - $850

Building Your Encoding Ladder with VMAF/CRF

- Simple case
- HEVC and advanced codecs
- Animations and synthetic videos

Using Quality Metrics – Finding the Ceiling

- What is the ceiling?
  - The lowest full resolution data rate that delivers acceptable quality
- Finding the ceiling
  - About CRF
  - VMAF correlation
  - Hollywood proof
- Choosing the resolutions

On Constant Rate Factor Encoding

- What is it:
  - An encoding mode in x264, x265, VP9
  - Adjusts data rate to achieve target quality
  - Quality range is 1-51; lower levels are higher quality

```
FFmpeg –i input.mp4  -b:v 5000k output.mp4
```

Delivers 5 Mbps; quality varies

```
FFmpeg –i input.mp4  -crf 23 output.mp4
```

Delivers crf 23 quality; bitrate varies

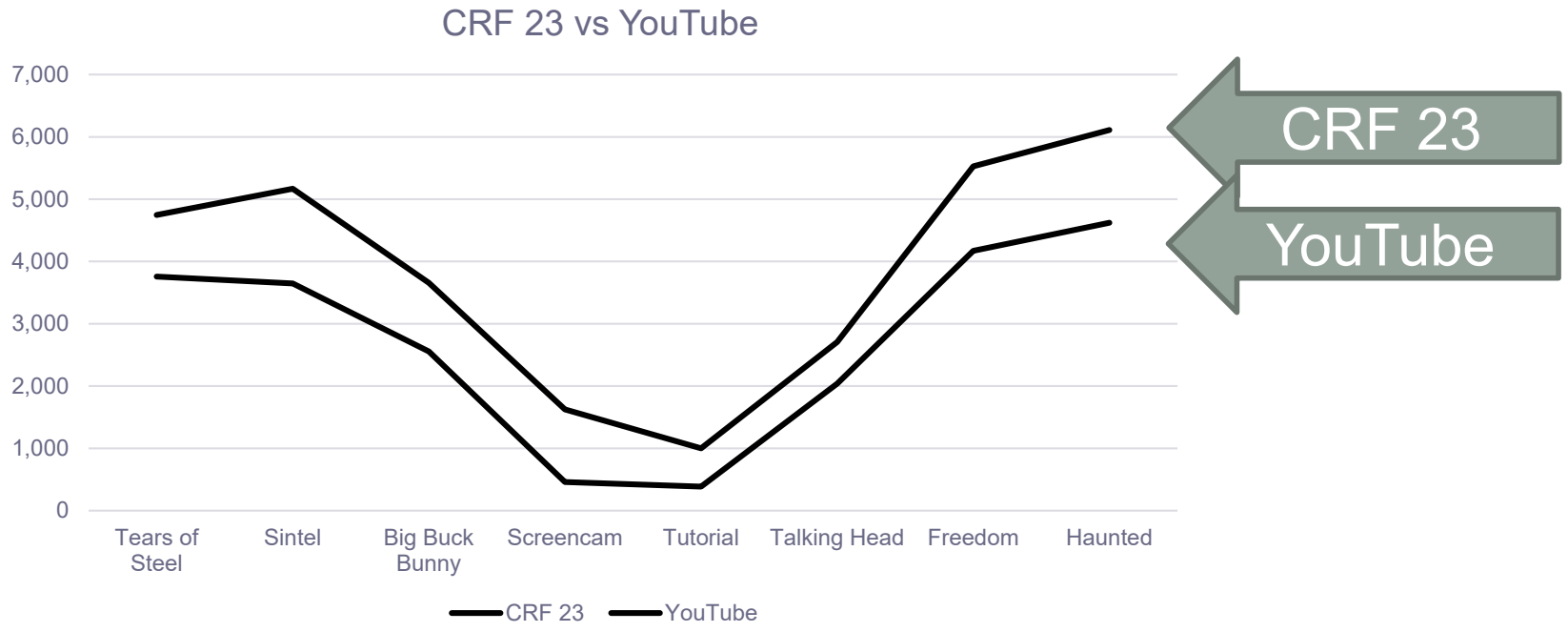
Finding the Optimal Data Rate for 1080p Content (Per-title)

- Compute data rate with CRF 23
  - Encoded 8 files using CRF 23
  - Data rates varied from 1,001 to 6,111 (over 600%)
- Measure VMAF rating
  - Values ranged from 92.74 to 96.88
  - Standard deviation was 1.39 (pretty small)
- Analysis
  - At 2.7 Mbps, a talking head video offers same quality as movie at 6.1 Mbps (even lower for synthetic videos)
  - Validating the benefits of per-title encoding

CRF23 - 1080p	FPS	Description	Data Rate	VMAF
Tears of Steel	24	Real world/CG movie	4,747	96.45
Sintel	24	Complex animation	5,168	96.96
Big Buck Bunny	30	Simple animation	3,657	96.88
Screencam	30	Camtasia-based video	1,625	96.59
Tutorial	30	PowerPoint and talking head	1,001	96.68
Talking Head	30	Simple talking head	2,706	95.47
Freedom	30	Concert footage	5,527	95.90
Haunted	30	DSLR movie-like production	6,111	92.74
Average			**3,818**	**95.96**
Standard deviation				**1.39**

- Conclusion:
  - Trying to find appropriate top data rate for videos, use VMAF 93
  - CRF 23 with x.264 typically delivers VMAF 93

Reality Check: YouTube Comparison



- Upload files to YouTube; measure data rate
- YouTube uses AI-based per-title optimization
- Pattern very similar

- YouTube averages 1 Mbps lower
- 3 VMAF points lower (1/2 JND)
- More validation that CRF 23 and VMAF 93 predict acceptable quality

Choosing the Data Rate for Individual Rungs

- Once you know the highest, the rest is just simple math
  - Step 1: Choose highest – VMAF 93
  - Step 2: Choose lowest – slowest speed you want to serve
  - Step 4: fill in the blanks (between 150/200% apart)
    - Don’t strand at too low a value
    - Too close together causes unnecessary stream switches and increases encoding costs

200 kbps

2x

400 kbps

2x

800 kbps

1.75x

1400 kbps

1. 5x

2100 kbps

1. 5x

3100 kbps

1. 5x

4600 kbps

What Resolution?

- Netflix approach
  - Compute VMAF scores at multiple resolutions at each data rate
  - Choose the best quality resolution (green) at each data rate

H.264	1080p	720p	540p	432p	360p	270p	234p
5000	96.22						
4800	96.01						
4600	95.80	95.27					
4400	95.55	95.10					
4200	95.30	94.96					
4000	94.96	94.73					
3800	94.60	94.53					
3600	94.14	94.30					
3400	93.70	93.99					
3200	93.11	93.64					
3000	92.48	93.24					
2800	91.70	92.78					
2600	90.75	92.25					
2400	89.70	91.59	90.39				
2200	88.37	90.80	89.76				
2000	86.72	89.85	88.95	86.93			
1800	84.68	88.66	88.00	86.10			
1600	82.13	87.13	86.77	85.02	81.58		
1400	78.65	85.19	85.16	83.67	80.28		
1200	73.91	82.56	83.01	81.84	78.57		
1000	67.39	78.86	80.02	79.24	76.19		
900	63.18	76.39	77.98	77.47	74.60	66.66	60.58
800	57.93	73.25	75.51	75.34	72.68	65.11	59.23
700	51.47	69.42	72.34	72.59	70.23	63.14	57.49
600	43.12	64.52	68.37	69.11	67.12	60.70	55.33
500	33.31	58.05	63.13	64.66	63.04	57.52	52.46
400	20.82	49.48	56.00	58.46	57.48	53.13	48.59
300	9.74	37.56	45.95	49.62	49.60	46.80	42.96
200	3.73	20.40	30.87	36.12	37.48	36.88	34.03
100		2.75	8.08	14.45	17.50	19.85	18.66

Encoding Ladders for HEVC/VP9/AV1

Tears of Steel

H.264

H.264	1080p	720p	540p	432p	360p	270p	234p
5000	96.22						
4800	96.01						
4600	95.80	95.27					
4400	95.55	95.10					
4200	95.30	94.96					
4000	94.96	94.73					
3800	94.60	94.53					
3600	94.14	94.30					
3400	93.70	93.99					
3200	93.11	93.64					
3000	92.48	93.24					
2800	91.70	92.78					
2600	90.75	92.25					
2400	89.70	91.59	90.39				
2200	88.37	90.80	89.76				
2000	86.72	89.85	88.95	86.93			
1800	84.68	88.66	88.00	86.10			
1600	82.13	87.13	86.77	85.02	81.58		
1400	78.65	85.19	85.16	83.67	80.28		
1200	73.91	82.56	83.01	81.84	78.57		
1000	67.39	78.86	80.02	79.24	76.19		
900	63.18	76.39	77.98	77.47	74.60	66.66	60.58
800	57.93	73.25	75.51	75.34	72.68	65.11	59.23
700	51.47	69.42	72.34	72.59	70.23	63.14	57.49
600	43.12	64.52	68.37	69.11	67.12	60.70	55.33
500	33.31	58.05	63.13	64.66	63.04	57.52	52.46
400	20.82	49.48	56.00	58.46	57.48	53.13	48.59
300	9.74	37.56	45.95	49.62	49.60	46.80	42.96
200	3.73	20.40	30.87	36.12	37.48	36.88	34.03
100		2.75	8.08	14.45	17.50	19.85	18.66

HEVC

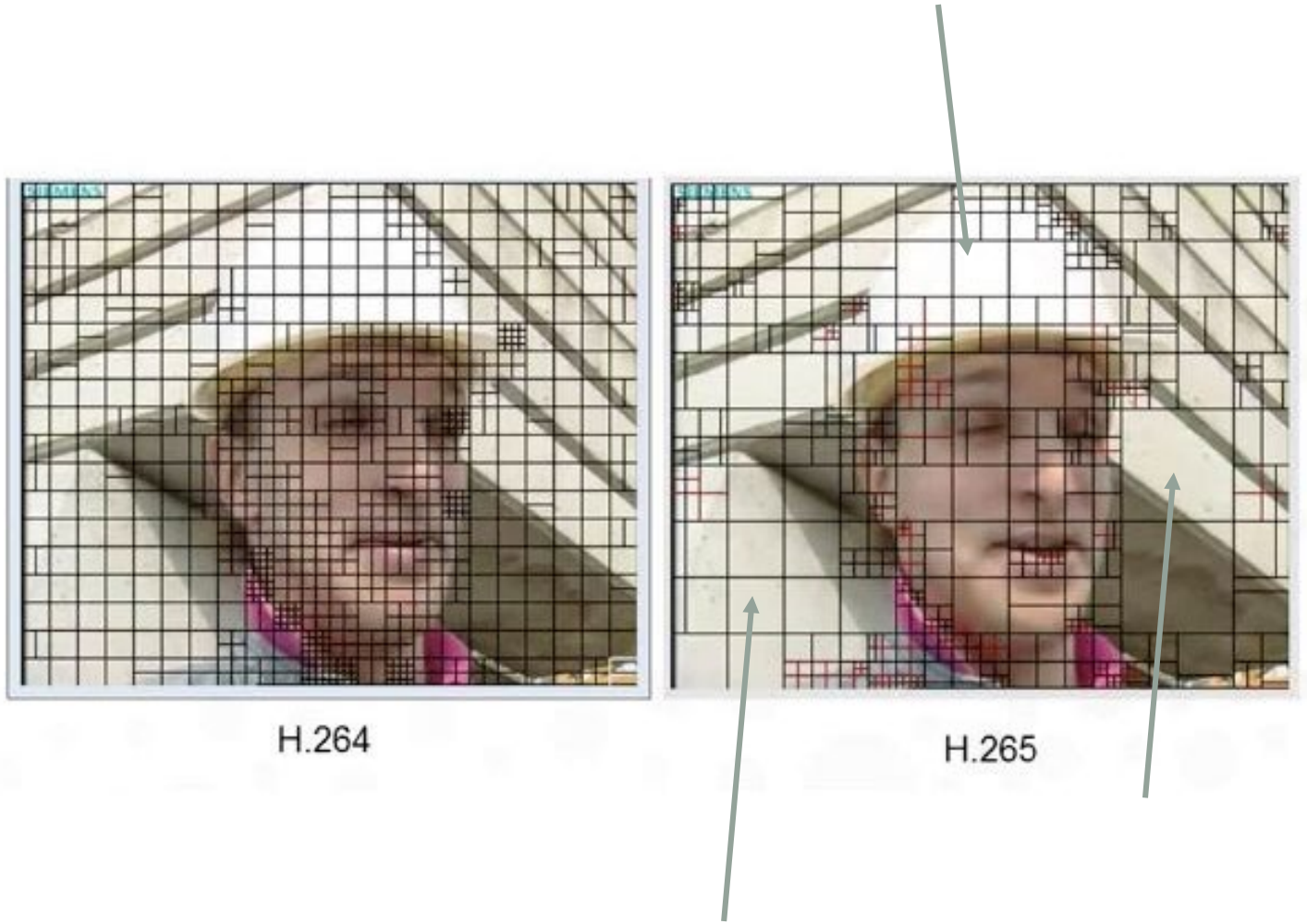
HEVC	1080p	720p	540p	432p	360p	270p	234p
5000	97.67						
4800	97.55						
4600	97.44						
4400	97.31						
4200	97.17						
4000	97.01						
3800	96.84						
3600	96.63						
3400	96.41						
3200	96.15	95.41					
3000	95.86	95.16					
2800	95.52	94.87					
2600	95.09	94.52					
2400	94.58	94.12	92.09				
2200	93.97	93.63	91.62				
2000	93.16	93.02	91.05				
1800	92.18	92.25		87.63			
1600	90.94	91.27	89.44	86.78	83.18		
1400	89.36	89.97	88.27	85.69	82.12		
1200	87.30	88.26	86.68	84.22	80.73		
1000	84.42	85.84	84.46	82.20	78.79		
900	82.39	84.21	83.02	80.86	77.51	68.45	62.18
800	80.03	82.20	81.23	79.19	75.91	67.09	60.92
700	77.04	79.67	78.90	77.07	73.91	65.38	59.35
600	73.10	76.34	75.88	74.29	71.36	63.21	57.34
500	68.11	71.98	71.82	70.61	67.89	60.30	54.69
400	61.01	65.92	66.31	65.54	63.19	56.29	51.05
300	50.13	57.34	58.21	58.06	56.18	50.40	45.69
200	25.00	44.30	45.88	46.47	45.24	40.96	37.13
100	4.14	13.75	24.62	26.16	25.85	23.86	21.53

1080p best quality at far lower data rates than H.264

Lower resolutions don't provide the best quality

Why is HEVC More Efficient?

- Simply a better codec
- One prominent advantage – larger block sizes
  - H.264 – 16x16
  - HEVC – 64x64
- Can encode large images more efficiently



That's Why Apple Has Different Ladders for H.264 and HEVC

Eliminates lower rez rungs

16:9 aspect ratio	H.264/AVC	Frame rate
416 x 234	145	≤ 30 fps
640 x 360	365	≤ 30 fps
768 x 432	730	≤ 30
768 x 432	1100	≤ 30
960 x 540	2000	sam
1280 x 720	3000	same as source
1280 x 720	4500	same as source
1920 x 1080	6000	same as source
1920 x 1080	7800	same as source

720p @ 2400 kbps, not 3000 kbps

16:9 aspect ratio	HEVC/H.265 30 fps	Frame rate
640 x 360	145	≤ 30 fps
768 x 432	300	≤ 30 fps
960 x 540	600	≤ 30 fps
960 x 540	900	≤ 30 fps
960 x 540	1600	same as source
1280 x 720	2400	same as source
1280 x 720	3400	same as source
1920 x 1080	4500	same as source
1920 x 1080	5800	same as source
2560 x 1440	8100	same as source
3840 x 2160	11600	same as source
3840 x 2160	16800	same as source

1080p @ 4500 kbps, not 6000 kbps

Conclusion

- Use different resolutions and switch points for H.264 and advanced codecs

What About Different Types of Content?

- In general:
  - Synthetic videos encode at higher quality at lower bitrates
  - Look better at higher resolutions
    - Push 1080p lower down in the encoding ladder
    - Push 720p further down the ladder
- Not huge difference here, but much more profound for screencams and similar videos

Tears of Steel (real world/CG)

HEVC	1080p	720p	540p	432p	360p	270p	234p
5000	97.67						
4800	97.55						
4600	97.44						
4400	97.31						
4200	97.17						
4000	97.01						
3800	96.84						
3600	96.63						
3400	96.41						
3200	96.15	95.41					
3000	95.86	95.16					
2800	95.52	94.87					
2600	95.09	94.52					
2400	94.58	94.12	92.09				
2200	93.97	93.63	91.62				
2000	93.16	93.02	91.05	88.30			
1800	92.18	92.25	90.34	87.63			
1600	90.94	91.27	89.44	86.78	83.18		
1400	89.36	89.97	88.27	85.69	82.12		
1200	87.30	88.26	86.68	84.22	80.73		
1000	84.42	85.84	84.46	82.20	78.79		
900	82.39	84.21	83.02	80.86	77.51	68.45	62.18
800	80.03	82.20	81.23	79.19	75.91	67.09	60.92
700	77.04	79.67	78.90	77.07	73.91	65.38	59.35
600	73.10	76.34	75.88	74.29	71.36	63.21	57.34
500	68.11	71.98	71.02	70.61	67.89	60.30	54.69
400	61.01	65.92	66.31	65.54	63.19	56.29	51.05
300	50.13	57.34	58.21	58.06	56.18	50.40	45.79
200	25.00	44.30	45.88	46.47	45.24	40.96	37.13
100	4.14	13.75	24.62	26.16	25.85	23.86	21.53

Sintel (animation)

HEVC	1080p	720p	540p	432p	360p	270p	234p
5000	97.83						
4800	97.74						
4600	97.63						
4400	97.50						
4200	97.36						
4000	97.19						
3800	97.01						
3600	96.78						
3400	96.52						
3200	96.22	94.39					
3000	95.86	94.11					
2800	95.45	93.78					
2600	94.94	93.40					
2400	94.32	92.93	89.84				
2200	93.62	92.37	89.34				
2000	92.72	91.69	88.71	85.40			
1800	91.63	90.84	87.94	84.72			
1600	90.21	89.76	87.00	83.84	79.64		
1400	88.44	88.36	86.74	82.74	78.62		
1200	86.02	86.39	84.07	81.24	77.24		
1000	82.81	83.73	81.70	79.13	75.35		
900	80.79	82.02	80.16	77.76	74.10	64.67	58.74
800	78.22	79.83	78.25	76.06	72.55	63.43	57.63
700	75.22	77.22	75.91	73.94	70.64	61.88	56.22
600	71.44	73.84	72.94	71.27	68.17	59.87	54.42
500	66.61	69.68	69.13	67.71	64.90	57.24	52.02
400	60.19	63.98	63.94	62.97	60.47	53.61	48.73
300	48.81	56.19	56.62	56.22	54.16	48.26	43.81
200	26.36	44.11	45.66	53.05	44.22	39.79	36.06
100	5.17	15.45	23.86	26.96	26.53	24.50	21.89

How Can You Use These Techniques

- Per-title encoding (with capped CRF)
- Category-specific encoding
  - What worked
    - Separate ladder for talk shows and sit coms for major OTT producer
      - Proved that 5 Mbps delivered 93+ VMAF for these types of shows
      - Action shows needed 8 Mbps
    - Online training company
      - Separate ladders for screencam/PowerPoint than real world videos
    - Online bike videos
      - Real world needed 1080p/simple yoga videos fine at 720p

How Can You Use These Techniques

- What didn't work
  - Separate ladders for different kinds of movie (action, etc)
    - Just too much differential within each category
  - Separate ladders for animations vs. movies
    - Just too much differential – Sintel vs. Big Buck Bunny vs SpongeBob

Questions?

- Questions
-

Should be 4:00

Current Status of CMAF

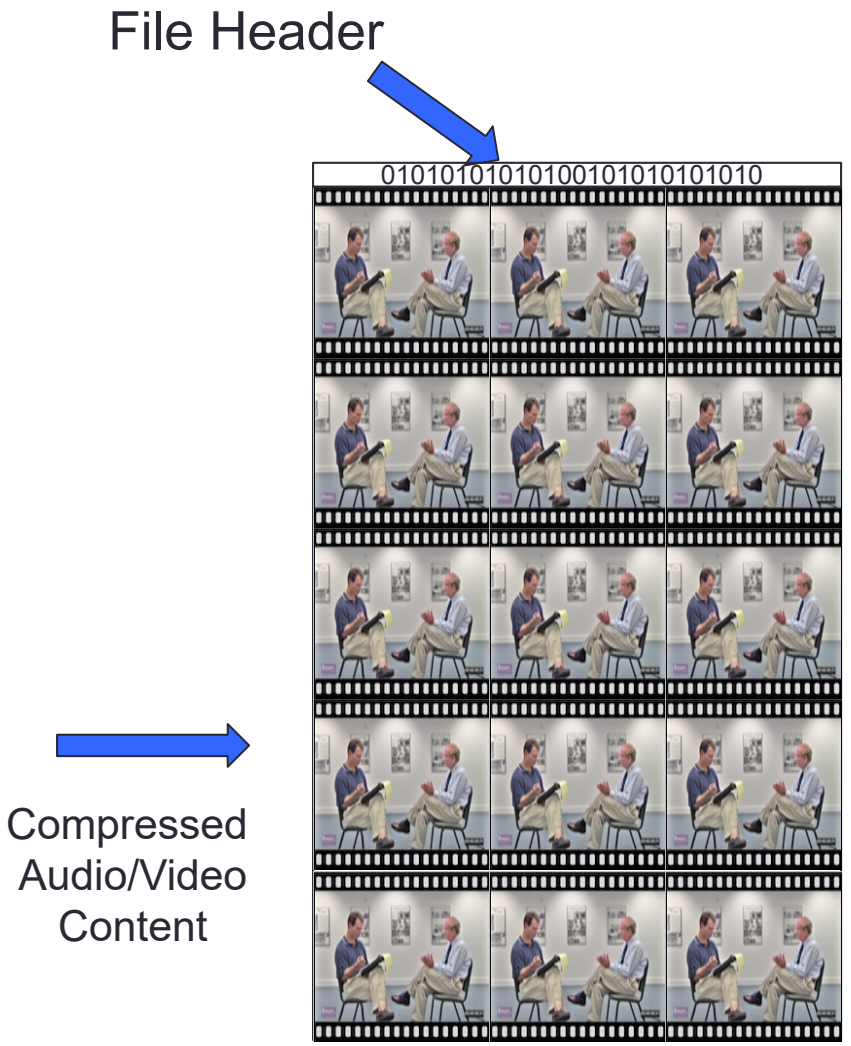
- Codecs vs. container formats
- What is CMAF (courtesy RealEyes Media)
- Current status of CMAF (courtesy RealEyes Media)

Codecs and Container Formats

- **Codecs:** Compression technologies
  - H.264, VP9, HEVC, AV1
- **Container formats**
  - Specs detailing how data/metadata are stored in a file
    - MP4 (DASH), .ts (HLS), .ISMV (Smooth), .F4F (HDS), FLV (Flash)
  - Also called “wrappers”
    - As in, “encoded the file using the H.264 codec in a QuickTime wrapper”
- Why important?
  - File must be in proper container format to play on target platforms

Where is Container Format?

- Text in the file header
  - Very small percentage of overall content
- Can quickly change the container format without affecting A/V content
  - Called transmuxing
  - Critical to operation of tools like Wowza Streaming Engine



Solving the Multiple Format Problem

- HLS (traditionally) needed MPEG-2 transport streams
  - .ts files
  - Now can use fMP4 as well
- DASH uses fMP4
- So, needed two file groups of files, one for HLS (desktop, mobile), one for DASH (OTT, Smart TVs)

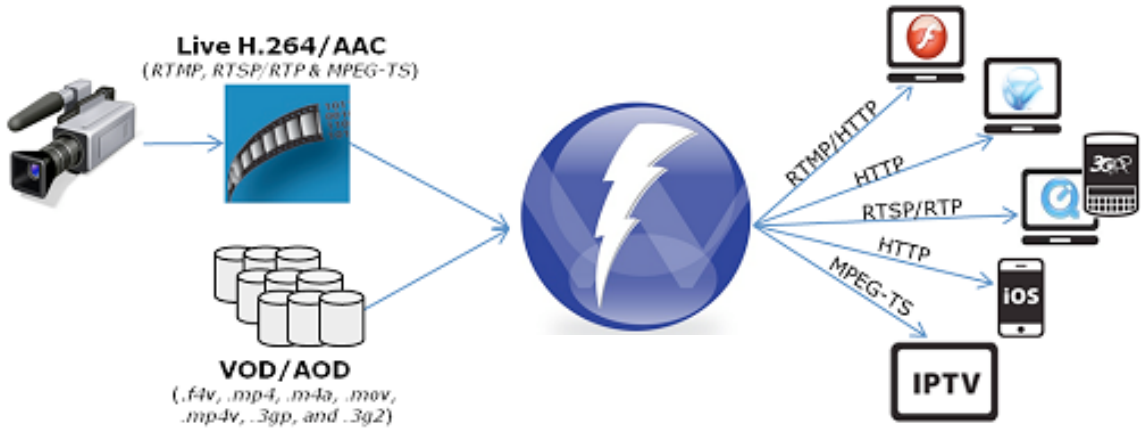
.ts

.mp4

- Double encoding cost and storage cost

Solution 1: Transmuxing

- Single format streams in
  - Live or VOD
- Multiple format output streams customized for target
  - Why so fast and efficient?
    - Just adjusting file header
    - Not changing compressed video data at all
- Issues
  - Need server component (Wowza/Nimble Streamer)
  - Cloud computers 24/7 which gets pricey (much more later)



Solution 2: Common Media Application Format (CMAF)

What is CMAF?

- Specifies the container, NOT the manifest (manifest agnostic)
- It is an ISOBMFF, fMP4 container, specifically ISO/IEC 14496-12:201
- Common Encryption (CENC) - ISO/IEC 23001-7: 2016
- Baseline supports HEVC, ACV, and AAC with interoperability (VP9, etc.)
- Captioning/Subtitling - WebVTT, IMSC-1 & CEA 608/708
- REQUIRES non-muxed audio and video segments

Solution 2: Common Media Application Format (CMAF)

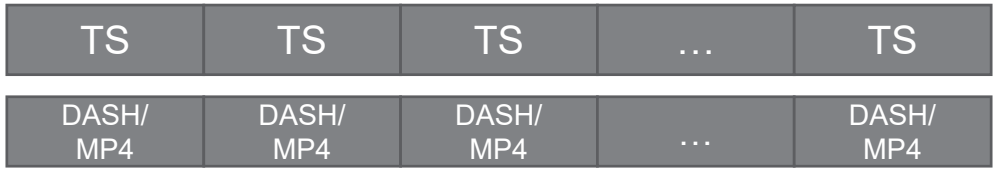
Translation?

- Specifies the container, NOT the manifest (manifest agnostic)
  - **USE HLS OR DASH**
- It is an ISOBMFF, fMP4 container, specifically ISO/IEC 14496-12:201
  - **SAME SEGMENTS WORK IN EITHER - THEY ARE FRAGMENTED MP4 (WHAT DASH USES)**
- Common Encryption (CENC) - ISO/IEC 23001-7: 2016
  - **DRM IS….COMPLICATED - SORRY**
- Baseline supports HEVC, ACV, and AAC with interoperability (VP9, etc.)
  - **CODEC DOESN'T MATTER**
- Captioning/Subtitling - WebVTT, IMSC-1 & CEA 608/708
  - **SAME OLD CC/SUBTITLE OPTIONS**
- REQUIRES non-muxed audio and video segments
  - **OHH….CRAP (MAYBE)**

Solution 2: Common Media Application Format (CMAF)

BEFORE

HLS .m3u8
DASH .mpd



- **Protocol selection was a long-term commitment to manifests and media containers.**
- **Maintaining two complete workflows was expensive**

AFTER

HLS .m3u8
DASH .mpd

- **Protocol selection is now a MANIFEST preference because media containers are common**

turner

Where are We?

Device Support?

- Over 95% of all iOS & tvOS support CMAF (HLS)
  - macOS 10.12+, iOS 10+, tvOS 10+
- Over 97% of all Android support CMAF (DASH/HLS)
- Roku (HLS/DASH)
- All major modern desktop Web browsers (and a bunch of lesser ones too)
  - Requires MSE for playback & EME for DRM

Problem: Encryption

- When launched - two incompatible encryption schemes
  - CBC (Cipher Block Chaining-Apple)
  - CTR (Counter Mode-everyone else)
  - Still need two copies of content
- Since then
  - Google supports CBC in Widevine
  - Playready to support CBC in PlayReady 4
    - https://www.microsoft.com/playready/newsroom/
  - So, one set of files deliverable to HLS and DASH clients once all devices are updated

Before CMAF

DASH

HLS

After CMAF

CMAF/CBC

CMAF/CTR

CMAF Challenges

- CBCS Compatibility
  - Widevine CBCS is not currently supported in Firefox (although they claim it works)
  - Playready CBCS is not currently supported in Edge
  - Likely to change by end of year or next year.
- CMAF Does Not Allow Muxed Audio
  - Both audio and video renditions need to be decrypted
  - Possibility of different decryption keys
- Open Source Player Availability is Limited
  - Shaka Player does not support Playready and Fairplay for HLS [2]
  - Hls.js does not support ANY DRM
  - Both platforms have ongoing efforts for support [3,4,5,6]

CMAF Checklist

- Encoder workflow
- Ads packaging workflow
- Ads trafficking
- DRM if applicable
- Player dev/testing
- Low latency

Questions?

- Questions
-

Should be 4:10

Lesson 7: Dynamic Packaging for VOD and Live

- Static vs. dynamic delivery
- Encoding for static delivery
  - Existing workflow
  - Encoding then packaging
  - Tool options
- Dynamic delivery
  - VOD
  - Live

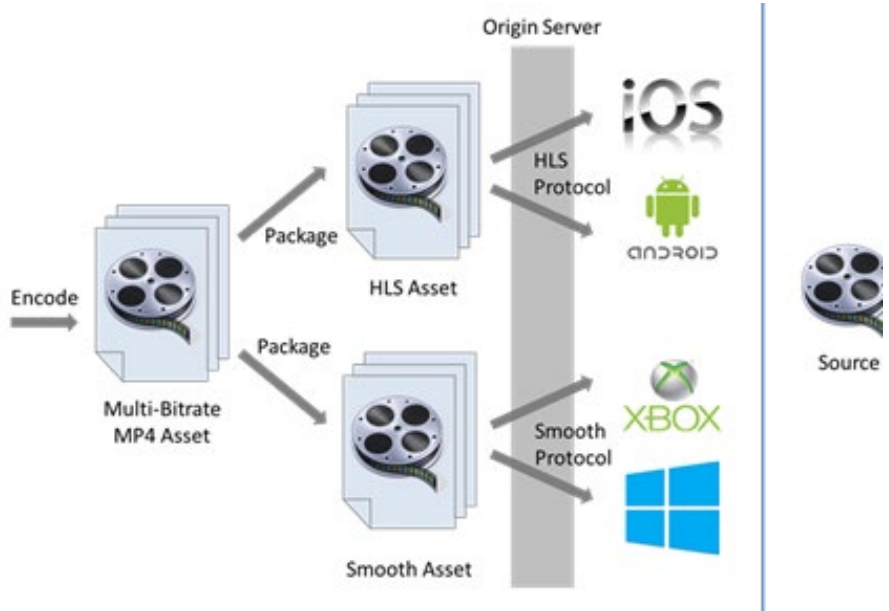
Perspective

- Problem – packaging for DASH, HLS, and (perhaps) MSS
- How to do this most cost effectively?
  - CMAF should solve down the road, but what about today?

Static vs. Dynamic Delivery

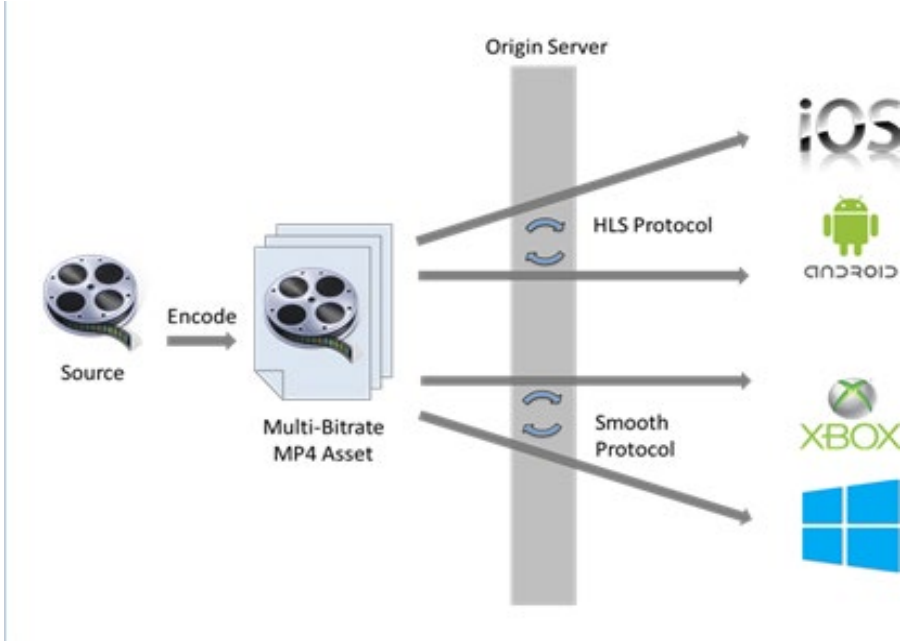
Static

- Create multi-bitrate MP4 files from mezz file
- Create ABR files from multi-bitrate files
- Upload ABR files to server
- Distribute ABR files from origin server



Dynamic

- Create multi-bitrate MP4 files and store on server
- Server dynamically creates ABR chunks and manifest files as needed



Static vs. Dynamic Delivery

Static: Pros/Cons

- Pros
  - Simple, no streaming server required
- Cons
  - Storage intensive
  - Major effort to support new formats
    - Must create new packaged files
    - Upload to servers

Dynamic: Pros/Cons

- Pros
  - Storage efficient
  - Very simple to support new formats/devices down the road
- Cons
  - More technically complex
  - May be more expensive
    - If server component costs more than extra storage + encoding

Static vs. Dynamic

- Consulting project; cloud encoding for library and ongoing
  - Static – increased encoding and storage costs
  - Dynamic – increased server costs (Wowza + cloud instance), but much cheaper overall

Two Year Projections	Static ABR/ Encode Library ASAP	Dynamic ABR - encode library ASAP	Delta
Ongoing cloud encode	$68,651	$49,914	-$18,737
Wowza		$20,405	$20,405
Extra storage for HLS/DASH	$34,009		-$34,009
Total ongoing	**$102,660**	**$70,319**	**-$32,341**
One Time Library Conversion	$125,091	$55,114	-$69,977
Total	**$227,751**	**$125,433**	**-$102,318**

If You Can’t Go Dynamic; Update Workflow

- From a single-step (encode & package)
- To a two-step (encode then package)

Updating The Static Encoding Workflow

Single-step workflow

Upload Mezz file

Encode and package in one step

Two-step workflow

Encode to Multi-Bitrate MP4

Save MP4s as “compressed Mezz files”

Package to HLS/DASH as needed

Single vs. Two-Step Encode

Single: Pros/Cons

- Pros
  - Traditional; no change required
- Cons
  - Have to re-encode existing library
  - Makes supporting new formats can be very expensive and time-consuming

Two-Step: Pros/Cons

- Pros
  - Supporting new ABR formats is fast and cheap
- Cons
  - Increased storage requirements to store complete encoding ladder (but can be cheap storage)

Implementing Two-Step Workflows

Encoder

- Any desktop, enterprise or cloud encoder that can create MP4 files

DASH Packagers

- edash-packager
  - bit.ly/Dash_pack1
- MP4Box - http://gpac.io.
- Rebaca MPEG DASH Segmenter
  - http://bit.ly/Dash_pack2
- castLabs DASH Encrypt Packager
  - https://github.com/castlabs/dashencrypt
- Bento4 - www.bento4.com

Implementing Two-Step Workflows

HLS Packagers

- Apple Media Stream Segmenter (MPEG-2 transport streams)
- Apple Media File Segmenter (MP4 inputs)
  - http://bit.ly/HLS_pack
- Apple Variant Playlist Creator
- FFmpeg – media playlists and packaging only
  - No master m3u8
- Bento4

Other Packagers

- Unified Packager (DASH, HLS, HDS, Smooth)
  - bit.ly/Uni_pack
- ProMedia Package (HLS, Smooth, HDS, DASH)
  - bit.ly/harm_pack

Dynamic Packaging

DIY

- Wowza Streaming Engine
- Nimble Streamer
- Elemental Delta
- Azure Media Services
- encoding.com
- Brightcove
- Many others

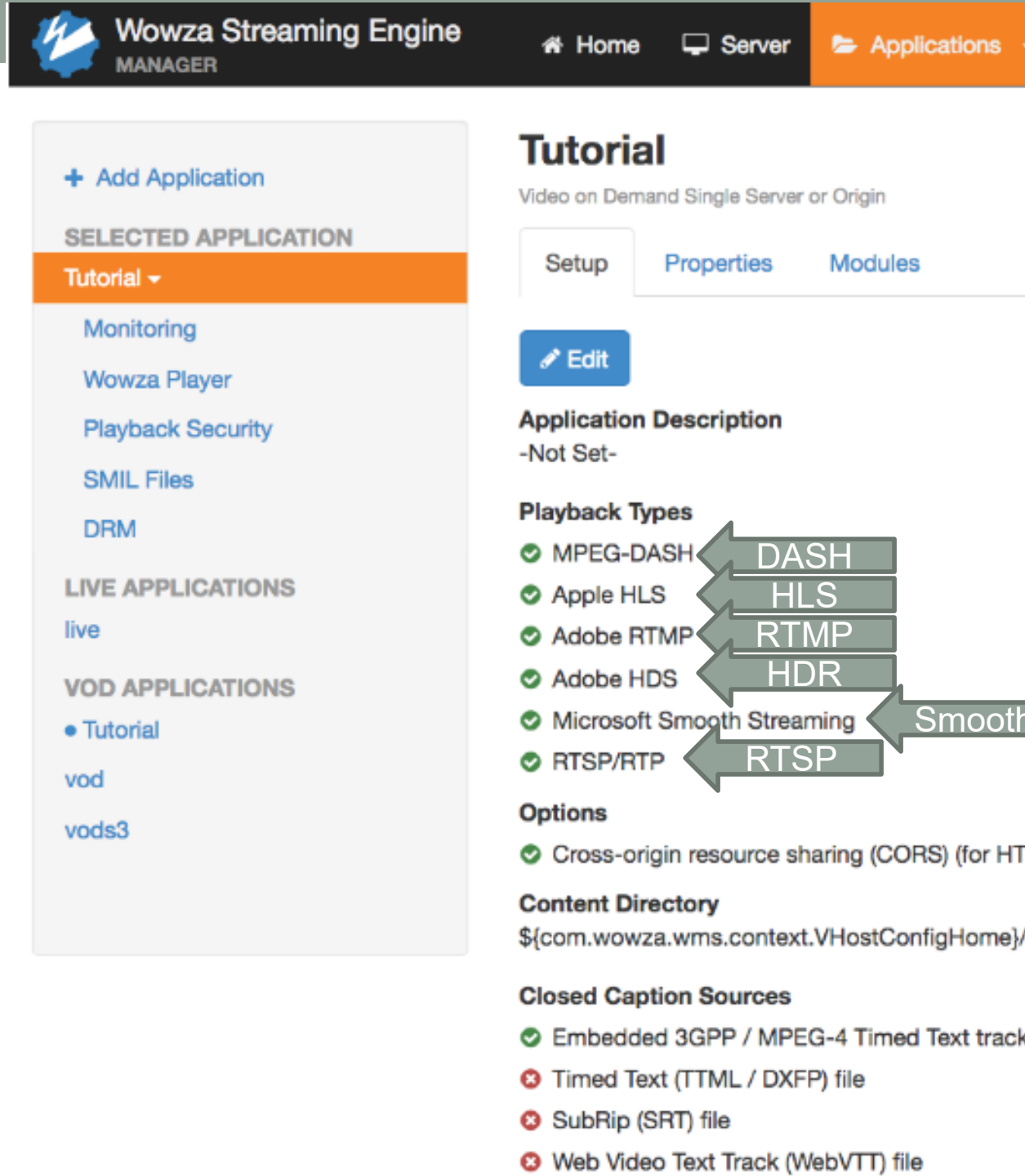
Service Providers

- Akamai
- Limelight

What it Looks Like in Wowza

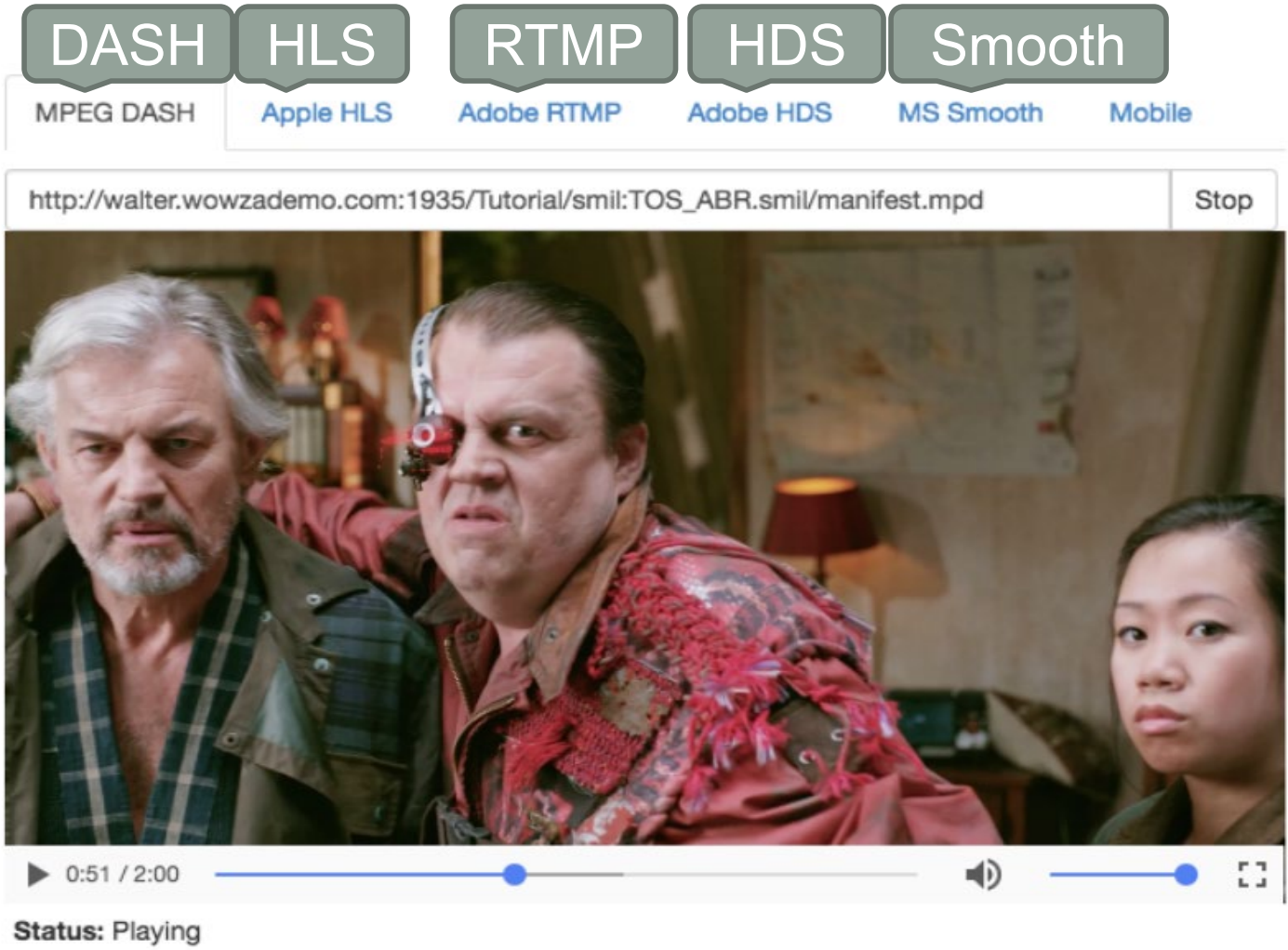
- Upload encoded video ladders in MP4 format
- Choose supported formats
  - DASH, HLS, RTMP, HDS, Smooth Streaming, RTSP/RTP

- Article: Dynamic Packaging with Wowza
  - http://bit.ly/wowza_dynamic



What it Looks Like in Wowza

- Under the hood, Wowza *transmuxes* to the required packaging formats
- Supplies separate URLS for the master manifest for each format
  - DASH
  - HLS
  - RTMP
  - HDS
  - Smooth Streaming
- It's relatively simple to use; no programming required



Live Encoding and Packaging

- Traditional workflow
- Dynamic transcode and packaging

Traditional Workflow

Significant CAPEX

On Site Encoder

HLS Packaged Output

Video Streams			
Codec	Width	Height	Bitrate (kbps)
H.264	1920	1080	4500 (VBR)
H.264	1280	720	3100 (VBR)
H.264	1280	720	2100 (VBR)
H.264	960	540	1500 (VBR)
H.264	640	360	1000 (VBR)
H.264	480	268	550 (VBR)
H.264	320	180	260 (VBR)

DASH Packaged Output

Video Streams			
Codec	Width	Height	Bitrate (kbps)
H.264	1920	1080	4500 (VBR)
H.264	1280	720	3100 (VBR)
H.264	1280	720	2100 (VBR)
H.264	960	540	1500 (VBR)
H.264	640	360	1000 (VBR)
H.264	480	268	550 (VBR)
H.264	320	180	260 (VBR)

Huge outbound bandwidth requirements

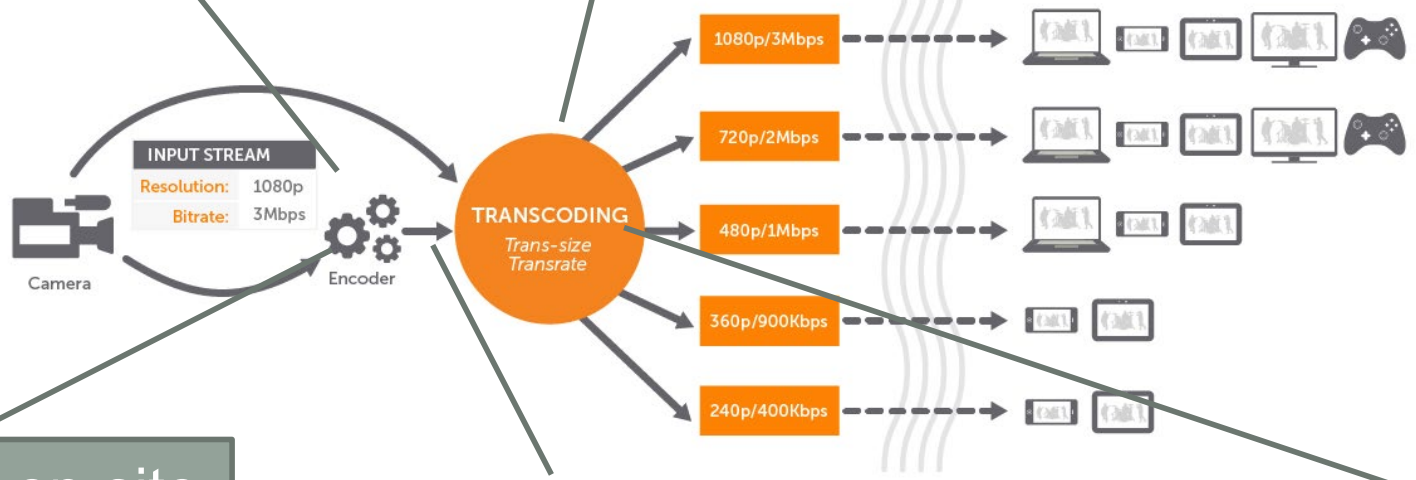
Origin Server in Cloud

Live Transcode/Transmux

Single stream encoder on-site

Transcode in the cloud

Transmux to desired format



Low-cost on-site encoder

Low bandwidth cost

Some transcoding/ transmux OPEX

Live: Traditional vs. Dynamic Packaging

Traditional: Pros/Cons

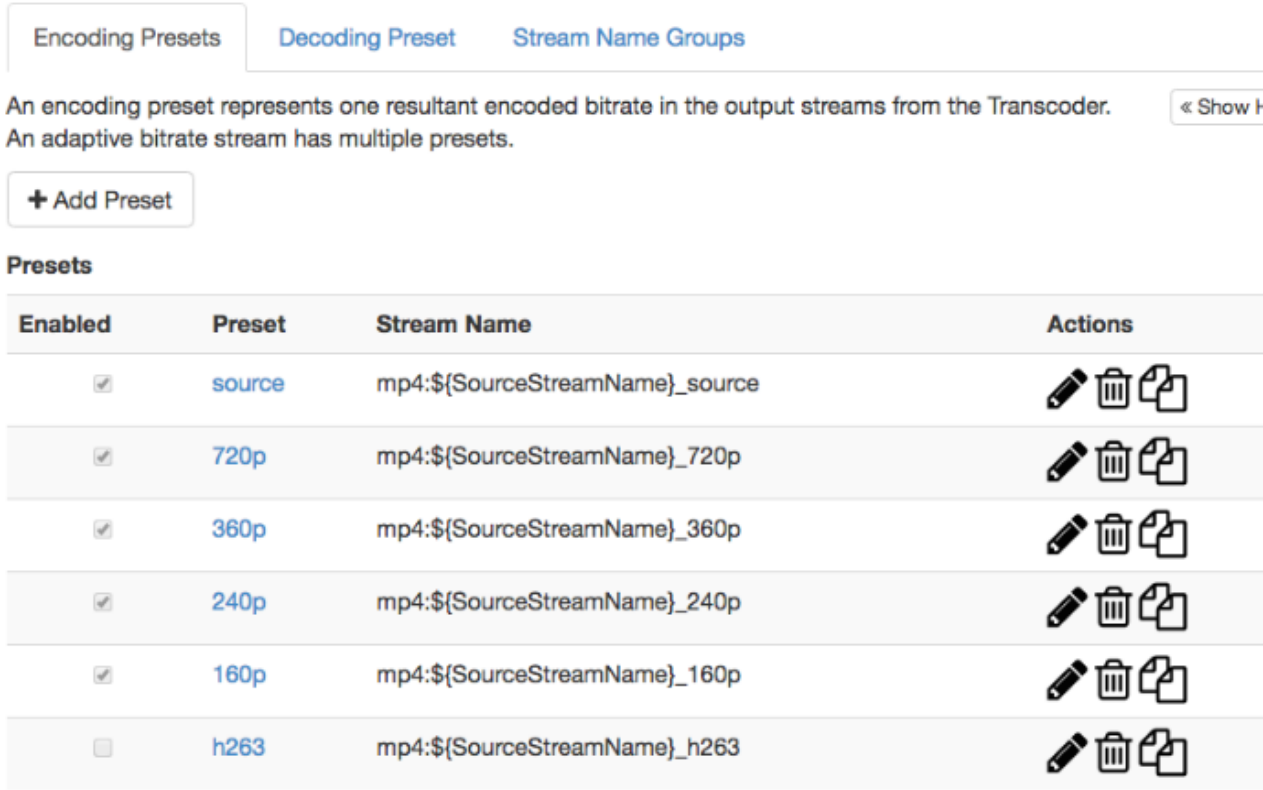
- Pros
  - Simple, no streaming server required
- Cons
  - Significant CAPEX
  - Significant ongoing bandwidth costs
  - Very tough for remote locations

Dynamic: Pros/Cons

- Pros
  - Low CAPEX
  - Low bandwidth (so very event friendly)
- Cons
  - More technically complex
  - Higher OPEX for transcode and transmux
    - Only while live, not 24/7

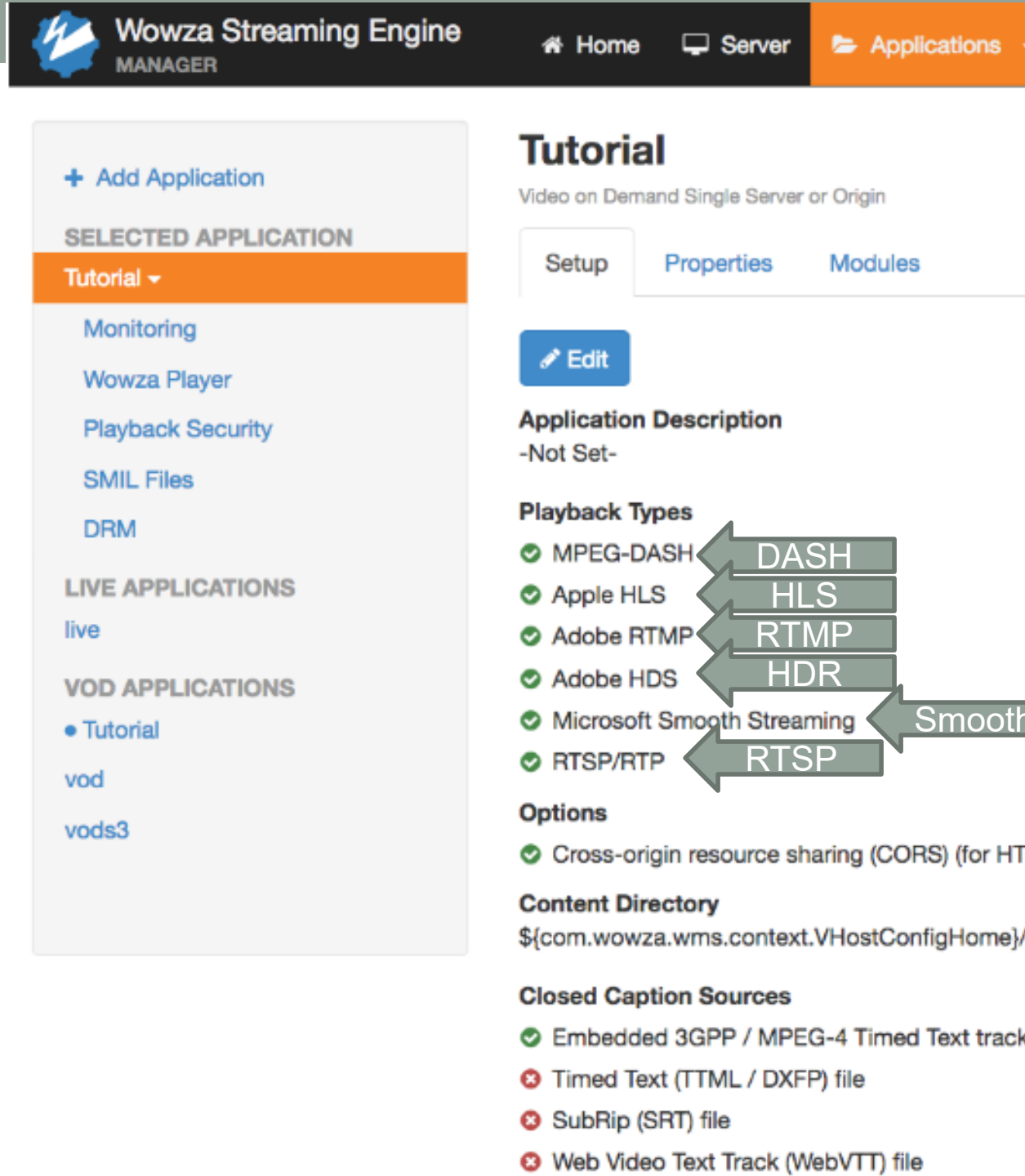
What it Looks Like in Wowza

- Key difference - create encoding ladder in the cloud



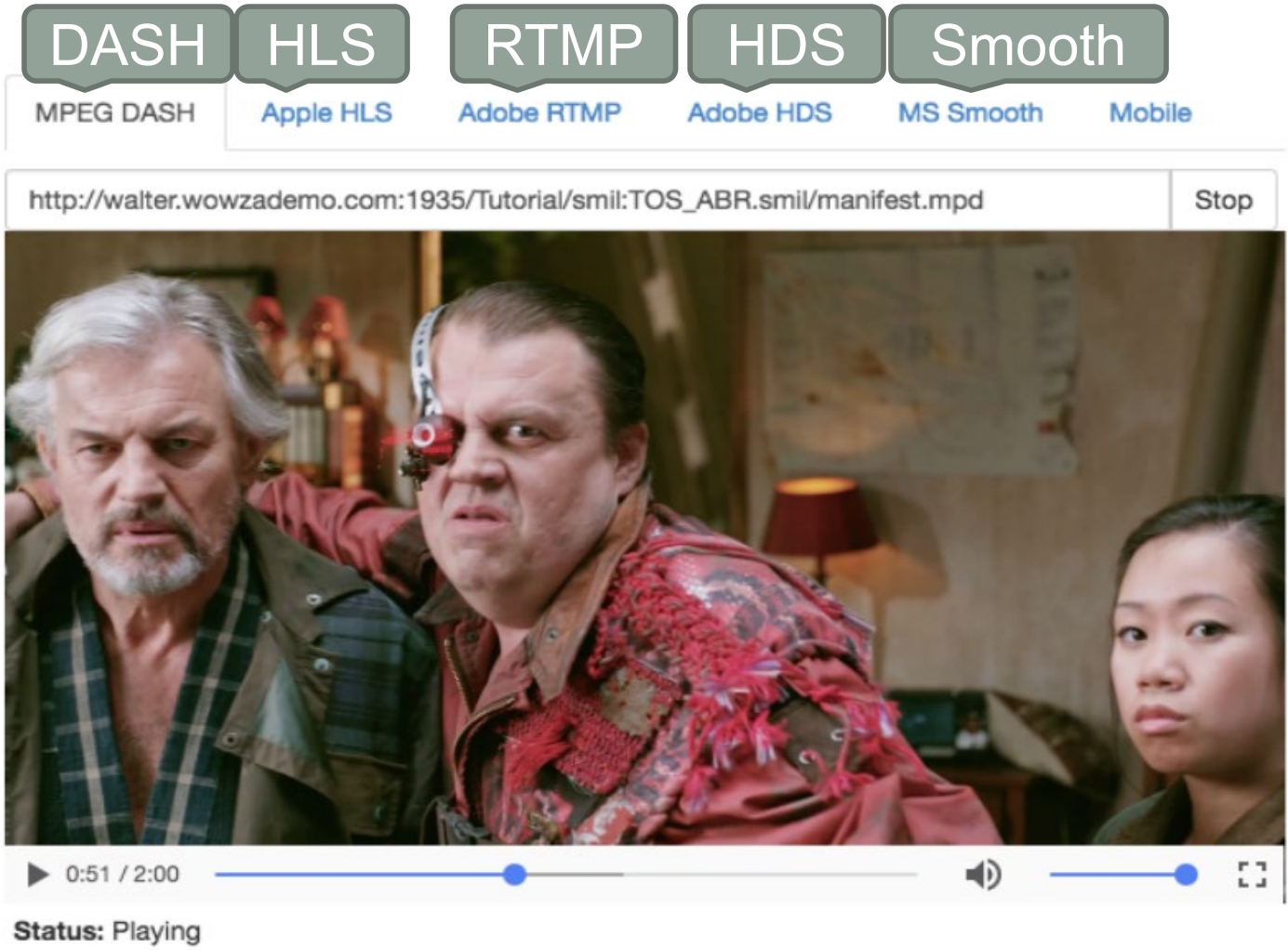
What it Looks Like in Wowza

- Choose supported formats
  - DASH, HLS, RTMP, HDS, Smooth Streaming, RTSP/RTP

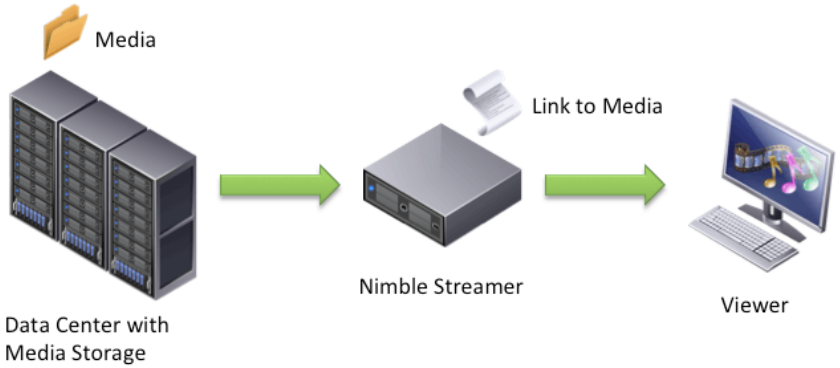


What it Looks Like in Wowza

- Under the hood, Wowza *transcodes* the incoming stream into the encoding ladder
- *Transmuxes* to the required packaging formats
- Supplies separate URLS for the master manifest for each format
  - DASH
  - HLS
  - RTMP
  - HDS
  - Smooth Streaming
- It's relatively simple to use; no programming required



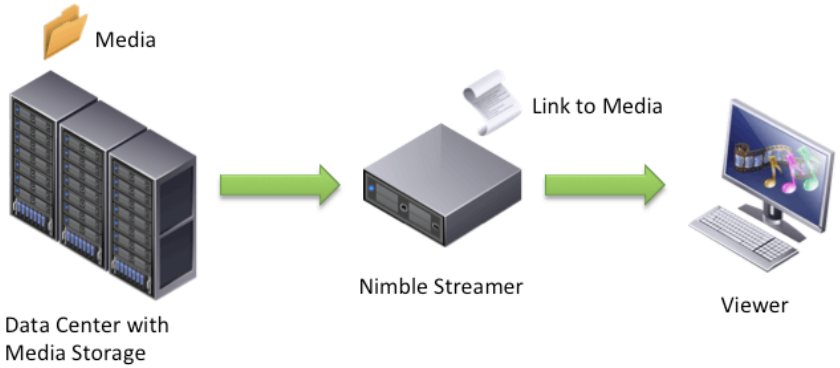
VOD Remote HTTP Origin



- Store original MP4 file on origin available via HTTP (e.g. AWS);
- Set up edge server to packetize from origin
- Download content and package on-demand
  - Only the required parts of original file are downloaded.

- Pros:
  - Lower storage OPEX;
  - Better flexibility;
  - Improved responsiveness to remote clients.
- Cons:
  - Increased OPEX for CPU

DVR for Live and VOD



- Static DVR
  - Save streams into small static chunks in all necessary formats
  - Significant storage hassle and cost
- Dynamic DVR
  - Record into larger files, transmux when requested for playback
    - E.g. using byte range requests
  - Large file, single format

Questions?