ENCODING LIVE AND VOD FOR HEVC/HLS

A Joint SLC/RealEyes Production

Agenda

Our assumptions and goals
Section I: Introduction to HEVC
Section II: Playback performance
Section III: Introduction to HLS
Section IV: Specification overview: HEVC in HLS
Section V: Producing HEVC/HLS

Assumptions and Goals

- Assumptions
 - Have some knowledge of how to produce HLS presentations
- •Goal: Teach you to add HEVC to HLS
 - •Encode HEVC
 - •Choose an HEVC encoding ladder
 - Integrate that into an HLS presentation
 - •With FFmpeg, Bento4 and some third party tools
- •Not a soup to nuts, here's how to do HLS session

Section I. Introduction to HEVC

- •About HEVC
- •HEVC and royalties
- HEVC codecs
- HEVC encoding parameters
- Codec specific encoding profiles

What HEVC Is and Why It's Important

- •HEVC is a standards-based compression technology
- Jointly sponsored by MPEG and ISO standards bodies
 - That's why it's called both HEVC and H.265
- OS support
 - Supported in MacOS via HLS
 - Supported in Windows 10/Edge if hardware decode is available
- Mobile Android and iOS
- Browser support
 - MacOS/Safari, Windows 8/Edge
 - Not supported in Chrome, Firefox, Opera, or Internet Explorer

HEVC and Royalties

- •Three royalty groups (MPEG-LA, HEVC Advance, and Velos)
 - MPEG-LA no royalties on content
 - HEVC Advance royalties on PPV/subscription (up to \$5 million)
 - Velos may be content royalties
- Technicolor also owns HEVC IP, but seems focused on larger entities

HEVC Codecs

- Because HEVC is a standard, there are many compliant HEVC codecs
 - •x265, MainConcept, Intel, Elemental, and many others
 - •x265 is the open-source HEVC encoder included with FFmpeg
 - Widely agreed to be one of highest quality HEVC codecs
 - Can also encode x265 with a separate executable, through FFmpeg is much more flexible, particularly regarding input files

Critical HEVC Encoding Parameters

- •Some parameters apply to all H.265 codecs
 - Profiles
 - No matter which HEVC codec you work with, you'll have to set these
 - Levels
- Some are codec specific
 - Schema for balancing quality and encoding time

What Profiles are and Why They Exist

- Profiles enable different encoding techniques to balance decoding complexity
 - Version 2 codecs use more advanced features
- Apple supports both
 - 1.6. Profile, Level, and Tier for HEVC MUST be less than or equal to Main10 Profile, Level 5.0, High.

Fasture	Vers	sion 1	
reature	Main	Main 10	
Bit depth	8	8 to 10	
Chroma sampling formats	4:2:0	4:2:0	
4:0:0 (Monochrome)	No	No	
High precision weighted prediction	No	No No	
Chroma QP offset list	No		
Cross-component prediction	No	No	
Intra smoothing disabling	No	No	
Persistent Rice adaptation	No	No	
RDPCM implicit/explicit	No	No	
Transform skip block sizes larger than 4x4	No	No	
Transform skip context/rotation	No	No	
Extended precision processing	No	No	

https://en.wikipedia.org/wiki/High_Efficiency_Video_Coding

Main or Main10?

- Main players can't play Main 10 encoded content
 - Some early HEVC players are Main only
 - If encoding for general-purpose playback, use Main
- Main 10 has a very slight quality advantage
 - If encoding for Main10 player, use Main 10
 - Main 10 players can play Main



720p - x265	Main	Main 10	Delta
Tears of Steel	37.05	37.73	1.84%
SIntel	41.37	41.25	-0.29%
Big Buck Bunny	37.21	37.16	-0.13%
Talking Head	41.15	41.15	0.00%
Freedom	39.70	39.57	-0.31%
Haunted	39.56	41.78	5.61%
Average	39.34	39.77	1.12%

HEVC Levels

Level	Max luma sample rate (samples/s)	Max luma picture size (samples)	Max bit rate and Main 10 pro	for Main files (kbit/s) ^[A]	Example picture resolution @ highest frame rate ^[B] (MaxDpbSize ^[C])		
			Main tier	High tier	More/Fewer examples		
1	552,960	36,864	128	-	176×144@15.0 (6)		
2	3,686,400	122,880	1,500		352×288@30.0 (6)		
2.1	7,372,800	245,760	3,000	-	640×360@30.0 (6)		
3	16,588,800	552,960	6,000	-	960×540@30.0 (6)		
3.1	33,177,600	983,040	10,000	-	1280×720@33.7 (6)		
4	66,846,720	0.000.004	12,000	30,000	2,048×1,080@30.0 (6)		
4.1	133,693,440	2,228,224	20,000	50,000	2,048×1,080@60.0 (6)		
5	267,386,880	0,912	25,000	100,000	4,096×2,160@30.0 (6)		

- Set constraints within profiles
- Enable compatibility with lower power devices
- Apple spec No higher than Main10 Profile, Level 5.0, High Tier
 - That peaks at 30 fps for 4K Apple sample streams are 60p
 - Safe to go to 60

Codec Quality/Encoding Time Presets

- Different HEVC codecs use different schemas to simplify quality/encoding time tradeoffs
 - x265 uses presets ultra fast to placebo
 - MainConcept uses a number from 1-28
- •What's important is understanding how the mechanism trades off encoding time and quality

x265 Presets

• Same name as x264; different parameters

	preset	0	1	2	3	4	5	6	7	8	9
	ctu	32	32	64	64	64	64	64	64	64	64
	min-cu-size	16	8	8	8	8	8	8	8	8	8
	bframes	3	3	4	4	4	4	4	8	8	8
	b-adapt	0	0	0	0	0	2	2	2	2	2
0 11 4 1	rc-lookahead	5	10	15	15	15	20	25	30	40	60
0. ultrafast	lookahead-slices	8	8	8	8	8	8	4	4	1	1
1. superfast 2. veryfast 3. faster 4. fast	scenecut	0	40	40	40	40	40	40	40	40	40
	ref	1	1	2	2	3	3	4	4	5	5
	limit-refs	0	0	3	3	3	3	3	2	1	0
5. medium (default)	me	dia	hex	hex	hex	hex	hex	star	star	star	star
6. slow	merange	57	57	57	57	57	57	57	57	57	92
7. slower	subme	0	1	1	2	2	2	3	3	4	5
8. veryslow	rect	0	0	0	0	0	0	1	1	1	1
9. placebo	amp	0	0	0	0	0	0	0	1	1	1
	limit-modes	0	0	0	0	0	0	1	1	1	0
	max-merge	2	2	2	2	2	2	3	3	4	5
	early-skip	1	1	1	1	0	0	0	0	0	0
	recursion-skip	1	1	1	1	1	1	1	1	0	0
	fast-intra	1	1	1	1	1	0	0	0	0	0
	b-intra	0	0	0	0	0	0	0	1	1	1
	sao	0	0	1	1	1	1	1	1	1	1
	signhide	0	1	1	1	1	1	1	1	1	1
	weightp	0	0	1	1	1	1	1	1	1	1
	weightb	0	0	0	0	0	0	0	1	1	1
	aq-mode	0	0	1	1	1	1	1	1	1	1
	cuTree	1	1	1	1	1	1	1	1	1	1
	rdLevel	2	2	2	2	2	3	4	6	6	6
	rdoq-level	0	0	0	0	0	0	2	2	2	2
	tu-intra	1	1	1	1	1	1	1	2	3	4
	tu-inter	1	1	1	1	1	1	1	2	3	4
	limit-tu	0	0	0	0	0	0	0	4	4	0
esets.html											_

http://x265.readthedocs.io/en/default/presets.ht

x265 Presets

	Ultrafast	Superfast	Veryfast	Faster	Fast	Medium	Slow	Slower	Veryslow	Placebo	Total Delta
Tears of Steel	37.25	38.06	38.04	38.05	38.34	38.39	38.84	38.86	38.93	39.00	4.70%
Sintel	35.87	36.89	36.66	36.67	37.11	37.25	37.74	37.79	37.90	37.97	5.86%
Big Buck Bunny	36.10	37.65	37.61	37.60	37.91	38.26	38.70	38.89	39.03	39.18	8.54%
Freedom	38.16	39.01	38.45	38.46	38.71	38.98	39.36	39.44	39.52	39.58	3.72%
Haunted	41.36	41.77	41.39	41.39	41.55	41.68	41.97	41.92	41.97	42.02	1.60%
Screencam	44.03	46.70	46.55	46.54	46.78	47.12	48.31	48.69	48.99	49.34	12.07%
Tutorial	42.46	47.14	46.46	46.42	46.52	47.19	48.35	47.65	48.02	48.53	14.31%
Average	38.64	39.51	39.30	39.31	39.58	39.74	40.13	40.18	40.27	40.35	6.70%

- Ultrafast is always the worst
 - Typically only use when necessary for live encoding

- Superfast is higher quality than Veryfast and Faster
- Starts increasingly steadily after Fast, with Placebo the best

Presets, Quality and Encoding Time



n - nanistra sontano en el cantino en el acor por transferencia el se

% Encoding Time
 % of Overall Quality

- Medium is reasonable for quality and throughput
- Superfast for good quality, fast throughput
- Slow for very good quality, reasonable throughput

MainConcept – Times and Quality



Level

How HEVC compares to H.264

- The assumption:
 - HEVC will work better on newer hardware that supports HW acceleration.
 - HEVC will have good quality with lower CPU consumption when HW acceleration is used.
- The caveat:
 - Many platforms still don't support HEVC: <u>http://caniuse.com/#search=h.265</u>





• iPhone 6 running iOS 11

 It's an older device and the H.264 codec seems much nicer on the CPU

H.264



• iPhone 7 running iOS 11

 On an iPhone 7 you get to leverage the newer and more efficient codec for roughly the same cost

H.264

		1	Activity Mo	nitor (My Pr	ocesses				
) ()	恭 ~	CPU	lemory	Energy	Disk	Network			Q, Search
cess Name		% CPU ~	Memory	CPU Time	Thread	s Idle Wa	ke Ups	PID	User
VTEncode	rXPCService	222.1	374.2 MB	8:02.37		7	82	2970	realeyes
VTDecode	erXPCService	28.4	62.9 MB	1:04.18		6	0	2965	realeyes
Q QuickTime	Player	11.3	25.5 MB	27.65		12	257	2957	realeyes
https://de	mo.bitmovin.com	8.2	128.8 MB	21.03		22	172	2963	realeyes
Screensha	aringAgent	7.8	25.2 MB	20.59		7	125	2937	realeyes
Activity M	onitor	2.2	30.7 MB	2:13.41		6	1	1336	realeyes
Ø Safari		0.7	225.9 MB	16.07		11	1	1322	realeyes
https://de	mo.bitmovin.com	0.1	159.2 MB	46.29	•	12	18	1359	realeyes
🗿 Google Ch	nrome	0.0	72.6 MB	1:01.01		35	1	1592	realeyes
Safari Net	working	0.0	21.9 MB	15.23		7	1	1327	realeyes
SafariBoo	kmarksSyncAgent	0.0	3.8 MB	1.25		4	1	828	realeyes
DisplayLin	kUserAgent	0.0	2.6 MB	0.43		2	0	923	realeyes
Spotlight		0.0	42.0 MB	5.37		6	0	933	realeyes
cfprefsd		0.0	4.4 MB	1.68		2	1	388	realeyes
passd		0.0	18.2 MB	0.50	0	3	1	1168	realeyes
com.apple	.Safari.SafeBrowsi	0.0	6.6 MB	1.15		4	1	1356	realeyes
SystemUI	Server	0.0	36.2 MB	4.94		3	1	900	realeyes
trustd		0.0	10.4 MB	4.81		5	1	423	realeyes
Dock		0.0	20.4 MB	2.94		3	1	899	realeyes
com.apple	CommerceKit.Tran	0.0	6.2 MB	0.50	0	4	0	1317	realeyes
VyprVPN		0.0	21.3 MB	2.71		10	1	898	realeyes
storeasse	td	0.0	6.6 MB	00			PU History		
Google Ch	nrome Helper	0.0	24.0 MB						
keyboards	servicesd	0.0	4.0 MB						
nsurisessi	ond	0.0	8.3 MB						
A Notificatio	on Center	0.0	21.6 MB						
loginwind	ow	0.0	20.6 MB						
parsecd		0.0	9.5 MB				0		
com.apple	.Safari.SearchHelper	0.0	4.2 MB						
akd		0.0	6.9 MB				_		
AppleIDAu	uthAgent	0.0	3.1 MB						
cloudd		0.0	10.6 MB						
commerce		0.0	8.9 MB						
com apple	neod	0.0	9.3 MB	_					
	System:	6.47%		CPU LOAD		Threads		113	8
	User:	68.13%				Processes:		32	6
	Idle:	25.41%							

roces	ss Name	% CPU ~	Memory	CPU Time	Threads	Idle Wake Ups	PID	User
	VTEncoderXPCService	221.5	372.9 MB	17:53.32	7	91	2970	realeves
	VTDecoderXPCService	21.8	56.7 MB	2:44.28	7	1	1365	realeves
0	QuickTime Player	12.3	34.7 MB	1:04.95	13	271	2957	realeves
1	https://demo.bitmovin.com	9.4	140.8 MB	1:08.67	25	185	1359	realeyes
	ScreensharingAgent	7.7	25.2 MB	43.70	7	130	2937	realeyes
	Safari Networking	3.6	27.0 MB	17.55	9	3	1327	realeyes
4.	Activity Monitor	2.2	30.7 MB	2:20.54	5	1	1336	realeyes
0	Safari	0.3	223.2 MB	18.14	10	1	1322	realeyes
	nsuristoraged	0.1	6.8 MB	1.75	4	0	839	realeyes
	cfprefsd	0.0	4.4 MB	1.71	5	0	388	realeyes
	Isd	0.0	10.4 MB	5.15	4	0	422	realeyes
Ç	Google Chrome	0.0	72.6 MB	1:01.11	35	0	1592	realeyes
	SafariBookmarksSyncAgent	0.0	3.8 MB	1.28	4	1	828	realeyes
9	VyprVPN	0.0	21.3 MB	2.74	11	1	898	realeyes
	Finder	0.0	29.0 MB	5.00	3	0	901	realeyes
	distnoted	0.0	4.6 MB	1.28	3	0	425	realeyes
	parsecd	0.0	9.5 MB	1.58	3	0	936	realeyes
	sharingd	0.0	12.4 MB	1.19	4	0	919	realeyes
	https://demo.bitmovin.com	0.0	72.2 MB	24.53	7	0	2963	realeyes
	DisplayLinkUserAgent	0.0	2.6 MB	0.44	2	0	923	realeyes
	Photos Agent	0.0	7.6 MB	0.33	3	1	832	realeyes
	bird	0.0	3.1 MB	00		CPU History	1	
	iconservicesagent	0.0	7.5 MB					
	SystemUlServer	0.0	36.2 MB					
	com.apple.Safari.SearchHelpe	r 0.0	4.2 MB			·		
	com.apple.geod	0.0	9.3 MB					
	VyprVPNUserAgent	0.0	2.6 MB					
	Wi-Fi	0.0	3.4 MB			0		
	storeuid	0.0	4.4 MB					
	ViewBridgeAuxiliary	0.0	2.2 MB					
	nbagent	0.0	3.0 MB			0		
1	LaterAgent	0.0	3.2 MB					
	ако	0.0	0.9 MB					
	nassn	0.0	10.2 MB					
	System:	7.63%		CPU LOAD	Th	reads	117	2
	User:	66.75%			Pri	ocesses:	33	1
	Idle:	25 6 29/	h /	m				

 2010 iMac running High Sierra

 VTDecoderXPCServi ce (video sandox) around 28-30% for H.264 and 22-24% for H.265

• ATI Radeon HD 5670 Hardware acceleration?

ocess Q C V h n k Ç N	Name QuickTime Player CalendarAgent WindowServer		% CPL	J~ N	/emory	CRU Time	Thursda	Idle Wake Line	BID	Licor
C C C C C C C C C C C C C C C C C C C	QuickTime Player CalendarAgent NindowServer					CFO TIME	Threads	fulle wake ops	PID -	0.861
C h n k C N	CalendarAgent WindowServer		33	2.1 1	99.9 MB	30.96	16	812	18608	jun
h n k c	VindowServer		21	1.9	29.6 MB	8.20	14	87	20119	jun
r n k R			21	1.3	17.52 GB	2:45:16.51	10	132	219	_windowserver
r k	https://demo.bitmovin.co	m	16	5.3 3	327.1 MB	4:50.44	23	219	15377	jun
k C N	nds stores		14	4.5	253.1 MB	6:13:38.81	8	77	245	root
S N	ernel_task		10	0.8	1.64 GB	2:04:34.82	144	1,090	0	root
· ·	Messages		5	9.9	74.0 MB	1:38.38	8	129	19670	jun
- P	nds		9	9.9	70.8 MB	3:19:38.49	10	31	89	root
00	Dropbox		8	8.7 1	69.7 MB	42:25.14	150	3	614	jun
	CalNCService		6	5.9	17.1 MB	2.47	4	12	20120	jun
10 A	dail			5.8	1.14 GB	3:38.94	12	21	19976	lun
1	AddressBookSourceSynr		9	5.4	8.2 MB	0.70	9	17	20127	iun
-	Activity Monitor			4.4	43.2 MB	10.74	6	1	20090	iun
	uppaetd			4.1	2 98 GB	37:47 44	27	5	845	iun
0.	afari Networking			3.4	35.8 MB	1:20.89	9	6	15348	jun
~ •	oreaudiod			3.1	7.0 MR	12:38 68	8	07	184	coreaudiod
	urmond			2.8	3.0 MB	12.34	3	7	241	root
	sysmolia			2.4	20.4 MD	2:46 72	6	10	297	iun
(B) z	Tunes			2.2 1	30.4 MD	2.40.73	22	10	10010	jun
	Tunes			2.3	0.0 MD	24.01	32		19919	jun
	ndworker			1.0	3.0 MB	0.37	0	4	20123	jun
	Jode			1.3	49.2 MB	5.42	41	9	19359	jun
C	com.apple.iCloudHelper			1.3	5.3 MB	1.13	5	D	19843	jun
1 9	slack			1.2	71.2 MB	23.93	41	3	19804	jun
n	ndworker			1.0	3.6 MB	0.58	5	4	20124	jun
t	rustd			5.9	12.6 MB	1:44.76	/	4	357	jun
n	ndworker		0	0.9	3.5 MB	0.28	5	4	20132	jun
n	ndworker		(0.9	3.6 MB	0.40	5	4	20131	jun
c	corespotlightd		0	5.8	13.5 MB	5:38.62	7	5	902	jun
n	ndworker		(D.8	3.5 MB	0.29		CP	U Histor	iun V
n	ndworker		(9.8	3.7 MB	0.35	_		-	
c	com.apple.MediaLibraryS	ervi	C	0.7	17.1 MB	12.04				
\$	securityd		(0.7	6.5 MB	2:03.27				
с	coreduetd		(0.6	14.0 MB	4:34.65	_			
f	seventsd		(0.6	8.3 MB	1:56.29				
n	notifyd		(0.5	2.3 MB	1:24.01				lin
n	ndworker		0	0.5	3.5 MB	0.25				
п	ndworker		(0.5	3.6 MB	0.51			0	
O	Google Chrome		(0.5 3	379.8 MB	1:28:45.06				. hl.m.h
٨	AppleSpell		(0.5	64.0 MB	3:19.05				
\$	ecd		(0.4	14.1 MB	18.25			0	
P	nsuristoraged		(0.4	15.4 MB	13.47				
p	outined		(0.3	64.5 MB	1:44.73				him at the
3 S	Spotlight		(0.3	54.6 MB	1:10.36				استوالكالم أدو
	S	ystem:	19.91%		CPU	LOAD	Threads	. :	2406	
	U	ser:	35.41%				Process	es:	480	
	lo	lle:	44.68%		Ν.	Mon	~			

		CPI	Mom	The Monitor	(All Process	Natwor	2			
		Gre	* ODU	Maria Maria	ODU T	There de		010		
Proces	s Name		76 CPU V	Memory	CPU Time	Inreads	Idle wake Ups	PID	User	-
	VTDecoderXPCService	9	28.9	57.1 MB	2:14.19	5	0	15390	jun	1
Q	QuickTime Player		28.0	209.8 MB	1:34.49	16	802	18608	jun	
	WindowServer		27.2	17.63 GB	2:46:27.90	9	205	219	_windowserver	
	mds_stores		21.7	397.4 MB	6:15:31.26	8	68	245	root	1
	https://demo.bitmovin.	.com	19.9	341.4 MB	5:29.42	27	193	15377	jun	
	mds		16.7	71.2 MB	3:20:47.08	9	11	89	root	
10	Mail		12.8	1.43 68	4:27.53	11	22	19976	jun	
50	Messages		10.1	74.0 MB	2:04.03	8	113	19670	jun	
-	kernel_task		9.8	1.63 GB	2:05:29.43	144	1,003	0	root	
-	Activity Monitor		3.7	56.1 MB	26.74	6	1	20090	jun	
	coreaudiod		3.2	7.0 MB	12:48.55	8	97	184	_coreaudiod	
	Safari Networking		3.1	39.7 MB	1:26.90	8	3	15348	jun	
0	sysmond		2.7	3.4 MB	21.54	3	3	241	root	
(J)	Tunes		2.2	181.0 MB	31.35	31	2	19919	jun	
	Dropbox		1.9	185.9 MB	44:34.68	148	3	614	jun	
0	Google Chrome		0.6	379.7 MB	1:28:47.10	41	4	656	jun	
E	Slack		0.3	71.3 MB	26.71	41	1	19804	jun	
	Google Chrome Helper		0.2	157.0 MB	7:25.69	21	6	765	jun	
	notifyd		0.2	2.3 MB	1:24.57	3	0	129	root	
0	Safari		0.2	73.5 MB	50.01	9	1	15345	jun	
A	SystemUlServer		0.1	22.2 MB	1:04.00	4	0	367	jun	
	suggestd		0.1	2.98 GB	37:51.38	25	0	845	jun	
	routined		0.1	64.4 MB	1:45.27	6	1	568	jun	
	Google Chrome Helper		0.1	184.4 MB	1:14.45	13	1	13855	jun	
	CalendarAgent		0.1	31.0 MB	44.35	8	1	20119	jun	
	launchd		0.1	22.0 MB	9:23.45	4	0	1	root	
	powerd		0.1	1.8 MB	42.66	2	1	77	root	
	Google Chrome Helper		0.1	498.1 MB	5:12.17	13	2	18010	jun	
	fseventsd		0.1	8.3 MB	1:56.99	12	2 CPI	RR U Histor	root	
	opendirectoryd		0.1	12.4 MB	1:34.55	-				
~	AppleSpell		0.0	64.0 MB	3:20.92		البرين التاليين		h	
0	Citrix Receiver Helper		0.0	8.0 MB	27.71					
	Isd		0.0	7.6 MB	5:27.43			uinne		-
-	logd		0.0	37.0 MB	2:37.78					
0	Spotlight		0.0	54.5 MB	1:10.69			6. dill		
	coreduetd		0.0	14.0 MB	4:35.25					<u>iii</u>
	Slack Helper		0.0	191.6 MB	1:15.39			0		
	Google Chrome Helper		0.0	46.2 MB	12.81				1	
	syslogd		0.0	2.2 MB	36.81					
-	corespotlightd		0.0	13.4 MB	5:38.99			0		
۲	Adobe Experience Man	ager	0.0	87.6 MB	20.10					
	dbfseventsd		0.0	4.3 MB	1.70					
	cfprefsd		0.0	6.6 MB	38.15					-
		System: 1	9.14%	CPU	LOAD	Threads		2214		
		Liser:	1.65%			Processo		474		
		Idles d	0.018		~ M	rioudsse		-474		
		iole: 4	9.21%		~~~~					

 Mid 2014 Macbook Pro running High Sierra

demo.bitmovin.com
 H.264 around 15-18%

demo.bitmovin.com
 H.265 around19-21% +
 VTDecoderXPCService
 28-30%

• Intel Iris Graphics NO Hardware acceleration?

H.265

Section III: A Brief Introduction to HLS

How HLS works – encoder side
How HLS works – player side
HLS content
HLS manifest files

How HLS Works – Encode Side



- Encoder creates:
 - Multiple sets of segmented video files
 - Index files (M3U8) with file descriptions (rez/data rate/profile) and chunk URLs
- Uploads to HTTP web server

How HLS Works - Player Slde



- Retrieves master index, retrieves segment from first variant listed in master index
- Monitors the buffer status
- Changes streams as needed using index files to find location
 - If heuristics are good, moves to higher quality stream
 - If heuristics are poor, moves to lower quality stream

HLS Content

- •Initially, used the MPEG-2 transport stream (.ts files)
 - Started with separate files (many, many .ts files)
 - Later enabled byte range requests (more later), enabling player to retrieve segments from a single file
 - Much easier to administrate
- •Later, adopted fragmented mp4 files (fMP4)
- •HEVC must use fMP4

Manifest or Playlist Files

Master

- Points to other playlists
- Variant
 - One for each piece of content (audio, video, subtitle, caption) in the HLS presentation
 - Points to actual location of content on the server

I-frame

 Enables trick play, or fast scrubbing backwards and forwards through the file

Master Manifest Files

		ZOOLANDER_1080p.m3u8
#EXTM3U		
#EXT-X-STREAM-INF:PROGRAM-ID=1,	BANDWIDTH=174000,	, RESOLUTION=512x288, CODECS="avc1.42001f, mp4a.40.2"
stream-1-110000/index.m3u8		
#EXT-X-STREAM-INF:PROGRAM-ID=1,	BANDWIDTH=294000,	, RESOLUTION=512x288, CODECS="avc1.42001f, mp4a.40.2"
stream-2-230000/index.m3u8		
<pre>#EXT-X-STREAM-INF:PROGRAM-ID=1,</pre>	BANDWIDTH=544000,	, RESOLUTION=512x288, CODECS="avc1.42001f, mp4a.40.2"
stream-3-480000/index.m3u8		
<pre>#EXT-X-STREAM-INF:PROGRAM-ID=1,</pre>	BANDWIDTH=1063900), RESOLUTION=640x360, CODECS="avc1.42001f, mp4a.40.2"
stream-4-990000/index.m3u8		
<pre>#EXT-X-STREAM-INF:PROGRAM-ID=1,</pre>	BANDWIDTH=2764000), RESOLUTION=852x480, CODECS="avc1.4d001f, mp4a.40.2"
stream-5-1800000/index.m3u8		
<pre>#EXT-X-STREAM-INF:PROGRAM-ID=1,</pre>	BANDWIDTH=4564000), RESOLUTION=1280x720, CODECS="avc1.4d001f, mp4a.40.2"
stream-6-3000000/index.m3u8		
#EXT-X-STREAM-INF:PROGRAM-ID=1,	BANDWIDTH=64000,	CODECS="mp4a.40.2"
stream-0-64000/index.m3u8		

This is the file you link to on your website – first file retrieved

Contains links to "variant" playlists that identify location of media files

- Contains enough data to allow player to choose correct streams
- · Codec/profile, resolution, bandwidth

Traditional Variant Playlist

- Name, location, and duration of all individual files
 - .ts files are MPEG-2 transport streams

🧊 TOS_720p_h.m3u8 - Notepad 📃 🔲 Σ	3
File Edit Format View Help	
<pre>#EXTM3U#EXT-X-VERSION: 3#EXT-X- TARGETDURATION: 7#EXT-X-MEDIA-SEQUENCE: 0 #EXTINF: 6.006000, TOS_720p_h0.ts #EXTINF: 6.006000, TOS_720p_h1.ts #EXTINF: 6.006000, TOS_720p_h2.ts #EXTINF: 6.006000, TOS_720p_h3.ts #EXTINF: 6.006000, TOS_720p_h4.ts #EXTINF: 6.006000, TOS_720p_h5.ts #EXTINF: 6.006000, TOS_720p_h6.ts #EXTINF: 6.006000, TOS_720p_h7.ts #EXTINF: 6.006000, TOS_720p_h8.ts #EXTINF: 6.006000, TOS_720p_h9.ts #EXTINF: 6.006000, TOS_720p_h10.ts #EXTINF: 6.006000, TOS_720p_h11.ts #EXTINF: 6.006000, TOS_720p_h12.ts #EXTINF: 6.006000, TOS_720p_h13.ts #EXTINF: 6.006000, TOS_720p_h14.ts #EXTINF: 6.006000, TOS_720p_h15.ts</pre>	11 III
#EXTINF:6.006000,TOS_720p_h17.ts #EXTINF:6.006000,TOS_720p_h18.ts #EXTINF:5.839167,TOS_720p_h19.ts#EXT-X-ENDLIST	

Variant Manifest - Byte Range Request

Single content file

Easier to administrate

Playlist points to byte ranges within the file
Need HLS version 5 compatible player

TOS_720p_h.m3u8 - Notep	ad		
File Edit Format View	Help		
#EXTM3U#EXT-X-VERSIO #EXT-X-MEDIA-SEQUENC BYTERANGE:2458476@OT #EXTINF:6.006000,#EX BYTERANGE:2733332@24 #EXTINF:6.006000,#EX BYTERANGE:2695168@51 #EXTINF:6.006000,#EX BYTERANGE:2452460@78 #EXTINF:6.006000,#EX	N:4#EXT-X-TARG E:0#EXTINF:6.00 os_720p_h.ts T-X- 58476TOS_720p_ T-X- 91808TOS_720p_ T-X- 86976TOS_720p_ T-X- 339436TOS_720p_	ETDURATION: 06000,#EXT- h.ts h.ts h.ts _h.ts _h.ts	:7 * -X-
#EXTINF:6.006000,#EX BYTERANGE:2943704@13 #EXTINF:6.006000,#EX BYTERANGE:3099932@16 #EXTINF:6.006000,#EX BYTERANGE:2494948@19 #EXTINF:6.006000,#EX BYTERANGE:2687836@22 #EXTINF:6.006000,#EX BYTERANGE:2966828@24 #EXTINF:6.006000,#EX	492008TOS_720p. T-X- 435712TOS_720p. T-X- 535644TOS_720p. T-X- 030592TOS_720p. T-X- 718428TOS_720p. T-X-	_h.ts _h.ts _h.ts _h.ts _h.ts _h.ts	-

I-Frame Manifest

- •Separate .m3u8 file
- Can point to existing media files, or be a video file with I-frames
 - Either way, the player scans the I-frame at the start of each segment
- Requires HLS version 5 player

#EXTM3U
#EXT-X-VERSION:4
#EXT-X-I-FRAMES-ONLY
<pre>#EXTINF:4.12,</pre>
#EXT-X-BYTERANGE:9400@376
segment1.ts
<pre>#EXTINF:3.56,</pre>
#EXT-X-BYTERANGE:7144@47000
segment1.ts
<pre>#EXTINF:3.82,</pre>
#EXT-X-BYTERANGE:10340@1880
segment2.ts

Section IV: Specification Overview

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

Apple Resources

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

HLS Authoring Specification for Apple Devices

About HLS Authoring

About HLS Authoring

About HLS Authoring

http://bit.ly/hls_spec_2017

- HTTP Live Streaming Examples
 - Provides sample streams and manifest files
- We will reference both during presentation

HTTP Live Streaming Examples

http://bit.ly/hls_samps

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

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	6
365	480 x 270	≤ 30 fps	High	4.2	2	6
730	640 x 360	≤ 30 fps	High	4.2	2	6
1100	768 x 432	≤ 30 fps	High	4.2	2	6
2000	960 x 540	source	High	4.2	2	6
3000	1280 x 720	source	High	4.2	2	6
4500	1280 x 720	source	High	4.2	2	6
6000	1920 x 1080	source	High	4.2	2	6
7800	1920 x 1080	source	High	4.2	2	6

* 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	480 x 270	1 fps	High	1	High	1
90	640 x 360	1 fps	High	1	High	1
250	960 x 540	1 fps	High	1	High	1
375	1280 x 720	1 fps	High	1	High	1
600	1920 x 1080	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 are preferred, but can use existing file

Configuration (HEVC)

- Main 10, Level 5, High
 - 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	416 x 234	≤ 30 fps	Main 10	5.0	2	6
300	480 x 270	≤ 30 fps	Main 10	5.0	2	6
660	640 x 360	≤ 30 fps	Main 10	5.0	2	6
990	768 x 432	≤ 30 fps	Main 10	5.0	2	6
1700	960 x 540	30	Main 10	5.0	2	6
2400	1280 x 720	30	Main 10	5.0	2	6
3200	1280 x 720	30	Main 10	5.0	2	6
4500	1920 x 1080	30	Main 10	5.0	2	6
5800	1920 x 1080	30	Main 10	5.0	2	6
8100	2566x1440	30	Main 10	5.0	2	6
11600	3840x2160	30	Main 10	5.0	2	6
16800	3840x2160	30	Main 10	5.0	2	6

* 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	480 x 270	1 fps	High	1	High	1
80	640 x 360	1 fps	High	1	High	1
210	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	416 x 234	≤ 30 fps	Main 10	5.0	2	6
360	480 x 270	≤ 30 fps	Main 10	5.0	2	6
800	640 x 360	≤ 30 fps	Main 10	5.0	2	6
1200	768 x 432	≤ 30 fps	Main 10	5.0	2	6
2050	960 x 540	30	Main 10	5.0	2	6
2900	1280 x 720	30	Main 10	5.0	2	6
3850	1280 x 720	30	Main 10	5.0	2	6
5400	1920 x 1080	30	Main 10	5.0	2	6
7000	1920 x 1080	30	Main 10	5.0	2	6
9700	2566x1440	30	Main 10	5.0	2	6
13900	3840x2160	30	Main 10	5.0	2	6
20000	3840x2160	30	Main 10	5.0	2	6

* 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	480 x 270	1 fps	High	1	High	1
100	640 x 360	1 fps	High	1	High	1
250	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=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",

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=533420.BANDWIDTH=582820.CODECS="avc1.64001f,mp4a.40.2",

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=303898,BANDWIDTH=339404,CODECS="avc1.64001f,mp4a.40.2",

#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

RESOLUTION=640x360, FRAME-RATE=30.000, CLOSED-CAPTIONS="cc", AUDIO="a1", SUBTITLES="sub1"v3/prog_index.m3u8

RESOLUTION=480x270,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v2/prog_index.m3u8

RESOLUTION=416x234,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v1/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=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=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

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",

#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

RESOLUTION=1920x1080,FRAME-RATE=60.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v18/prog_index.m3u8

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=3441312,BANDWIDTH=4079770,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=2635933,BANDWIDTH=2764701,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",

#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",

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=522229,BANDWIDTH=548927,CODECS="hvc1.2.4.L123.B0,mp4a.40.2",

#EXT-X-STREAM-INF:AVERAGE-BANDWIDTH=314941,BANDWIDTH=340713,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

RESOLUTION=1280x720.FRAME-RATE=60.000.CLOSED-CAPTIONS="cc".AUDIO="a1".SUBTITLES="sub1"v15/prog_index.m3u8

RESOLUTION=640x360.FRAME-RATE=30.000.CLOSED-CAPTIONS="cc".AUDIO="a1".SUBTITLES="sub1"v12/prog_index.m3u8

RESOLUTION=480x270,FRAME-RATE=30.000,CLOSED-CAPTIONS="cc",AUDIO="a1",SUBTITLES="sub1"v11/prog_index.m3u8

,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"

Section V. Producing HEVC/HLS

•DIY – VOD

- FFmpeg create the A/V files
- Bento4 package and manifest files

Third party alternatives

- Live
- VOD

Creating HEVC Files in FFmpeg

•Use the x265 codec

· Widely recognized as one of the fastest and highest quality

Need to compile Main10-specific version

•All scaling and other syntaxes apply

•Need to choose profile and preset (unless defaults OK)

•Must use –x265-params command for some parameters

Encoding x265 in FFmpeg

ffmpeg -y -i TOS_1080p.mov -c:v libx265 -preset slow -x265-params profile=main:keyint=48: min-keyint=48:scenecut=0:ref=5:bframes=3:b-adapt=2:bitrate=4000:vbv-maxrate=4400:vbv-bufsize=4000 -an -pass 1 -f mp4 NUL && \

ffmpeg -i TOS_1080p.mov -c:v libx265 -preset slow -x265-params profile=main:keyint=48: min-keyint=48:scenecut=0:ref=5:bframes=3:b-adapt=2:bitrate=4000:vbv-maxrate=4400:vbv-bufsize=4000 -an -pass 2 TOS_1080p_h.mp4

ffmpeg -i TOS_1080p.mov -c:v libx265 -s 1280x720 -preset slow -x265-params profile=main: keyint=48: min-keyint=48:scenecut=0:ref=5:bframes=3:b-adapt=2:bitrate=1000:vbv-maxrate=1100: vbv-bufsize=1000 -an -pass 2 TOS_720p_1.mp4

Integrate x265 commands into FFmpeg

-x265-params - start of x265 commands, in x265 syntax

- http://x265.readthedocs.io/en/default/
- One string of commands, separated by colon, no spaces until finished

- Preset, an (audio no), pass, format, and Null outside of this structure
- Scaling commands outside of –x265-params structure

FFmpeg Learning Resources

- Includes H.264/H.265
 - Creation of variant playlists with FFmpeg
 - Variant/master playlists with Apple tools
 - No Bento
 - No cloud stuff
- D103 HOW TO: Building a More Robust Cloud Encoder With FFMPEG & More
 - Thus 1:45 2:30



http://bit.ly/ffmpeg_30

Introduction to Bento4

- What it is: A fast, modern, open source C++ toolkit for all your MP4, HLS, and MPEG DASH media format needs
 - <u>https://www.bento4.com/</u>
 - Documentation for HLS <u>https://www.bento4.com/developers/hls/</u>
- What you can do with Bento4
- Bento 4 vs. FFmpeg
- HLS options and Bento4 sytax

What can I do with Bento4?

- HLS generation, including master manifests, stream level manifests, mpeg-2 ts files, and fMP4 (fragmented MP4)
- MP4 to fMP4 conversion
- DASH generation
- Parsing and multiplexing of H.264 and AAC streams
- Support for DRM (Marlin, PlayReady, Widevine and FairPlay).
- Support for H.264, H.265, AAC, AC3, eAC3, DTS, ALAC, and other codec types.
- Dual generation of HLS and DASH from fragmented MP4
- Atom/box editing, and stream/codec information
- A lot more... https://www.bento4.com/

Bento4 vs FFMPEG

- Bento4 focuses on MP4 based content: Packaging & Transmuxing
- FFMPEG is a broad spectrum tool for media conversion, encoding & packaging

HLS options

- Master playlists
- Single file output with byte range requests
- I-Frame only playlists
- AES encryption
- DRM
- Audio stream sidecar
- Subtitle sidecar
- fMP4

Create Multiple Bitrate Assets

mp4hls --hls-version 4 input_7000kb.mp4 input_5000kb.mp4 input_3500kb.mp4

- Outputs:
- Master.m3u8
- Stream.m3u8 for each bitrate
- Iframe.m3u8 for each bitrate
- ts fragments for each bitrate

Multiple Audio Streams

mp4hls video.mp4 spanish_audio.m4a (different audio file)
mp4hls video.mp4 [+language=es]audio.m4a (multiplexed audio file, getting the spanish stream)

Outputs:

- Master.m3u8
- Stream.m3u8 for video and audio
- Iframe.m3u8 for video and audio
- ts fragments
- Audio.m3u8 and aac fragments

WebVTT Subtitles

mp4hls video.mp4 [+format=webvtt,+language=en]english.vtt

<u>Outputs</u>

- Master.m3u8
- Stream.m3u8
- · Webvtt manifest and .vtt file

Encryption and Single Segment

mp4hls --hls-version 4 --output-single-file --segment-duration 6 --encryption-mode AES-128
 --encryption-key abaa09cd8c75abba54ac12dbcc65acd7 --encryption-url
 <u>http://getmyKey?token=token</u> video.mp4

<u>Outputs</u>

- All HLS assets (master, stream with byterange requests, iframe, single ts file)
- Assets are encrypted with AES-128, and encryption URL is added to the stream manifests
- Segment duration will be set to 6 seconds, but will only segment at the closest i-frame

Dual HLS and DASH From fMP4

mp4fragment input.mp4 output.mp4 (converts mp4 to fmp4)
mp4dash --force --hls --no-split --use-segment-timeline output.mp4
(without --no-split it will output .m4s segments)

<u>Outputs</u>

- Master.m3u8
- Audio.m3u8
- Video.m3u8
- Stream.mpd (DASH manifest)

Dual HLS and DASH From fMP4

DEMO Let's see this happen

Example Master Playlist for Single Bitrate

#EXTM3U #EXT-X-VERSION:6

Media Playlists

Audio

#EXT-X-MEDIA:TYPE=AUDIO,GROUP-ID="audio/mp4a",LANGUAGE="en",NAME="English",AUTOSELECT=YES,DEFAU LT=YES,URI="audio-en-mp4a.m3u8"

Video

#EXT-X-STREAM-INF:AUDIO="audio/mp4a",AVERAGE-BANDWIDTH=3454711,BANDWIDTH=4209761,CODECS="avc1. 640020,mp4a.40.2",RESOLUTION=1280x720 video-avc1.m3u8

Other Info

- Bento will only segment at an i-frame
- Creates HLS assets faster than ffmpeg or shaka packager
- Gathers its metadata while segmenting, so codecs, average bandwidth, bandwidth, and resolution are automatically added to the manifests
- A full set of DASH and metadata options

List of all Bento4 binaries: <u>https://www.bento4.com/</u>

VOD: Server-based HEVC/HLS Asset Generation

- Overview
- Sizing your server
- Our experience
- Hardware starting point
- GPU pipeline
- Getting the software

Implementing Steps

- VOD: Server-based HEVC/HLS asset generation
- Cloud workflow
- Scaling
- Cloud encoding (the server)

OVERVIEW

- Choose your Cloud:
 - AWS
 - Azure
 - RackSpace
 - IBM SoftLayer
- Or don't (On-prem)
- Or a hybrid (e.g. On-prem and S3)

SIZING YOUR SERVER

General

• What general bitrates are you dealing with?

Live

How many concurrent live streams?

Are you also transcoding optional renditions for ABR?

• VOD

• How many concurrent videos being processed?

- Is it transcoding or just transmuxing?
- Do you need to create sidecar assets?

OUR EXPERIENCE

 In AWS we've found m3.large to be a pretty cost effective, decently performant and reliable instance size

 We made our decision in Azure based on AWS and went with as similar a match we could find, DS2_V2

• We use Linux as our base since it's friendlier with our software stack. Mostly RHEL.

STARTING POINT

• Get started with ec2 instances:

http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html

 Get started with Azure VMs: <u>https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-linux-quick-create-portal/</u>

GPU PIPELINE

Offload processing from CPU to dedicated hardware

• FFmpeg has some support for GPU Acceleration

You need to have specific supported hardware

• Example: AWS EC2 g2.2xlarge + CUDA + FFmpeg with -hwaccel option specified

GETTING THE SOFTWARE

You'll need to download and install software

- Our preferred toolset:
 - Bento4/FFmpeg (Video processing and Static Builds are easy install)
 - ImageMagick (spritesheets, thumbnails and image manipulation)
 - Node.js (You need an application server wrapper)
 - MongoDB (You need some data persistence)
 - Cloud Provider SDK (e.g. AWS SDK for JavaScript in Node.js)

Cloud Workflow: Making it Happen

- Designing a workflow API
- Workflow: file transfer
- Workflow: queue
- Open source libraries
- Sample code
DESIGNING A WORKFLOW - API

- You need a good workflow architecture
- Similar to AWS Simple Workflow Service for logical and atomic chunks:
 - Workflow (End to End Execution)
 - Steps (Ingestion, Processing, Transfer)
 - Tasks (Create alternate bitrate rendition, Thumbnails)
 - Adaptors (We added this to be agnostic.
 - E.g. AWS S3 vs. Azure Blob vs. On-prem)

WORKFLOW: FILE TRANSFER

- Try to leverage any performance enhancements available
- Day to Day Ingestion
 - AWS Multipart Upload
 - Azure Streaming Put a BlockBlob
- Initial Content Migration
 - AWS Import/Export Snowball
 - Azure Import/Export Service

WORKFLOW: QUEUE

- Gracefully handle all your users
- Processing takes time. You need to line up requests.
- Queuing w/persistence also lets you keep track of job status and what's pending in case of restart.

OPEN SOURCE LIBRARIES

• When there's a vibrant community you never have to reinvent the wheel

- We use Node.js which has node modules.
 - aws-sdk: AWS JavaScript Library for Node.js
 - fluent-ffmpeg: A node wrapper for the FFmpeg command line tool
 - q: A node promise library
 - async: Asynchronous JavaScript helper

SAMPLE CODE

Check out the demo: <u>https://github.com/realeyes-media/demo-encoder</u>
Here's a snippet

```
input.inputOptions = options.inputOptions;
output.outputOptions = ["-hls_time 8", "-hls_list_size 0", "-bsf:v h264_mp4toannexb", "-threads
0"];
input.inputURI = path.join(__dirname, '../../' + options.inputURI);
output.outputURI = directory + '/' + options.fileName + options.timestamp + '_' + bitrate + '.'
+ options.outputType;
options.outputURI = output.outputURI;
output.outputOptions.push('-b:v ' + bitrate + 'k', '-r ' + options.fps);
// Use options to call ffmpeg executions in parallel
```

executeFfmpeg(input, output)

Scaling

- Scaling and concurrency
- Scaling multiple instances
- Multi-instance balancing
- Auto-scaling
- Container swarms

SCALING & CONCURRENCY

How high can we go?

FFmpeg will not error when the CPU is busy, just takes longer to process.

• First - Determine the Scenario:

- The volume of files you need to simultaneously process
- The average size of the files you need to process
- The processing time that's acceptable for you org
- The kinds of operations that need to occur (e.g. Just transmux? Transcode to 4 renditions?)
- Second Run Performance Tests

SCALING - MULTIPLE INSTANCES

• Bigger instance or more instances?

Bigger Instance

- PRO: Handles more concurrency
- CONS: Can be more costly

More Instances

- PRO: Cheaper Can be scaled up and down to only pay when needed
- CONS: More complicated to manage

MULTI INSTANCE BALANCING

- Scale Horizontally Transparently
 Clients hit a load balancer
- You can add more instances as needs grow in a transparent and simple way
- If your architecture is sound there's no need for session stickiness between the clients and the transcoding system
- AWS Elastic Load Balancer: <u>https://aws.amazon.com/elasticloadbalancing/</u>
- Azure Load Balancing: <u>https://azure.microsoft.com/en-us/documentation/articles/virtual-machines-linux-load-balance/</u>

AUTO-SCALING

- Leverage Auto Scaling Features
- Automate the spin up/down of instances based on a number of criteria:
 - Instance Load
 - Periodic Need for Faster Processing
 - Time of Day
 - Specific Events
- AWS Auto Scaling: <u>https://aws.amazon.com/autoscaling</u>
- Azure Auto Scale:

https://azure.microsoft.com/en-us/documentation/articles/cloud-services-how-to-s cale-portal/

CONTAINER SWARMS

- Docker is all the rage. Swarms and Service Discovery
- Create a swarm of Docker containers for a highly repeatable processing server snapshot that utilizes system resources efficiently
- Further increase automation through service discovery
- Implement "auto scaling" on steroids

Cloud Encoding (The Server)

•>>> DEMO <<<

LIVE: Streaming with HEVC/HLS

- x265 Boost from Intel Xeon Scalable processor family
- Wowza
- Encoding basically it comes down to hardware or cloud

HEVC Live – Intel Scalable Processor Family

<u>x265 Boost from Intel Xeon Scalable</u> <u>Processor Family</u>

- x265 show a 67% average per-core gain for encoding using HEVC Main profile
- 50% average gain with Main10 profile across different presets





• Wowza:

https://www.wowza.com/docs/how-to-stream-using-hevc-h-265-transcoding



Live 4K HEVC/H.265 Software Encoding

 Haivision demoed live 4Kp60 HEVC software-only (x265) performance video streaming w/off the shelf hardware

• In the end it all comes down to hardware for live

More Demos

- Manifest Demo
- Playback demo and discussion (H.265 only)
- Playback demo and discussion (mixed H.264 and H.264)
- Playback demo and discussion (H.264 only)
- Additional resources

Manifest Demo: Walking through VOD and LIVE HEVC/HLS during playback (manifest viewer)

Manifest Demo: Walking through VOD and LIVE HEVC/HLS during playback (manifest viewer)

Playback Demo/Discussion: H.265 only

Playback Demo/Discussion: Mixed H.265 + H.264

Playback Demo/Discussion: H.264 only

Resources

• Slides: http://bit.ly/2gwlYs5

Third Party Alternatives

•Live

- Full transcode and package
- Contribution
- Cloud transcode

•VOD

- Appliance
- Software
- Cloud

Live: Full Transcode and Package

- DVEO Gearbox265
- Elemental Live
- Harmonic Electra XT
- Harmonic VOS Cloud Software
- Telestream Vantage Lightspeed

Full Transcode and Package: DVEO Gearbox265



Hardware appliance

Full Transcode and Package: Elemental Live



 Linux-based software; deploy anywhere

Full Transcode and Package: Harmonic Electra XT, X2, X2S, VS



 Linux-based software; deploy anywhere

Cloud Transcode: Harmonic VOS Cloud Software



- Licensed software
- Deploy in OpenStack or AWS

- No pricing info on website
 At Streaming Media West
- Live and VOD

Full Transcode and Package: Telestream Lightspeed Live Stream



 Linux-based software; deploy anywhere

Live Contribution

Harmonic

•LiveU

Teradek

Cloud Transcode: Harmonic ViBE 4K



Hardware/VODNeeds external packager for HLS

Contribution: LiveU





Cube 755 \$2,990 (Ethernet + Wi_Fi)



Slice 756 \$3,990 (Ethernet + Wi_Fi)

HEVC Pro Card (for LU) 600 \$2,790 (Ethernet)

Contribution: Teradek





Cube 705 \$2,790 (Ethernet) Cube 755 \$2,990 (Ethernet + Wi_Fi)



Slice 756 \$3,990 (Ethernet + Wi_Fi)

Live Cloud Transcode

Harmonic VOS 360 cloud service

• Wowza

Cloud Transcode: Harmonic VOS 360 Service

VOS 360 ECOSYSTEM



 Linux-based software; deploy anywhere
Wowza

- •Can transcode to HEVC/not yet compliant with HLS spec
 - No CMAF yet
 - Here at show; ask when they will have

HEVC, HLS, and Live Production: A Wowza Interview



Wowza VP of Engineering Barry Owen

http://bit.ly/wz_hls



ApplianceSoftwareCloud

Appliance: AWS Elemental Server



 Linux-based software; deploy anywhere No pricing info on websiteAt Streaming Media West

Software: Vantage Media Processing Platform



Can run on servers or on public and private virtualized infrastructures
At show

Cloud: AWS Elemental Cloud



• True cloud-based product; extensible with other products No pricing info on websiteAt Streaming Media West

Software/Cloud: Bitmovin Video Encoding



 Available as a SaaS offering or for internal deployment No pricing info on websiteAt Streaming Media West

Cloud: Hybrik Cloud

🔍 🔍 🖉 🕒 🕞 harmonic "e 🗙 🜔 M	ledia Proce 🗙 👍 HEVC - Tere 🗙 Hy Hybrik Pre: 🗙 🕅 Transcoder 🗴 💋	Single and IN X D Real Time H X Jan
← → C 🔒 Secure https://der	mo.hybrik.com/wf/main#/presets/transcode 😽 @	λ 🛧 🖸 🚰 📄 👘 📣 🎯 🖬 😁 🗄
🗎 OO testing 🔺 Bookmarks 🤱 Lea	rn to Produce Vi 🤱 Video Encoding by t 🤱 Producing Streamin 💄 Maste	aring Webcam » 🗎 Other Bookmarks
Hybrik	Presets / Transcode	⊕ LogOut
175111	Action -	□ ≎
Jan Ozer	Name 👙	
	System Presets	1
🐵 Dashboard	🕒 Broadcast	
	Distribution	
Storage <	Apple HLS	
Q Media Analyzer	MPEG-DASH	
	Microsoft Smooth Streaming	
≓ Tasks 〈	🔁 WebM	
	🖶 mp4	
🛢 Jobs <		
🗅 Watch Folders	Preset Details Preset History	
i≣ Presets ~		
Transcode Presets	Copyright Hybrik Inc © 2014-2017	

ALL HYBRIK PLANS INCLUDE:

- Dedicated Machines 24/7/365
- Virtual Private Cloud
- Total Control

- Accelerated Transfers
 Easy-to-Integrate API
- Email and Phone Support
- Transcoding and QC





Currently VOD; moving to live

At Streaming Media West

Other Vendors

Live

Contribution

Vitec – multiple encoders

•VOD

- SDKs
 - Beamr
 - MainConcept
 - Multicoreware