



# OPTIMIZING MULTI-SCREEN (AND MULTI-CODEC) STREAMING DELIVERY

YURIY REZNIK | [YREZNIK@BRIGHTCOVE.COM](mailto:YREZNIK@BRIGHTCOVE.COM)



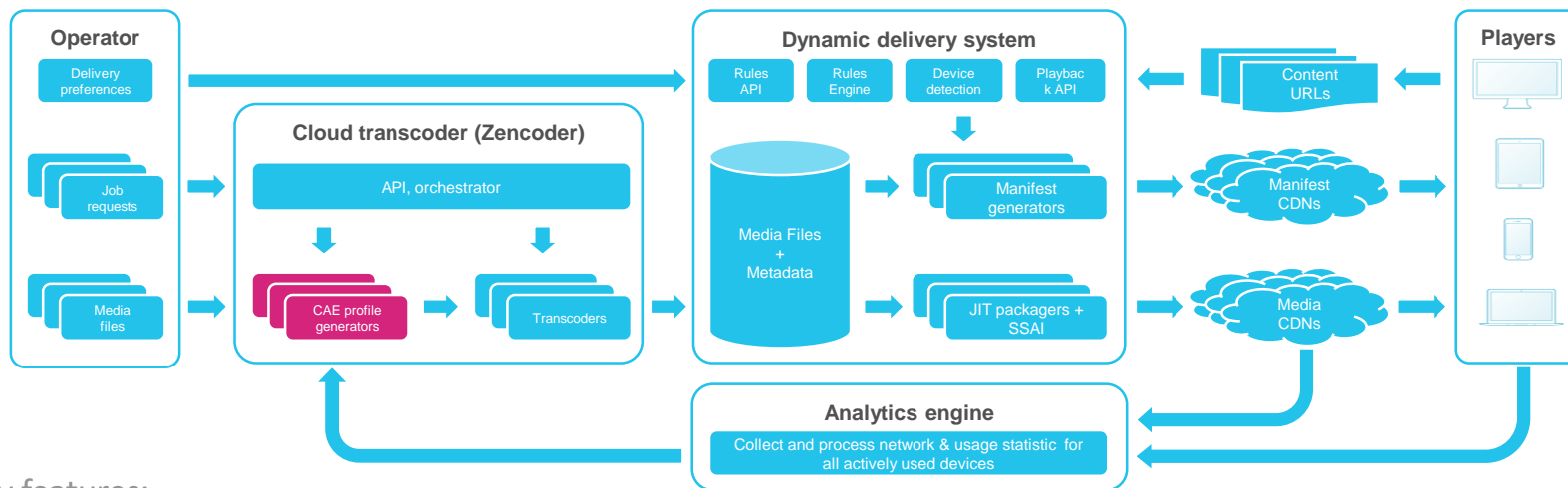
# OUTLINE

- Multi-format delivery system architecture overview
- ABR ladders
- ABR ladder generation as end-to-end system optimization tool
  - A bit of theory
    - single codec case
    - multi-codec cases
- From theory to practice
  - Brightcove CAE
  - Some recent case study results
- Conclusions



# MULTI-FORMAT DELIVERY SYSTEM ARCHITECTURE

- Top-level architecture:



- Key features:

- CAE = context-aware encoding
- Internal format is used for storage of transcoded streams
- Just-in-time transmuxing to final delivery formats
- Just-in-time, device- and operator- specific manifest generation
- Analytics engine & close loop to CAE for end-to-end optimizations



# ABR ENCODING PROFILES

- Define sets of encoding parameters for each rendition
  - Resolutions, Bitrates, Codec constraints, etc.
- Examples of existing ABE profiles:
  - Apple HLS guidelines:

HEVC/H.265	H.264/AVC	Resolution	Frame rate
145	145	416 x 234	≤ 30 fps
350	365	480 x 270	≤ 30 fps
660	730	640 x 360	≤ 30 fps
990	1100	768 x 432	≤ 30 fps
1700	2000	960 x 540	same as source
2400	3000	1280 x 720	same as source
3200	4500	same as source	same as source
4500	6000	same as source	same as source
5800	7800	same as source	same as source

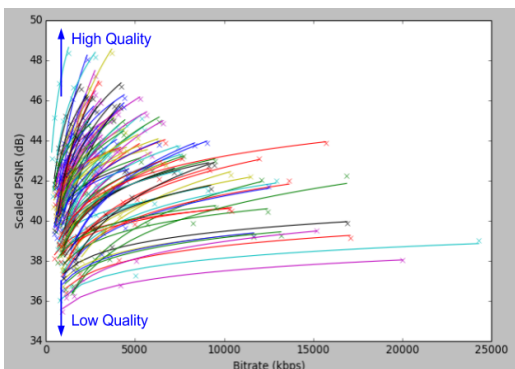
Brightcove VideoCloud (legacy static profiles):

video bitrate	decoder bitrate cap	decoder buffer size	max frame rate	width	height	h264 profile
450	771	1028	30	480	270	baseline
700	1194	1592	30	640	360	baseline
900	1494	1992	30	640	360	main
1200	1944	2592	30	960	540	main
1700	2742	3656	30	960	540	main
2500	3942	5256	30	1280	720	main
3500	5442	7256	30	1920	1080	high
3800	6192	8256	30	1920	1080	high

# WHY STATIC ABR PROFILES ARE BAD?

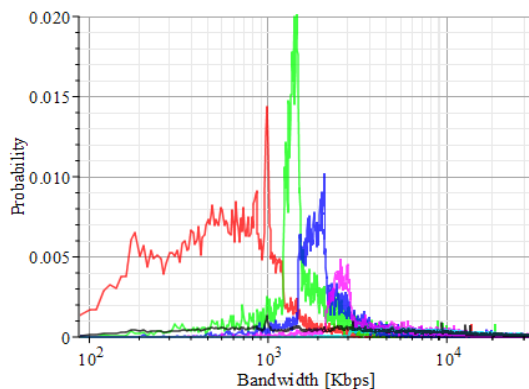
- Static encoding profiles are not accounting for:

- differences in video complexity:



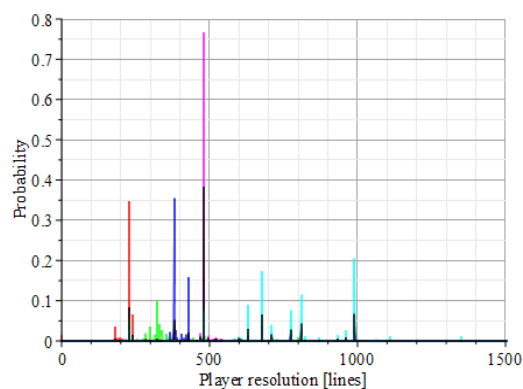
Source: Netflix, 2015

- differences in networks:



Source: Brightcove VideoCloud analytics, 2019

- differences in devices & user preferences:



Source: Brightcove VideoCloud analytics, 2019

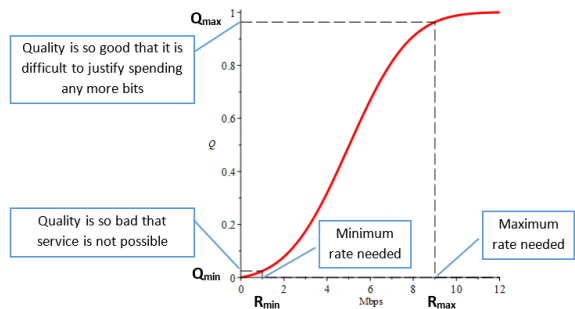
- A better approach is to design encoding profiles dynamically, accounting for characteristics of

- content → content-aware encoding (aka per-title encoding)
- network → network-aware encoding
- full context (content + network + user statistics) → **context-aware encoding**

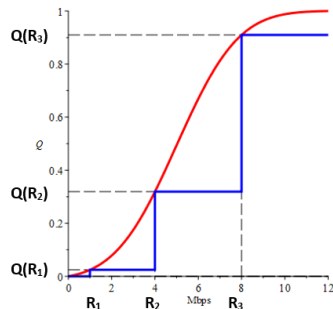


# A BIT OF THEORY – QUALITY-OPTIMAL LADDER FOR SINGLE CODEC

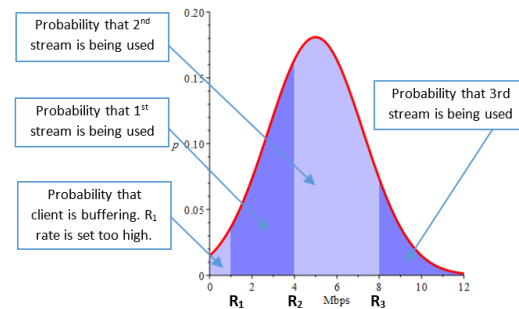
- Quality-rate function  $Q(R)$ :



- Quality delivered by streaming client:



- Probabilities of loading of each stream:



- Average quality for a given ladder of rates  $R_1, \dots, R_n$ , quality-rate function  $Q(R)$ , and network density  $p(R)$ :

$$\bar{Q}(R_1, \dots, R_n, p) = Q(R_1) \int_{R_1}^{R_2} p(R) dR + Q(R_2) \int_{R_2}^{R_3} p(R) dR + \dots + Q(R_n) \int_{R_n}^{R_{\max}} p(R) dR$$

- A quality-optimal encoding ladder is a set of rates  $\hat{R}_1, \dots, \hat{R}_n$ , such that:

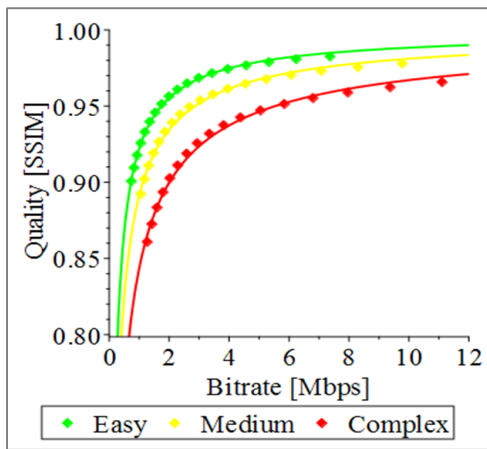
$$\bar{Q}(\hat{R}_1, \dots, \hat{R}_n, p) = \max_{\substack{R_{\min} < R_1 \leq \dots \leq R_n < R_{\max} \\ R_1 \leq R_{1,\max}}} \bar{Q}(R_1, \dots, R_n, p)$$



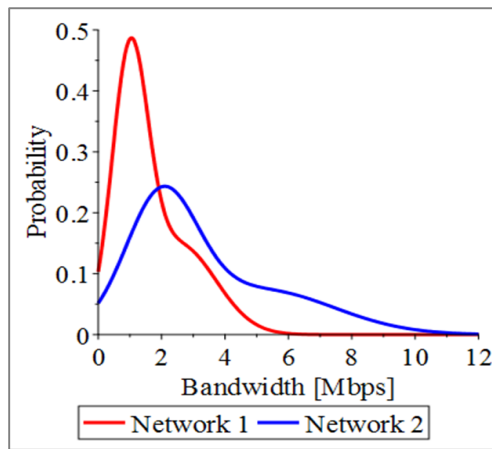
# EXAMPLE INPUTS

- Content:

Resolution=720p25  
 Codec=H.264/Main  
 Quality metric=SSIM  
 3 video sequences:  
 "Easy", "Medium",  
 "Complex"



- Networks:



Based on data from:  
 J. Karlsson, and M. Riback.  
 Initial field performance  
 measurements of LTE,  
*Ericsson review*, 3, 2008.

- Quality-rate models:

$$Q(R) = \frac{R^\beta}{\alpha^\beta + R^\beta}$$

Content	$\alpha$	$\beta$
Easy	0.0555	0.8550
Medium	0.0724	0.8016
Complex	0.1015	0.7364

- Network models:

$$p(R) = \alpha \mathcal{N}_{\mu_1, \sigma_1}(R) + (1 - \alpha) \mathcal{N}_{\mu_2, \sigma_2}(R)$$

Network	$\alpha$	$\mu_1$	$\sigma_1$	$\mu_2$	$\sigma_2$
Network 1	0.584	0.996	0.564	2.554	1.165
Network 2	0.584	1.992	1.129	5.108	2.331



# QUALITY-OPTIMAL LADDERS

Optimal ladders for Network 1:

Content	N	Ladder bitrates [kbps]	$Q_n$	$\bar{Q}$	$\xi$ [%]
Easy	2	138, 803	0.909	0.867	6.58
	3	100, 512, 1209	0.931	0.888	4.35
	4	100, 411, 866, 1645	0.946	0.897	3.34
	5	100, 349, 694, 1155, 2087	0.955	0.902	2.76
Medium	2	175, 854	0.881	0.830	7.98
	3	100, 518, 1219	0.906	0.854	5.31
	4	100, 416, 876, 1663	0.924	0.866	4.00
	5	100, 354, 701, 1165, 2104	0.936	0.873	3.25
Complex	2	234, 931	0.825	0.769	10.2
	3	145, 590, 1304	0.867	0.797	6.96
	4	102, 431, 898, 1704	0.888	0.812	5.22
	5	100, 363, 716, 1183, 2134	0.904	0.821	4.16

Optimal ladders for Network 2:

Content	N	Ladder bitrates [kbps]	$Q_n$	$\bar{Q}$	$\xi$ [%]
Easy	2	232, 1457	0.940	0.906	5.14
	3	116, 811, 2124	0.955	0.924	3.27
	4	100, 589, 1421, 2803	0.964	0.932	2.40
	5	100, 486, 1107, 1974, 3577	0.971	0.937	1.92
Medium	2	293, 1549	0.920	0.878	6.23
	3	158, 893, 2216	0.939	0.899	4.04
	4	100, 601, 1438, 2828	0.949	0.909	2.97
	5	100, 495, 1123, 1995, 3615	0.958	0.915	2.35
Complex	2	391, 1685	0.887	0.833	7.98
	3	232, 1018, 2358	0.910	0.857	5.29
	4	156, 712, 1569, 3001	0.924	0.869	3.94
	5	114, 537, 1179, 2060, 3727	0.935	0.877	3.11

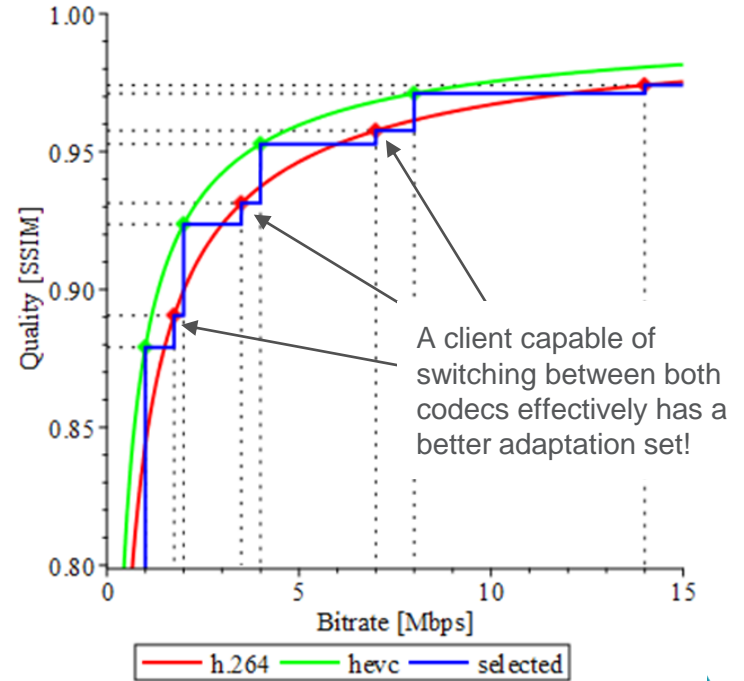
where:

- $Q_n$  = quality at top rendition [SSIM]
- $\bar{Q}$  = average quality [SSIM]
- $\xi$  = gap to average quality achievable with infinite number of renditions [%]
- Key observation:
  - **optimal profiles designed for different sources and networks are different!**



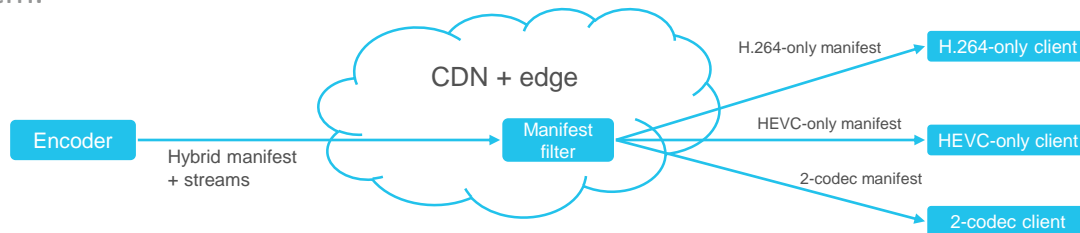
# SYSTEM WITH 2 CODECS AND 3 TYPES OF CLIENTS

- Consider a system with
  - 2 codecs, e.g. H.264 and HEVC
  - 3 types of client devices:
    - 1<sup>st</sup> can decode only first codec (H.264)
    - 2<sup>nd</sup> can decode only second codec (HEVC), and
    - 3<sup>rd</sup> can decode streams encoded using **both codecs**, and it that can also switch between such streams
- The existence of the 3<sup>rd</sup> type of client is important, as it could, in principle, achieve better quality than the other two clients:
- This of course requires special design of a mixed ladder: interleaving of rates allocated to each codec, making sure that quality-wise they form monotonically increasing sequence, that steps between renditions offer meaningful increments, etc. But this is all doable!



# DEPLOYMENT MODEL

- Assumes the use of dynamic delivery system:

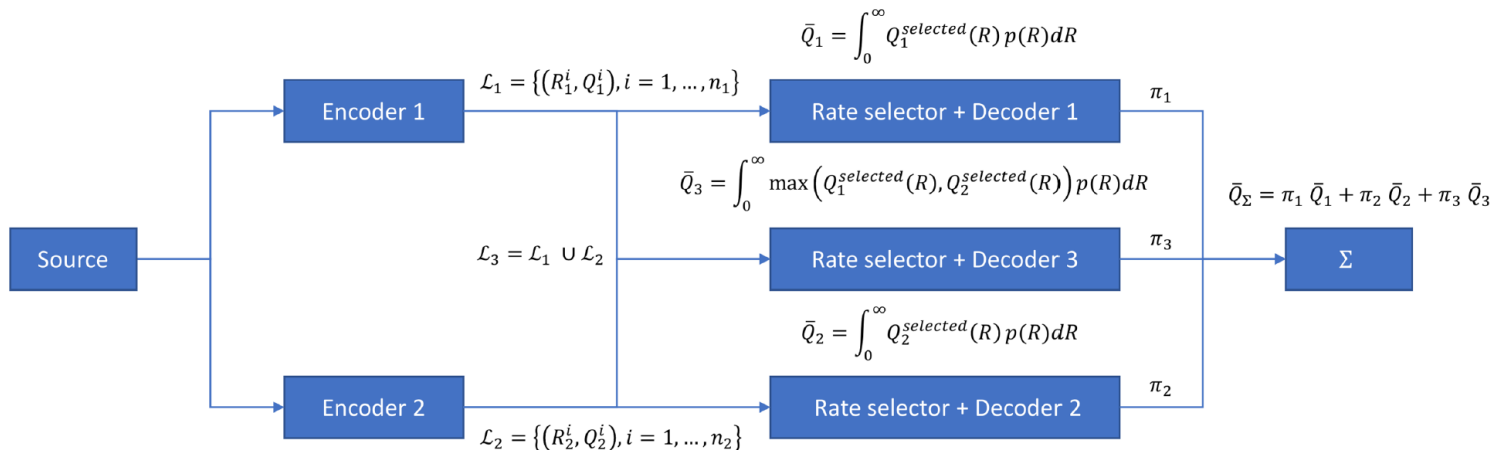


- Device detection:
  - Determines type of client requesting access to manifest
- Manifest filtering logic:
  - removes HEVC streams from manifests if requesting device is a legacy, non-HEVC capable device
  - removes H.264 streams from manifests heading to HEVC capable non-switchable devices
  - leaves both HEVC and H.264 streams if devices are capable of decoding both codecs and switching between them
- 2-codec manifests:
  - HLS: mixed variant streams (ordered by bitrate)
  - DASH: separate adaptation sets for HEVC and H.264 + supplemental properties declaring them as switchable



# QUALITY OPTIMAL 2-CODEC LADDERS

- Let's now formalize the problem of optimal design of dual-codec ladder:



- The problem: find numbers  $\hat{n}_1 + \hat{n}_2 = n$ , and ladder rates  $\hat{R}_1^1, \dots, \hat{R}_1^{\hat{n}_1}$  and  $\hat{R}_2^1, \dots, \hat{R}_2^{\hat{n}_2}$ , such that overall quality  $\bar{Q}_\Sigma$  is maximal:

$$\bar{Q}_\Sigma(p, \pi, n, \hat{R}_1^1, \dots, \hat{R}_1^{\hat{n}_1}, \hat{R}_2^1, \dots, \hat{R}_2^{\hat{n}_2}) = \max_{\substack{n_1 + n_2 = n \\ R_{\min} \leq R_1^1 \leq \dots \leq R_1^{n_1} \leq R_{\max} \\ R_{\min} \leq R_2^1 \leq \dots \leq R_2^{n_2} \leq R_{\max} \\ R_1^1, R_2^1 \leq R_{\max}^1}} \bar{Q}_\Sigma(p, \pi, n, R_1^1, \dots, R_1^{n_1}, R_2^1, \dots, R_2^{n_2})$$



# A RELATED PROBLEM

- It might be refreshing to note, that from mathematical perspective, the described problem is nothing new!
- It generally belongs to what is known in information theory as “**multiple description coding**” problem, and where specific example of a system with 2 encoders and 3 receivers was studied by L. Ozarow in 1980:

Copyright © 1980 American Telephone and Telegraph Company  
THE BELL SYSTEM TECHNICAL JOURNAL  
Vol. 59, No. 10, December 1980  
Printed in U.S.A.

**On a Source-Coding Problem with  
Two Channels and Three Receivers**

By L. OZAROW  
(Manuscript received May 15, 1980)

*This paper treats the problem of communicating a memoryless unit-variance Gaussian source to three receivers. Two channels are available, each with a separate receiver. A third receiver has the outputs of both channels available. We obtain an expression for the simultaneously achievable distortions (mean-squared error). This*

**A natural problem for the network of Fig. 1 is to characterize the set**

```
graph LR; X((X)) --> ENCODER[ENCODER]; ENCODER --> f1((f1(X))); ENCODER --> f2((f2(X))); f1 --> D1[DECODER 1]; f2 --> D2[DECODER 2]; f1 --> JD[JOINT DECODER]; f2 --> JD; D1 --> X1((X-hat1)); D2 --> X2((X-hat2)); JD --> X3((X-hat3));
```

**Fig. 1—The channel-splitting problem.**

**1910 THE BELL SYSTEM TECHNICAL JOURNAL, DECEMBER 1980**

- The practical difference, however, is that information theory is mostly concerned with finding “regions of achievability” of R/D tradeoffs, whereas in our case, we are trying to find maximum of a mix of all outputs.

# OPTIMAL 2-CODEC LADDERS

- Network 1, Easy content, 2-codec ladder,  $\pi_{hevc} = 0.4, \pi_{h264} = 0.2$ :

N	H.264 bitrates [kbps]	HEVC bitrates [kbps]	$Q_n$	$\bar{Q}_\Sigma$	$\xi_\Sigma$ [%]	$\bar{R}_\Sigma$
2	91, 719		0.97	0.9607	1.77	627.4
3	59, 403, 1222		0.9767	0.9676	1.06	929.3
4	50, 293, 773, 1736		0.9802	0.9706	0.751	1160
5	50, 242, 585, 1123, 2214		0.9824	0.9723	0.584	1331
6	91, 719	50, 286, 758, 1706	0.9843	<b>0.9733</b>	0.85	1050
7	59, 403, 1222	50, 286, 758, 1706	0.9843	<b>0.9744</b>	0.728	1152
8	59, 403, 1222	50, 237, 573, 1104, 2182	0.9861	0.9756	0.61	1249

## Observations:

- if  $N < 6$  single codec (H.264) is used
- at  $N=6$  dual-codec ladder attains same average quality as 6-point H.264 ladder, yet reducing bitrate by almost 40%
- at  $N=7$  dual codec ladder attains same quality as 8-stream H.264 ladder + 4-stream HEVC ladder constructed separately

- Optimal H.264-only ladders:

N	Ladder Bitrates [kbps]	$Q_n$	$\bar{Q}$	$\xi$ [%]	$\bar{R}$
2	91, 719	0.97	0.9607	1.77	627.5
3	59, 403, 1222	0.9767	0.9676	1.06	929.4
4	50, 293, 773, 1736	0.9802	0.9706	0.751	1160
5	50, 242, 585, 1123, 2214	0.9824	0.9723	0.584	1331
6	50, 209, 473, 850, 1421, 2568	0.9836	<b>0.9733</b>	0.482	1445
7	50, 187, 401, 692, 1087, 1687, 2843	0.9844	0.9739	0.414	1527
8	50, 170, 351, 589, 893, 1302, 1933, 3076	0.9849	<b>0.9744</b>	0.366	1590

- Optimal HEVC-only ladders:

N	Ladder Bitrates[kbps]	$Q_n$	$\bar{Q}$	$\xi$ [%]	$\bar{R}$
2	85, 695	0.9755	0.9674	1.53	611.3
3	54, 384, 1188	0.9812	<b>0.9735</b>	0.914	913
4	50, 286, 758, 1706	0.9843	<b>0.9761</b>	0.646	1151
5	50, 237, 573, 1104, 2182	0.9861	0.9775	0.504	1323
6	50, 205, 463, 835, 1399, 2537	0.9871	0.9784	0.418	1438
7	50, 183, 393, 679, 1068, 1662, 2812	0.9878	0.979	0.361	1520
8	50, 166, 343, 577, 876, 1280, 1904, 3045	0.9883	0.9794	0.32	1584



# OPTIMAL 2-CODEC LADDERS, COMPLEX CONTENT

- Network 1, Complex content, 2-codec ladder,  $\pi_{hevc} = 0.4, \pi_{h264} = 0.2$ :

N	H.264 bitrates [kbps]	HEVC bitrates [kbps]	$Q_n$	$\bar{Q}_\Sigma$	$\xi_\Sigma$ [%]	$\bar{R}_\Sigma$
2	210, 946		0.8598	0.8598	6.11	773.7
3	391	163, 860	0.8833	<b>0.8833</b>	5.94	651.6
4	391	111, 509, 1363	0.8956	0.8956	4.63	879.9
5	210, 946	111, 509, 1363	0.9072	<b>0.9072</b>	3.36	954.9
6	210, 946	85, 364, 859, 1847	0.9129	0.9129	2.77	1118
7	147, 576, 1456	85, 364, 859, 1847	0.9168	0.9168	2.34	1172
8	147, 576, 1456	69, 281, 630, 1169, 2261	0.9201	0.9201	1.99	1288

## Observations:

- if  $N < 3$  single codec (H.264) is used
- at  $N=3$  dual-codec ladder attains higher average quality as 3-point H.264 ladder, yet reducing bitrate by almost 40%
- at  $N=5$  dual-codec ladder attains quality comparable to one of 8-stream H.264 + 2-stream HEVC ladders constructed separately!

- Optimal H.264-only ladders:

N	Ladder Bitrates [kbps]	$Q_n$	$\bar{Q}$	$\xi$ [%]	$\bar{R}$
2	210, 946	0.8971	0.8598	6.11	773.8
3	147, 576, 1456	0.9182	0.8796	3.95	1043
4	114, 418, 928, 1942	0.9301	<b>0.8893</b>	2.88	1239
5	93, 327, 686, 1233, 2339	0.9369	0.8951	2.26	1375
6	79, 267, 544, 925, 1499, 2640	0.9409	0.8988	1.85	1470
7	69, 226, 451, 744, 1137, 1735, 2868	0.9436	0.9013	1.57	1540
8	61, 197, 387, 627, 930, 1338, 1967, 3099	0.946	<b>0.9032</b>	1.36	1599

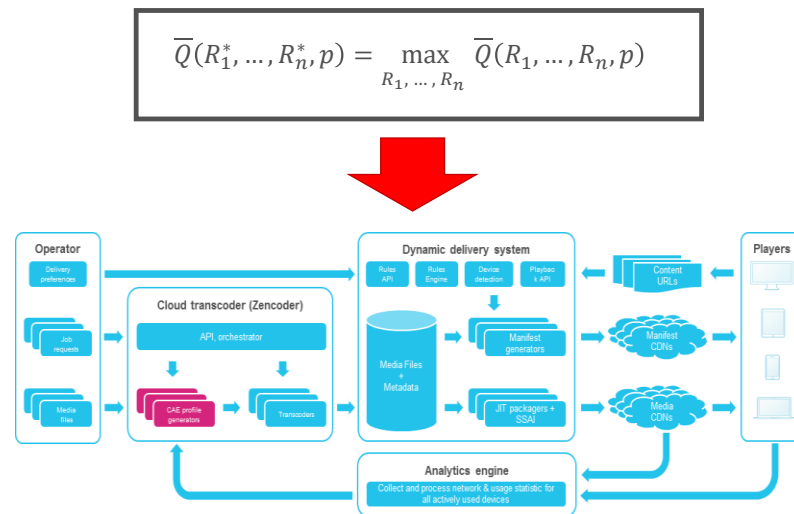
- Optimal HEVC-only ladders:

N	Ladder Bitrates[kbps]	$Q_n$	$\bar{Q}$	$\xi$ [%]	$\bar{R}$
2	163, 860	0.9292	<b>0.9044</b>	4.24	721.2
3	111, 509, 1363	0.9442	0.9191	2.68	1000
4	85, 364, 859, 1847	0.9524	0.9261	1.93	1205
5	69, 281, 630, 1169, 2261	0.9573	0.9302	1.5	1350
6	58, 228, 494, 870, 1437, 2576	0.9601	0.9328	1.22	1450
7	51, 192, 408, 697, 1087, 1682, 2830	0.9621	0.9346	1.03	1526
8	50, 174, 356, 592, 893, 1298, 1922, 3059	0.9636	0.9359	0.896	1589



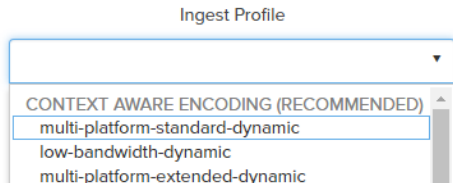
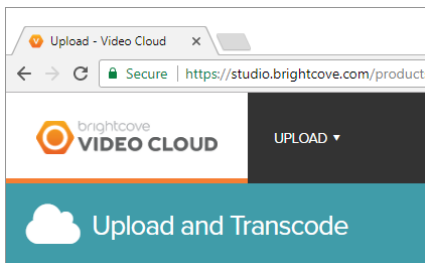
# FROM MATH TO PRACTICAL SYSTEM

- The math is relatively simple:
  - these are all typical constrained optimization problems
  - a number of existing techniques exists to solve them numerically
- However, in practice, the problem is much more complex.
- We have to consider:
  - multiple types of devices
  - multiple types of clients that may run on devices
  - multiple networks
  - multiple resolutions at which content can be encoded
  - distributions of resolutions at which content may be viewed
  - usage statistics (devices, content length, resolutions)
  - consider publisher preferences in delivery to different devices
  - factor in other QOE parameters, such as buffering probability, start time, etc.
  - factor in delivery costs (cloud storage, origin, CDN, CDN midgress, transcoding, transmuxing, etc.)
  - ...



# PRACTICAL SYSTEM

“CAE” ingest profiles in VideoCloud and Zencoder:



- In VideoCloud there are several standard CAE profiles, as shown above.
- Advanced users can also define custom CAE profiles using JSON descriptor.
- This way users may specify limits for:
  - number of renditions
  - range of bitrates to be used
  - list of allowed video resolutions, framerates, codecs, codec profiles & levels
  - allowed granularity of rate steps in the profile
  - network and usage statistics, etc.

Example custom CAE profile:

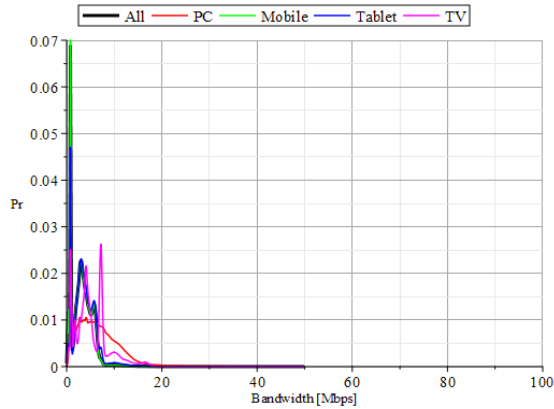
```
{
  "id": "1234567890",
  "version": 1,
  "name": "custom-cae-profile",
  "dynamic_origin": {
    "dynamic_profile_options": {
      "min_renditions": 2,
      "max_renditions": 6,
      "max_resolution": {
        "width": 1920,
        "height": 1080
      },
      "max_bitrate": 4200,
      "max_first_rendition_bitrate": 400,
      "max_frame_rate": 30,
      "keyframe_rate": 0.5,
      "codecs": ["h264", "hevc"],
      "max_granularity": 100,
      "video_configurations": [
        {"width": 1280, "height": 720},
        {"width": 960, "height": 540},
        {"width": 640, "height": 360}
      ]
    }
  }
}
```



# RESULTS – ADAPTATIONS TO OPERATORS/NETWORKS

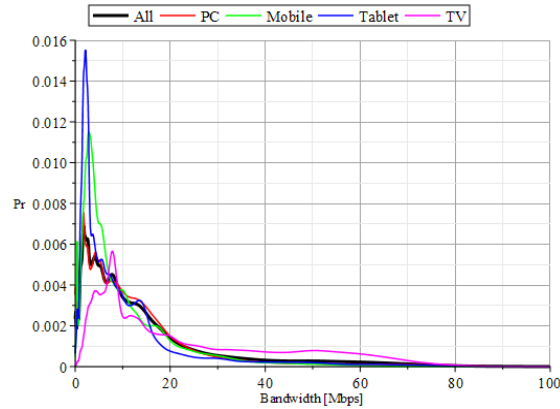
Study considering 3 operators, same content, same reference profiles:

Operator 1:



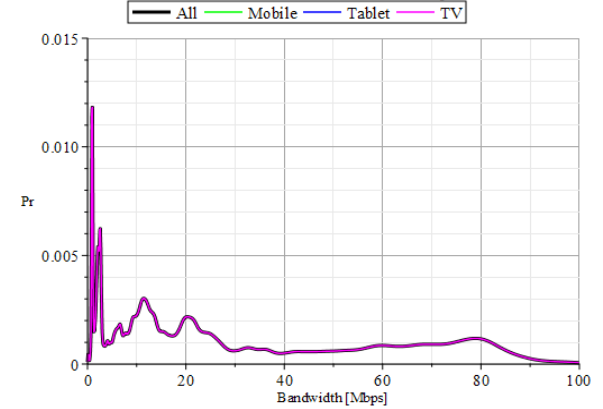
Device type	Usage [%]	Average bandwidth [Mbps]
PC	0.004	7.5654
Mobile	94.321	3.2916
Tablet	5.514	3.8922
TV	0.161	5.4374
<b>All devices</b>	<b>100</b>	<b>3.3283</b>

Operator 2:



Device type	Usage [%]	Average bandwidth [Mbps]
PC	63.49	14.720
Mobile	6.186	10.609
Tablet	9.165	12.055
TV	21.15	24.986
<b>All devices</b>	<b>100</b>	<b>16.393</b>

Operator 3:



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



# RESULTS – ADAPTATIONS TO OPERATORS/NETWORKS

Relative performance changes for each operator:

Metric	Operator 1	Operator 2	Operator 3
Renditions	-22.2%	-22.2%	-44.4%
Storage	-57.9%	-56.9%	-68.7%
Bandwidth	-8.4%	-31.3%	-33.8%
Resolution	+27.3%	+6.59%	+2.03%
SSIM	-0.9%	-0.74%	-0.68%
Buffering	-1.74%	-1.04%	-1.56%
Start time	-5.7%	-1.0%	-1.6%

## Notes:

- For operator 1, having worst networks, the savings in bandwidth are smaller, but the average delivered resolution increases by 83%
- For operators 2 and 3, the savings in bandwidth increase to 31.3 and 33.89% respectively
- For operator 3, the number of streams is further reduced, leading to significant savings in transcoding and storage costs
- In all cases optimization have also improved start up time and % of time buffering
- All savings are achieved with negligible changes in codec noise as indicated by relative SSIM change values



# RESULTS – ADAPTATIONS TO DIFFERENT TYPES OF CONTENT

## Study conducted using:

- 500 assets
- 120 hours of view time
- 34 different content categories
- Including movies, cartons, sports, etc.

## Reference profile – Apple TV:

Resolution	Bitrate	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

## Relative changes [in %] for each category of content:

Category	Streams	Storage	Bandwidth	Resolution
Action	-35.05	-77.28	-59.16	+3.57
Adventure	-29.63	-70.17	-51.33	+3.32
Comedy	-25.12	-62.16	-41.28	+2.33
Drama	-32.36	-73.29	-55.83	+3.55
Scifi	-31.38	-71.89	-53.17	+3.27
Cartoon	-30.15	-68.82	-47.71	+2.93
Video game	-29.2	-67.76	-46.17	+3.17
Baseball	-21.57	-61.09	-50.89	+0.76
Basketball	-22.1	-57.82	-34.15	+1.72
Boxing	-23.71	-65.33	-43.03	+3.1
Cricket	-14.29	-58.12	-50.13	+0.97
Cycling	-23.11	-58.92	-36.55	+2.35
Field hockey	-22.22	-51.57	-22.66	+1.1
Football	-28.57	-79.12	-52.25	+1.69
Golf	-28.57	-79.38	-74.2	+1.69
Gymnastics	-26.1	-65.45	-44.01	+2.79
Hockey	-22.22	-51.26	-20.39	+0.08

Category	Streams	Storage	Bandwidth	Resolution
Mixed sports	-23.63	-55.47	-29.22	+1.35
Racing	-28.57	-74.68	-66.96	+1.5
Running	-23.3	-56.66	-31.99	+2.52
Squash	-27.56	-67.18	-47.11	+3.22
Swimming	-22.22	-50.04	-19.67	+0.17
Tennis	-18.72	-61.04	-51.44	+1.07
Weightlifting	-31.44	-72.6	-51.66	+3.78
Documentary	-25.72	-59.85	-34.19	+2.19
Game show	-28.16	-65.18	-40.95	+3.02
Interview	-37.33	-81.17	-74.2	+1.6
Kids channel	-24.75	-59.52	-34.04	+1.69
Talk show	-36.07	-77.76	-59.02	+3.99
News	-25.97	-62.36	-39.64	+2.24
Reality TV	-24.94	-58.51	-33.52	+2.46
Sitcom	-31.49	-71.93	-54.04	+3.23
Soap opera	-34.92	-76.61	-58.83	+3.8
<b>Overall</b>	<b>-28.42</b>	<b>-65.64</b>	<b>-43.76</b>	<b>+2.65</b>



# CONCLUSIONS

- Reviewed architecture of mass-scale multi-format delivery system
- Explained tools and types of optimizations proposed to improve its efficiency, including:
  - adaptations to usage and networks statistics as specific to each operator
  - adaptations to specifics of content
- Explained mathematics behind CAE profile generation
  - including optimizations for multiple codecs and client types
  
- It all works, but there are certainly many other domains of improvements:
  - optimizations considering aspects of user behavior (e.g. preferences w.r.t. content, viewing setup, etc.),
  - optimizations considering specifics of usage of CDNs
  - optimizations considering specifics of cloud and edge computing platforms
  - etc.
  
- There is certainly more work... and more fun ahead!





**THANK YOU!**

YURIY REZNIK | [YREZNIK@BRIGHTCOVE.COM](mailto:YREZNIK@BRIGHTCOVE.COM)

