Measuring IP Video Playback Quality of Experience

A Technical Paper prepared for SCTE by

Srinath V Ramaswamy
Principal Solution Architect
Comcast
1800 Arch Street, Philadelphia, PA 19103
(215) 301 5047
srinath_ramaswamy@cable.comcast.com

Table of Contents

List of Figures

List of Tables

1. Introduction

To measure IP Video Playback Quality of Experience (QoE) there are several distinct metrics available that need to be reviewed individually to determine the overall playback quality. These may include rebuffering, startup latency, video bitrate, errors, and downshifts. This paper presents a novel approach to objectively predict the IP video playback quality as perceived by the end-user by calculating a single, final overall metric referred to as Video Viewing Quality (VVQ). VVQ factors in several distinct metrics such as rebuffering, playback errors, video quality, downshifts, startup latency, and recency. Inherent in some of these metrics are the impact from audio issues such as dropouts, missing audio. VVQ scores are currently being utilized to determine IP video playback QoE on various IP video devices and to troubleshoot playback issues. We will also discuss how this single metric, or score can further be leveraged to enhance IP Video playback experience by performing bandwidth and network optimizations. Measuring IP video playback quality is an important aspect of the transition to the delivery of video over IP from legacy QAM.

2. IP Video Architecture

At a high level, the IP video architecture is as shown in the Figure 1 below. In this figure, the compressed or uncompressed video stream is transcoded into several streams (variants), each at different fixed bitrate, to accommodate the varying network bandwidth available between IP video consumer devices and the content delivery network, and the various consumer devices. The transcoded streams might be encoded using one of the MPEG compression codecs (MPEG-2[4], MPEG-4/AVC [2], HEVC [3]). This is then packaged into one of the many streaming formats, for example MPEG-DASH [5] or HLS [6], where they are typically split into time-aligned segments of a few seconds duration. These are then placed on the Origin Server and IP CDN for delivery to IP video clients. This would apply to both linear and on-demand IP video. Consumer devices will typically try to download the highest bitrate video that they can support, with the intent of providing higher video quality to the viewer.

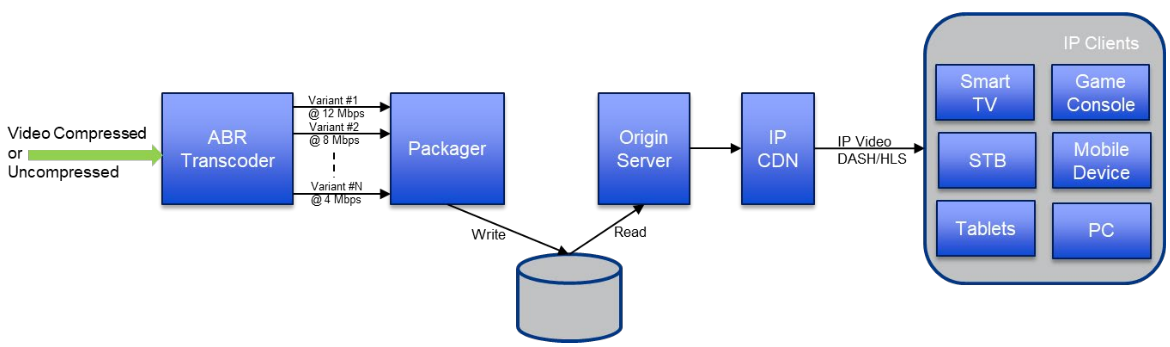

Figure 1 – IP Video Architecture

2.1. Playback Artifacts

During IP video playback several customer impacting issues could arise; some of them are listed below.

- Stalling of video playback due to rebuffering or errors in the stream
- Playback of low-quality video for extended durations

- Bad quality of encoded video/audio, causing video issues like macroblocking, blurry images, noisy, out of lip sync, flashing, video buffering errors (underflow, overflow), or missing audio
- Large startup latency
- Media playback failures (playback start fails or leads to ending playback)
- Playback control failures (for example, while pausing or exercising trick modes)
- Live latency
- Multiple and frequent occurrences of playback issues during a session

2.2. Measuring IP Video Playback QoE

Measuring IP Video playback QoE is important for numerous reasons including:

- To allow for the seamless transition to delivery of video over IP from legacy QAM; customers should not experience a degradation in video playback experience when they are switched to IP video
- Early detection of IP video infrastructure issues and enable self-healing
- Validation of new streaming technologies (for example content adaptive streaming, new players, codec)
- Facilitating IP video system component upgrades
- For the retention of video customers in a highly competitive marketplace
- To provide an entry point for identifying streaming video issues and assist in their resolution

3. Video Viewing Quality(VVQ)

As described earlier, there is a need to measure IP video playback QoE. There are several distinct metrics that can be reviewed individually to determine the overall IP video playback QoE including average video bitrate, startup latency, rebuffering duration, rebuffering count, top errors/failures, encoded video quality, and others. Viewing these distinct metrics one at a time to detect bad user experience is not feasible, especially with large-scale deployments. One option is to look at only a subset of them, but because customer experience can be impacted by any one of them, all need to be taken into consideration.

What is needed is a holistic metric for measuring IP Video Playback Quality of Experience (QoE) and the VVQ score meets that need. It objectively determines the IP video playback quality as perceived by the end-user and is expressed as a single, final overall score. VVQ would be applicable to any IP video playback use cases such as IP linear, VOD, and time-shifted streams.

3.1. VVQ Model

The VVQ model uses several metrics to compute the single final overall score. They are listed below.

- Video quality (as a result of lossy encoding)
  In this model, variants/segments in the ABR ladder are classified into three video quality categories Low, Medium, High. There are several classification options that may include:
  - Video bitrates
  - Complex video analysis
  - Video quality metric from Peak Signal to Noise Ratio (PSNR), Structural Similarity Index (SSIM) based tools
  - Quantization values
- Quality switching (Downshift to Low/Medium video quality segments)

Considered in the quality switching are peak frequency value and number of downshifts (High to Medium, Medium to Low, High to Low)

- Low/medium video segments playback
  Considered here are
  - The total playback duration of these lower quality segments during the entire measurement period
  - Continuous playback duration: duration of low/medium video playback before switching to optimal quality segment
- Rebuffering events
  - Total duration of rebuffering during the entire measurement period
  - Peak frequency value for these events
  - Continuous duration of rebuffering event
  - Number of rebuffering events
- Startup latency (Primacy)
- Last bad video quality impact (Recency)
  Elapsed time since the last bad video quality event such as low-quality video or rebuffering
- Playback failures/errors
  Considered here are failures and errors that disrupt video playback like frozen video, loss of audio, or error messages displayed to the user. During these types of errors/failures the following are determined
  - Total duration of error events during the entire measurement period
  - Peak Error frequency
  - Continuous duration of error event
  - Number of failures/error events
- End-user actions
  Latency when executing trick mode commands like Pause, Seek, Fast Forward, Rewind
- Display dimensions

3.2. VVQ Formula

The formula to calculate VVQ score is shown next. It is an equation that factors in several IP video playback artifacts. From the customer point of view the main artifacts impacting them are quality of video and audio, rebuffering, startup latency, playback failures, quality switching, and time between bad video quality events.

Video Viewing Quality Score = 100 - Impact from Low Quality Video Playback - Impact from Medium Quality Video Playback - Rebuffering impact – Impact from the time interval between bad quality events - Playback startup times impact - Impact from Playback Errors - Impact from trick play latency

Expanding this to the actual terms used in the formula we would have,

$$VideoViewingQualityScore = 100 - (LowQualVideoImpact + ContinuousLowQualityImpact + DownshiftToLowQualImpact + MediumQualVideoImpact + ContinuousMediumQualityImpact + DownshiftToMediumQualImpact + ContinuousRebufferingImpact + RebufferingFrequencyImpact + OverallRebufferingImpact + LastBadQualityEventImpact + StartupTimeImpact + PlaybackFailureImpact + ContinuousErrorImpact + ErrorPeakFrequencyImpact + OverallErrorImpact + SeekLatencyImpact)$$

The terms in the above equation are further detailed below.

1. Low-quality video impact computation: This measures the overall impact from the playback of low-quality video at certain times or during the entire measurement period.

$$LowQualVideoImpact = A1 * \frac{m1}{n1}$$

$$A1 = Low\ Quality\ Video\ Impact\ Coefficient$$

$$m1 = Total\ Duration\ of\ low\ quality\ video\ playback$$

$$n1 = Total\ Duration\ of\ video\ playback$$

2. Medium-quality video impact computation: This measures the overall impact from the playback of medium-quality video at certain times or during the entire measurement period.

$$MediumQualVideoImpact = A2 * \frac{m2}{n1}$$

$$A2 = Medium\ Quality\ Video\ Impact\ Coefficient$$

$$m2 = Total\ Duration\ of\ medium\ quality\ video\ playback$$

3. Continuous low video quality impact computation: Computes impact based on how long the user client device is playing low-quality video before it switches to either medium or optimal quality video.

$$ContinuousLowQualityImpact = \sum_{l=0}^{m3} A3 * e^{A4*t(l)}$$

$$A3 = Continuous\ Low\ Quality\ Impact\ Coefficient$$

$$A4 = Continuous\ Low\ Quality\ Impact\ Exponent\ \ Coefficient, < 1$$

$$t(l) = Low\ quality\ continuous\ duration,$$

$$l = 0\ to\ m3. m3 - Total\ number\ of\ continuous\ low\ video\ quality\ events$$

4. Continuous medium-video quality impact computation: Computes impact based on how long the user client device is playing medium-quality video before it switches to optimal or low-quality video.

$$ContinuousMediumQualityImpact = \sum_{l=0}^{m5} A5 * e^{A6*t(m)}$$

$$A5 = Continuous\ Medium\ Quality\ Impact\ Coefficient$$

$$A6 = Continuous\ Medium\ Quality\ Impact\ Exponent\ Coefficient, < 1$$

$$t(m) = Medium\ quality\ continuous\ duration$$

$$l = 0\ to\ m5. m5 - Total\ number\ of\ continuous\ medium\ video\ quality\ events$$

5. Downshift to low video quality impact computation: Computes impact when user client device is downshifting to low-quality video during playback; uses peak downshift frequency and total number of downshift events

$$DownshiftToLowQualImpact = A7 * m7 + m4 * C1$$

$$A7 = Downshift\ to\ Low\ Quality\ Impact\ Coefficient$$

$$m7 = Peak\ frequency\ of\ downshift\ to\ low\ quality\ events$$

$$m4 = Total\ number\ of\ downshift\ to\ low\ quality\ events$$

$$C1 = Impact\ to\ score\ from\ downshift\ to\ low\ quality$$

6. Downshift to medium-video quality impact computation: Computes impact when user client device is downshifting to medium-quality video during playback; uses peak downshift frequency and total number of downshift events

$$DownshiftToMediumQualImpact = A8 * m8 + m6 * C2$$

$$A8 = Downshift\ to\ Medium\ Quality\ Impact\ Coefficient$$

$$m8 = Peak\ frequency\ of\ downshift\ to\ medium\ quality\ events$$

$$m6 = Total\ number\ of\ downshift\ to\ medium\ quality\ events$$

$$C2 = Impact\ to\ score\ from\ downshift\ to\ medium\ quality$$

7. Continuous rebuffering impact computation: Computes impact based on how long the user client device is rebuffering before it restarts video playout.

$$ContinuousRebufferingImpact = \sum_{p=0}^{m9} A9 * e^{A10*t(p)}$$

$$A9 = Continuous\ Rebuffering\ Impact\ Coefficient$$

$$A10 = \ Continuous\ Rebuffering\ Impact\ Exponent\ \ Coefficient, < 1$$

$$t(p) = Continuous\ Rebuffering\ duration,$$

$$p = 0\ to\ m9. m9 - Total\ number\ of\ rebuffering\ events$$

8. Rebuffering frequency impact computation: Computes impact based on how often the user client device is rebuffering.

$$RebufferingFrequencyImpact = A11 * m15$$

$$A11 = Rebuffering\ Frequency\ Coefficient$$

$$m11 = Peak\ frequency\ of\ Rebuffering\ events$$

9. Overall rebuffering impact computation: Computes the overall impact from rebuffering during the entire video playback.

$$OverallRebufferingImpact = A12 * \frac{m12}{n1}$$

$$A12 = Overall\ \ Rebuffering\ Impact\ Coefficient$$

$$m12 = Total\ Duration\ of\ video\ rebuffering$$

$$n1 = Total\ Duration\ of\ video\ playback$$

10. Last bad video quality event impact: A logarithmic curve is used to compute the impact to user viewing experience for the time intervals between bad video quality events such as a downshift to a low-quality video, rebuffering but then recovers after a certain time. Therefore, this impact is dependent on the time between the bad video quality events.

$$LastBadQualityEventImpact = \sum_{k=0}^{m10} C1 + A14 * \log_{m11} t(k)/A15$$

$$C1 = Last\ Bad\ Quality\ Event\ Impact\ Constant$$

$$A14 = Last\ Bad\ Quality\ Event\ Impact\ Coefficient$$

$$m11 = \ Last\ Bad\ Quality\ Event\ Impact\ Base < 1$$

$$t(k) = Time\ since\ last\ bad\ video\ quality\ event,$$

$$k = 0\ to\ m10. m10 - Total\ number\ of\ bad\ video\ quality\ events$$

$$A15 = Time\ scale\ factor$$

The time of occurrences of low-quality video, rebuffering, failure events are used to determine t(k), the time since last bad video quality event.

11. Playback failure impact: This would be applicable to failures such as unable to playback video due to inability to retrieve video segments or unable to decode video or corrupt video without recovery.

$$PlaybackFailureImpact = \ A16$$

$$A16 = Playback\ Failure\ Constant$$

12. Startup time impact: Computes impact based on how long it takes for video playback to start after the user has initiated it. An exponential growth curve is used to compute the impact.

$$StartupTimeImpact = \ A20 * e^{A21*t(z)}$$

$$A20 = Startup\ time\ Impact\ Coefficient$$

$$A21 = \ Startup\ time\ Impact\ Exponent\ \ Coefficient, < 1$$

$$t(z) = Startup\ time$$

13. Seek latency impact: Computes impact based on how long it takes for video playback to start after the user has initiated a seek or trick mode operation like FFWD or RWD. An exponential growth curve is used to compute the impact.

$$SeekLatencyImpact = \sum_{p=0}^{m14} A22 * e^{A23*t(w)}$$

$$A22 = Seek\ Latency\ Impact\ Coefficient$$

$$A23 = \ Seek\ Latency\ Impact\ Exponent\ \ Coefficient, < 1$$

$$t(w) = Seek\ Latency$$

$$p = 0\ to\ m14. m14 - Total\ number\ of\ seek\ events$$

14. Continuous error impact computation: Computes impact based on how long the user client device is in error state before it restarts video playout.

$$ContinuousErrorImpact = \sum_{p=0}^{m15} A24 * e^{A25*t(p)}$$

$$A24 = Continuous\ Error\ Impact\ Coefficient$$

$$A25 = Continuous\ Error\ Impact\ Exponent\ \ Coefficient, < 1$$

$$t(p) = Error\ state\ duration,$$

$$p = 0\ to\ m15.\ m15 - Total\ number\ of\ error\ events$$

15. Error frequency impact computation: Computes impact when the player runs into error events, uses peak error frequency and total number of error events.

$$ErrorPeakFrequencyImpact = A26 * m16 + \ m17 * C3$$

$$A26 = Error\ Peak\ Frequency\ Impact\ Coefficient$$

$$m16 = Peak\ frequency\ of\ error\ events$$

$$m17 = Total\ number\ of\ error\ events$$

$$C3 = Impact\ to\ score\ from\ all\ error\ events$$

16. Overall error impact computation: Computes the overall impact from errors during the entire video playback.

$$OverallErrorImpact = A27 * \frac{m18}{n1}$$

$$A27 = Overall\ \ Error\ Impact\ Coefficient$$

$$m18 = Total\ Duration\ of\ Error\ events$$

$$n1 = Total\ Duration\ of\ video\ playback$$

3.3. VVQ Coefficients Calibration

Calibration of VVQ coefficients in the above formula involved analyzing IP video playback data from production, publicly available Mean Opinion Score (MOS) data, and technical documents specifying the customer thresholds for various IP video playback issues.

In particular, the publicly available University of Waterloo QoE database test samples with MOS scores using 25 subjects was utilized [1].

4. VVQ Implementation

VVQ scoring implementation could be done either in IP Video Players or remote monitoring system/analytics engine as shown in the Figure 2. In the case of the VVQ scoring in analytics engine, the players are expected to emit playback data such as timestamps for bitrate changes, rebuffering start and stop times, and other events, to the analytics engine to permit VVQ scoring there. In the case of VVQ scoring in the IP video players, the calculated score is communicated to the remote monitoring system. The remote monitoring system analyzes the VVQ scores and determines the measures that need to be taken to address issues resulting in low scores.

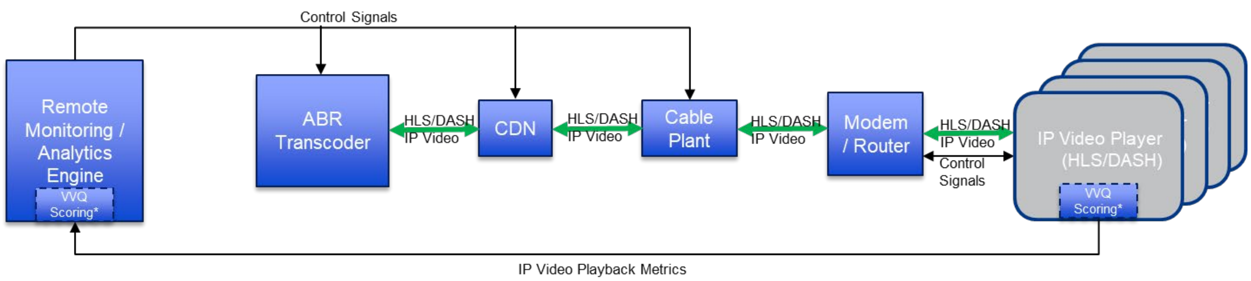

Figure 2 – Scoring in Analytics Engine or IP Video Player for Video QoE Management

4.1. VVQ in Analytics Engine Reference Architecture

Figure 3 below shows one implementation of VVQ in Analytics Engine. The IP video players in Set Top Boxes (STBs), Connected TVs, Mobile Devices and Desktops send player analytics data to Headwater 3 during IP video playback. Details on the Headwaters is available here [7]. Data in Headwaters is then stored in a Data Lake and ingested by ETL AWS Glue VVQ code to generate the scores. These scores are plotted using QuickSight user interface considering various parameters such as device type, application type, time of occurrence, and others.

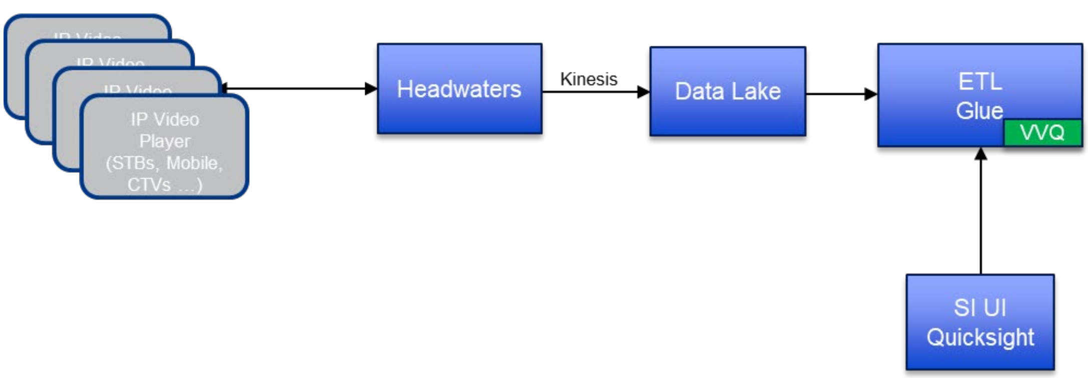

Figure 3 – VVQ Implementation in Analytics Engine

Variants/Segments in the ABR ladder are classified into three video quality categories - Low, Medium, High. The classification could be done via several approaches such as no reference or reference video encoding quality measurement tools, or simply based on video bitrates. The approach chosen here is using the video bitrates and their assignments to the three quality categories is predetermined.

The VVQ scores can computed at various points in time during video playback. Some of the options are:

- At the end of video playback sessions, for example at the end of a movie playback, end of linear channel viewing session
- At predetermined fixed intervals during VOD or linear playback session
- Continuously as, and when, new playback data is available
- At the end of a program during linear channel viewing utilizing program metadata from Gracenote

In the architecture here the VVQ is computed at the end of video playback session.

Figure 4 below shows the screenshot of the VVQ scores generated from an implementation of this reference architecture for Linear IP Video Playback sessions.

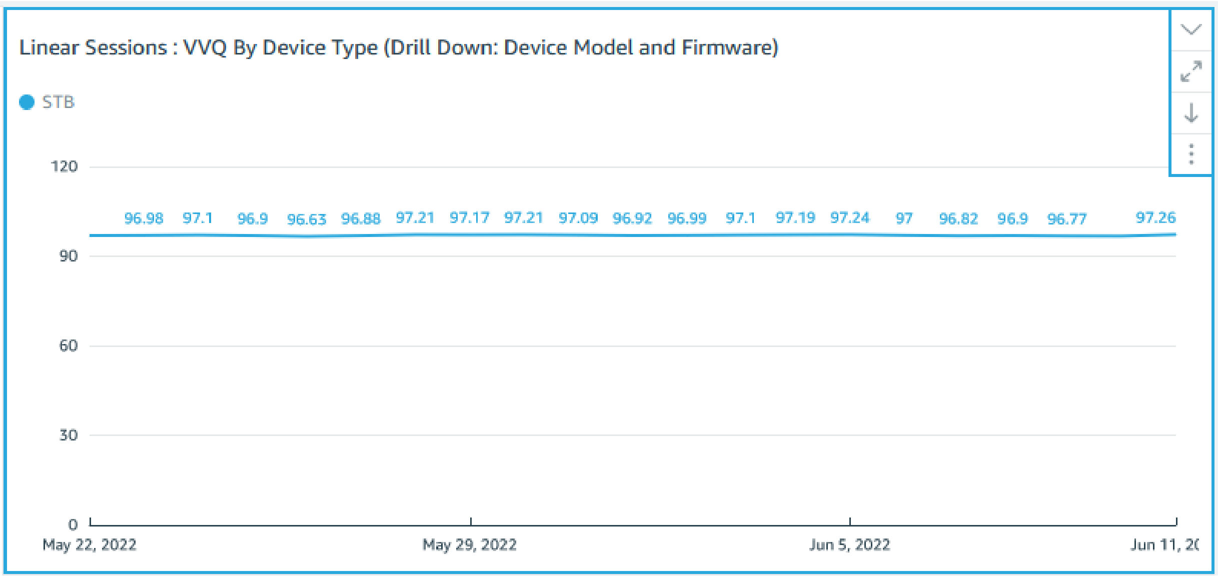

Figure 4 – VVQ Average Scores Linear

Figure 5 below displays the VVQ score components average values for Linear IP Playback sessions. The components are the various terms detailed in the section 3.2 VVQ Formula.

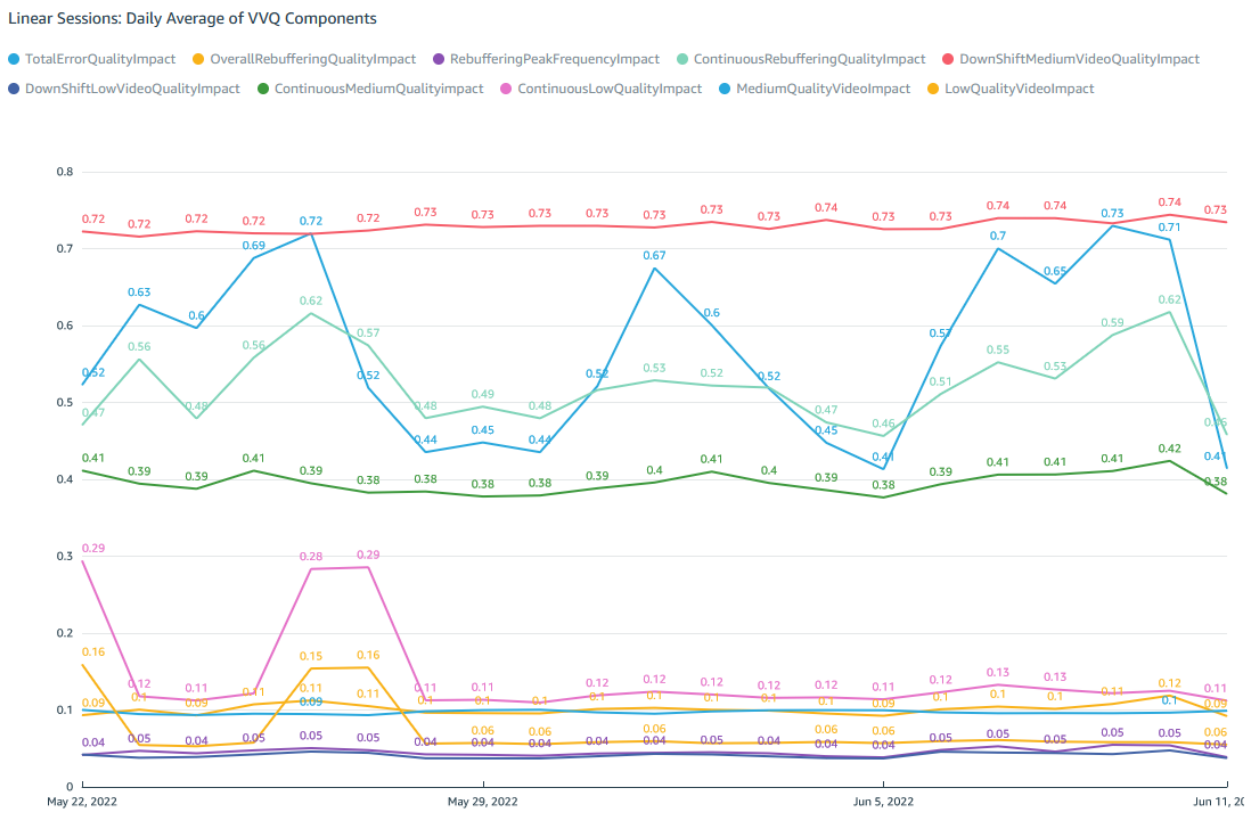

Figure 5 – VVQ Score Components Average values

4.2. Sample VVQ Scores

Listed in the table below are sample VVQ scores generated from IP video playback sessions across various devices.

Table 1 – Sample VVQ Scores

Device	VVQ Score	Description
STB	0	Total Linear playback duration: 88 mins 2 Rebuffering events, Max duration 7 mins 6 downshifts to bitrate 0.8Mbps (34s on low bitrate video*) 26 downshifts from 6 to 2Mbps (4mins on medium bitrate video*) In home issue, attributed to packet loss and Wi-Fi latency
STB	73.34	Total Recorded content playback duration: 51 mins 5 downshifts to 2.1Mbps Video (13mins on medium bitrate video*)
iPhone	0	Total VOD playback duration: 35 mins 9 mins on low bitrate video**
iPhone	82	Total VOD playback duration: 36 mins 3 Rebuffering events, Max duration 8s 2 downshifts to bitrate 0.7Mbps (31s on low bitrate video**)

Device	VVQ Score	Description
Desktop Edge	0	Total Linear playback duration: 48 mins 2 downshifts to low bitrate video (20mins on low bitrate video**) 3 downshifts to medium bitrate video (28mins on medium bitrate video**) 1 Rebuffering event, Max duration 3s
Android	20	Total Linear playback duration: 19 mins 17 downshifts to bitrate 0.5Mbps (3 mins on low bitrate video**) 56 downshifts to medium bitrate video 2Mbps (7mins on medium bitrate video**)

* Low bitrate video < 1.8Mbps, Medium bitrate video < 3Mbps, Video compression – AVC
** Low bitrate video < 0.75Mbps, Medium bitrate video < 1.5Mbps, Video compression - AVC

Please note that "We collect, store, and use all data in accordance with our privacy disclosures to users and applicable laws".

5. IP Video Playback QoE Management – Self healing

One important objective of this scoring is to drive a self-healing feature of IP video playback issues. In the Figure 2 above the remote monitoring system based on the VVQ scores calculated interacts with ABR transcoder, CDN, cable plant, and with the client to control changes that will lead to better playback video experience (higher VVQ score). The client interacts with modem/router, for example, to switch to channels with less interference.

Possible self-healing scenarios are listed below,

- If the score is low due to in home Wi-Fi issues, automatically switch channels with less interference; also recommend upsell to better routers. This is made possible when the Wi-Fi Gateway and routers network analytics data is accessible by the IP Video playback device and is also controllable by it.
- If the in-home network bandwidth is maxed out causing low scores, automatically increase bandwidth and display upsell packages for higher BW product to customer
- If unable to fix occasional network issues via configuring in-home Wi-Fi equipment, display error messages for users to troubleshoot and identify possible sources of interference
- Detect low scores with new deployments and pull back releases/upgrades without manual intervention
- Detect low scores from all clients in a plant attributable to a plant DOCSIS channels outages and/or incorrect configuration and switch to redundant paths
- Detect CDN delivery issues (e.g., network connectivity failures) and allocate additional resources on the existing CDN or switch to alternate CDNs. If the issue is with an external CDN then potentially interface with their control interface to address these issues and indicate so to the customer.
- Switch to alternate transcoder delivery lanes when consistent media failures are detected, for example on linear channels.
- If most IP video players in a region are reporting low scores due to network BW capacity saturation, the remote monitoring system could instruct the transcoder to generate the highest profile stream in the ABR ladder at a lower bitrate that would reduce the saturation
- If the score is low due to out-of-home Wi-Fi/Carrier signal issues, provide scores to carriers for alternate solutions and recommend upgrades
- Proactively inform users of upcoming IP video outages

6. Conclusion

Measuring IP Video Playback Quality of Experience is important for several reasons including transition to all IP video, early detection of IP video infrastructure issues, enable self-healing, and video customer retention. VVQ meets these needs by presenting one final overall score that holistically measures IP Video Playback QoE rather than looking at several distinct metrics individually.

The VVQ model utilizes several playback metrics such as rebuffering, playback errors, video quality, downshifts, startup latency, recency, and others to compute the score and was validated with production data and publicly available MOS scores. VVQ is flexible and conducive to implementation either in analytics engine or IP Video Player. One such reference architecture implementation and its results are shown in this paper. VVQ is actively being utilized to gauge IP video playback QoE on various IP video devices and troubleshoot playback issues.

One another key benefit is self-healing of issues in the IP video eco system utilizing VVQ scores.

Author would like to thank several Comcast Video research and engineering teams that assisted with this work.

Abbreviations

ABR	Adaptive Bitrate
AVC	Advanced Video Coding
CDN	Content Delivery Network
DASH	Dynamic Adaptive Streaming over HTTP
HEVC	High Efficiency Video Coding
HLS	HTTP Live Streaming
IP	Internet Protocol
MPEG	Motion Picture Experts Group
PSNR	Peak Signal to Noise Ratio
QAM	Quadrature Amplitude Modulation
SSIM	Structural Similarity Index
VVQ	Video Viewing Quality

Bibliography & References

[1] QoE Modeling for HTTP Adaptive Video Streaming-A Survey and Open Challenges. NABAJEET BARMAN, (Member, IEEE), AND MARIA G. MARTINI, (Senior Member, IEEE)

[2] H.264, Coding of Audio-Visual Objects — Part 10: Advanced Video Coding (AVC), ISO/IEC 14496-10:2010.

[3] H.265, High efficiency coding and media delivery in heterogeneous environments — Part 2: High efficiency video coding, ISO/IEC 23008-2:2013

[4] H.262, Generic Coding of Moving Pictures and Associated Audio Information (H.262), ISO/IEC 13818-2:2013.

[5] Dynamic Adaptive Streaming over HTTP (DASH), ISO/IEC 23009-1:2014

[6] R. P. Pantos, editor, HTTP Live Streaming, https://datatracker.ietf.org/doc/html/rfc8216

[7] Self-Service Dimensional Data Analytics, Francesco Dorigo, 2018 SCTE Cable-Tec Expo. https://www.nctatechnicalpapers.com/Paper/2018/2018-self-service-dimensional-data-analytics/download

