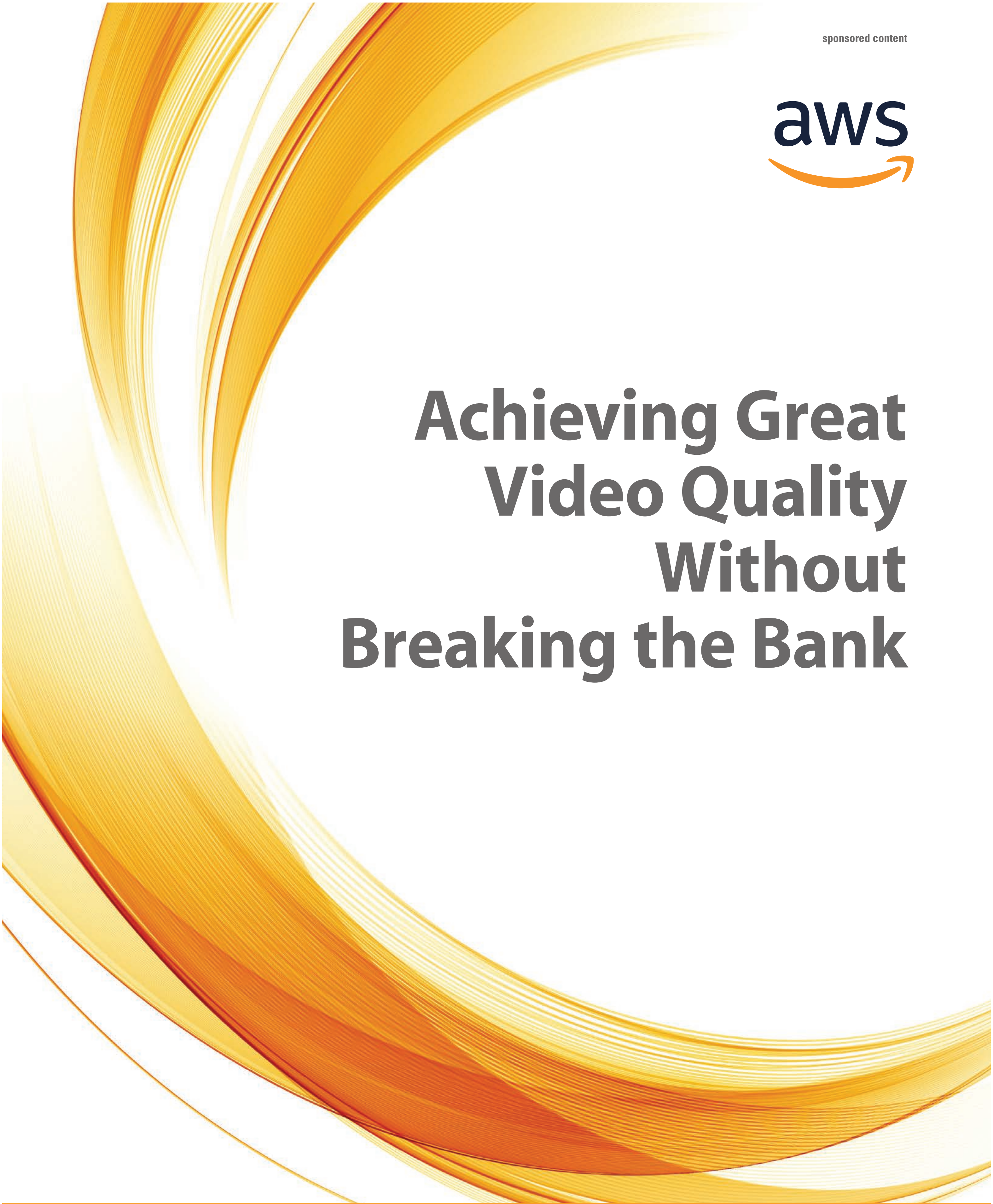


Achieving Great Video Quality Without Breaking the Bank

Achieving Great Video Quality Without Breaking the Bank

Everyone in the business of delivering video, whether over the internet or any other channel, is interested in quality. If you're going to deliver video, you want that video to look great. You also want to achieve great quality without breaking the bank, whether in CapEx, like hardware, or OpEx, like delivery costs. This e-book will examine the fundamentals of video encoding and quality, then discuss how a solution like AWS Elemental's Quality-Defined Variable Bitrate control can help achieve significant quality improvements along with bandwidth—and cost—savings, and finally, suggest two additional strategies to start maximizing your video quality today.

ENCODING VIDEO: ONE SIZE DOES NOT FIT ALL

If you analyze almost any movie or television show scene by scene, you'll notice the content has varying needs in terms of its fundamental complexity. Scenes with a lot of action and detail need a lot more bits in order to hit a quality target, whereas other scenes—say, a single actor delivering a monologue—can achieve the same quality target with a reduced number of bits.

Ideally, to maintain quality, we would want the bitrate to vary over time to maintain consistent quality regardless of the complexity of the image. **Figure 1** represents a few minutes of video, with four scenes that differ in their fundamental complexity—easy, moderate, hard, and very hard to compress. The "very hard" content might be a panning shot over a crowd, a shot of confetti falling, or simply a scene with a lot of high motion. Scenes such as these require more bits to convert all the motion and detail into a high-quality output that can be decoded and recreated accurately.

A moderate scene, perhaps a close-up of a car, or an easy scene, like a single person speaking with no camera movement, will require fewer bits to deliver the same quality target as the harder scenes. In order to most efficiently encode the entire video, we'd ideally have a rate control mode that allocates more bits to the complex scenes, and fewer bits to the easier ones.

Frequently, encoding rates are set on a title-by-title basis, a scene-by-scene basis, a frame-by-frame basis, or a macroblock-by-macroblock basis. The more granular you get with your encoding settings and rates, the greater savings you will realize. AWS Elemental's Quality-Defined Variable Bitrate (QVBR) operates scene by scene, frame by frame, and even macroblock by macroblock.

QVBR doesn't require a pre- or post- analysis pass of the entire title, and because of this can be used with both live and VOD encoding solutions from AWS Elemental. The encoder changes its quantization and bitrate in order to hit a consistent quality target, using the statistics that are already being generated during compression, in order to ensure it is achieving optimal perceptual encoding. The key takeaway is that QVBR saves bits when the quality level for a given scene is reached and uses more bits only when the content complexity requires it.

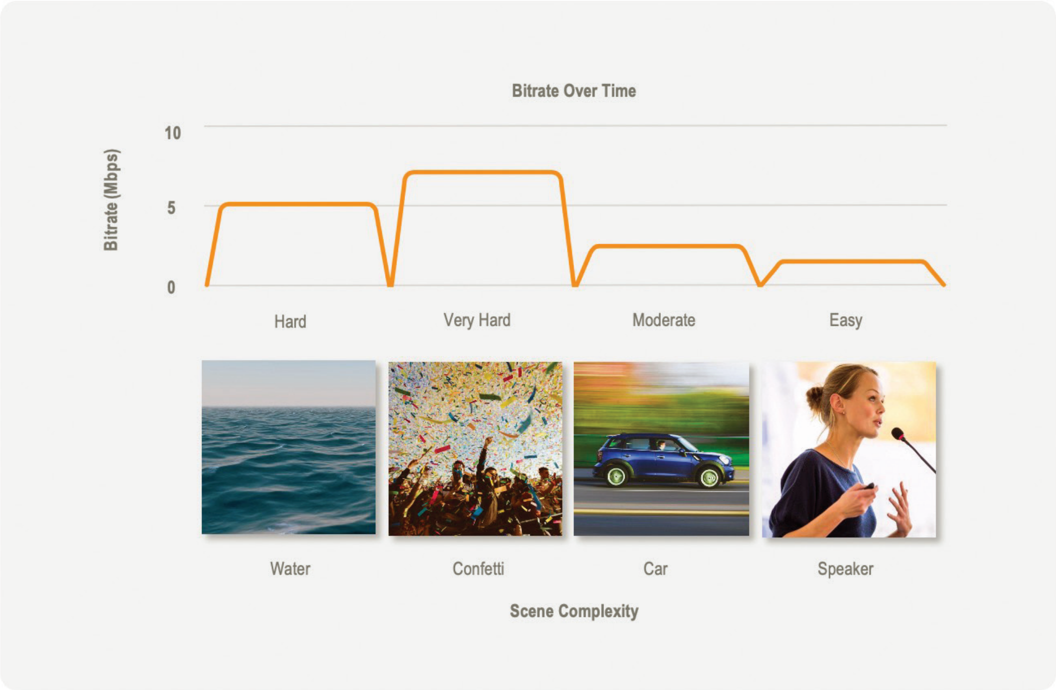


Figure 1. Analysis of scene complexity across several minutes of video

WHAT ARE BITRATE, CBR, AND VBR?

Before we look at QVBR, let's look at the basics of encoding and the types of bitrate control that have traditionally been used. Bitrate is the measurement unit of how much data is output by a video encoder or transcoder, generally measured in bits per second of video (and in today's distribution, megabits per second, or Mbps). The job of a video encoder or transcoder is to remove redundant information and hard-to-perceive picture and motion details, to shrink the size of video so it can be stored or distributed easily. Generally, higher bitrate means more perceivable picture or motion details are retained, which leads to higher video quality (VQ). Lower bitrate means fewer perceivable details are retained, resulting in lower VQ for transcoded video. Controlling the bitrate is one of the most critical operations of an encoder. The goal is video files that are small enough to be stored, and if needed carried over transmission channels cost effectively, while achieving the best possible VQ and end-viewer experience.

Constant Bitrate (CBR) is a rate control mode of an encoder that outputs a constant bitrate over time. Variable Bitrate (VBR) rate control mode outputs a variable, or changing, bitrate over time depending on the source video content (see **Figure 2**). Although uncompressed video is stored in constant data rate form (e.g., a fixed number of 10-bit pixels per second), the non-redundant information carried within it varies from video frame to video frame. Due to the variable information rate in video content, VBR gives more consistent video quality than CBR. CBR can also result in higher-than-necessary VQ and wasted output bits for some frames, and lower-than-expected VQ in others.

Even though CBR is less efficient because it results in wasted bits, it has traditionally been the most common rate control mode for live video distributed over fixed-bandwidth networks. A limited form of VBR, called statistical multiplexing (statmux), is used to carry multiple channels over fixed bandwidth quadrature amplitude modulation (QAM) and transponders. In statmux, individual video channels vary their bitrate according to each channel's content, but as a group of channels they sum up to a constant bitrate. Early HTTP delivery also used CBR as the dominant form of rate control.

For video-on-demand (VOD) storage and delivery use cases, a form of multi-pass capped VBR can be used to improve efficiency. A maximum bitrate is defined so that an encoder does not produce too much data instantaneously, and an average bitrate is set to control the overall size of the encoded asset. The encoder evaluates the content asset in multiple passes, assigning higher bitrates to scenes containing more information, and lower bitrates to scenes with less information. This leads to more efficient use of bits and more consistent VQ. Many VOD television shows and movies delivered over HTTP or downloaded to playback devices use VBR.

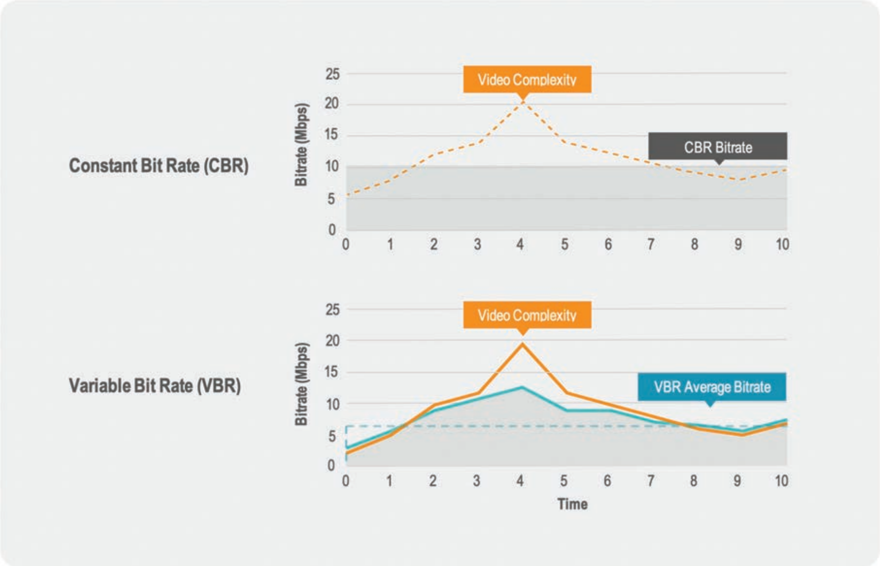


Figure 2. Constant bitrate (CBR) vs variable bitrate (VBR) encoding

The problem with both CBR and VBR is that they waste bits, fail to achieve quality targets, or both. **Figure 3** shows a graph representing CBR encoding. In this segment of video, we have low complexity content at the beginning of the clip. In the middle of the segment, we have peak complexity—a crowd pan, explosion, or confetti—before the video returns to less complex scenes. The gray area represents what we would get if we transcoded this clip using CBR rate control mode, chopping the video up into three different regions.

In the first region, the encoder is wasting bits. CBR is cranking out more bits, even though we're already past the point at which a human viewer can discern an increase in quality. In the middle section, however, we likely have some obvious video quality (VQ) artifacts, because we didn't authorize the encoder with enough bits to reach the peak complexity of that scene. Then, when the scene complexity drops again, we're still moving along at the same bitrate, wasting bits again. We could adjust the CBR level, but the results would still be unsatisfactory, and we'd have to adjust the CBR bitrate depending on each source video. We could solve our peak bitrate problem, but only at the price of wasting more bits. We could waste fewer bits, but we'd make the peak bitrate problem worse.

Traditional VBR solves some of these problems, but still comes up short (**Figure 4**). With VBR, the encoder will use as many bits as required to achieve an average bitrate, wasting bits by potentially using them even when they are not needed. Additionally, the operator needs to choose an average bitrate and peak bitrate for each asset, and optimal bit savings can only be achieved by encoding multiple times and iterating to arrive at the lowest bitrate required per video asset.

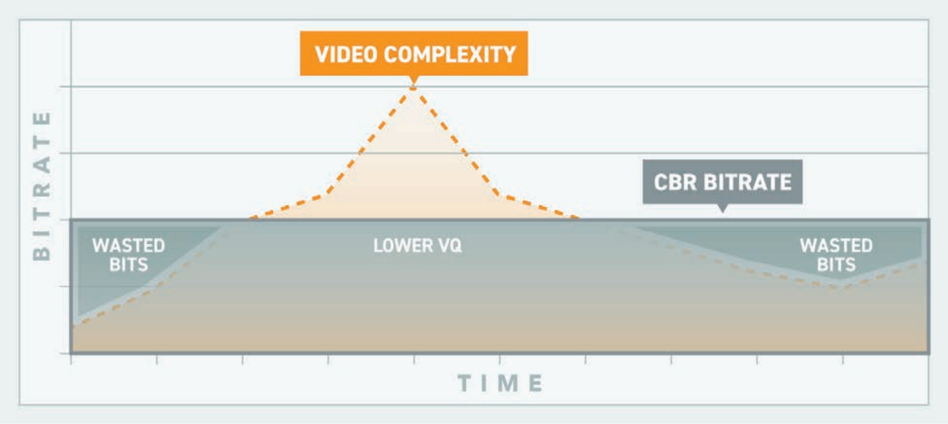


Figure 3. CBR encoding can waste bits and fail to accurately encode complex scenes.

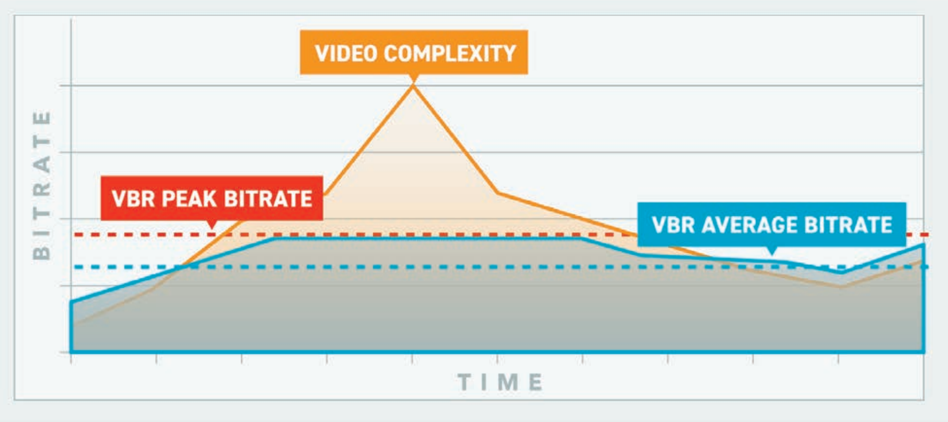


Figure 4. VBR is an improvement, but still wastes bits and requires multiple encoding passes.

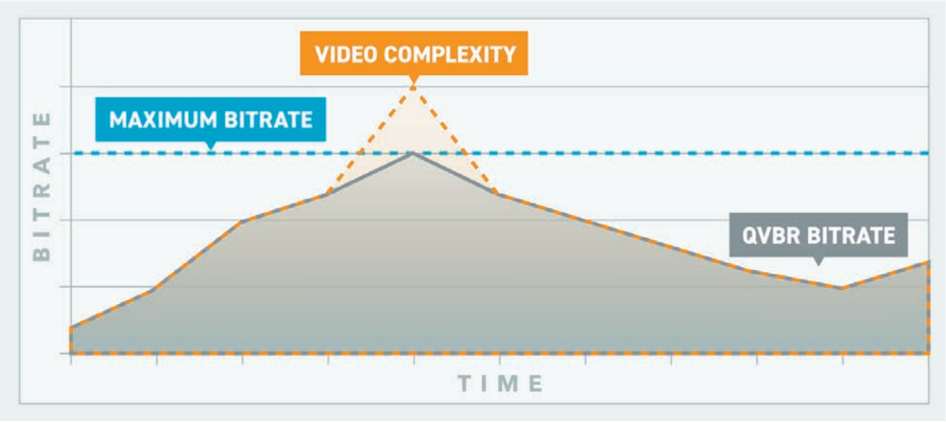


Figure 5. QVBR encoding allocates the necessary bits based on the complexity of the scene.

QVBR ENCODING

QVBR solves these problems. **Figure 5** shows the same video content encoded using a QVBR rate control mode. In the low-complexity periods of content, QVBR is authorized to use fewer bits, and in the left- and right-hand portions of the graph, the result is no wasted bits. In the middle of the content, where complexity is the highest, QVBR is able to deliver substantially more bits up to the maximum bitrate, and therefore deliver consistently high video quality.

Figure 5 also highlights the other property of QVBR rate control, which is maximum bitrate. Some video is so complex, if we allowed the bitrate to vary with no limits, we might end up allocating more bits than we want to that section of the video, which would result in a throughput rate that your delivery method or the viewing device can't support. With QVBR, you can specify not only the quality target but also a maximum bitrate, essentially a "speed limit" that keeps the algorithm from ever going higher than that maximum target. By using QVBR, you can achieve the quality you want without exceeding a throughput rate that your delivery channels or devices can't handle.

HOW DOES QVBR WORK?

QVBR analyzes content information scene by scene, frame by frame, and macroblock by macroblock during the look-ahead stage of video encoding, then allocates the bitrate to optimize quality. It does this without having to run an extra analysis pass on the content either before or after the actual encoding pass. Because there isn't any extra processing or analysis required, QVBR can be used in both file-based and live video workflows. QVBR single-pass mode enables real-time encoding for live use cases, as well as fast processing for file-based sources. Since no additional processing is required, QVBR does not incur any penalties with latency or transcode speed across the range of encoding products and services: AWS Elemental Live, AWS Elemental Server, AWS Elemental MediaLive, and AWS Elemental MediaConvert.

To enable this feature, an operator simply sets the rate control mode to QVBR and configures two parameters:

- Quality level
- Maximum bitrate

Quality level is a number ranging from one (lowest video quality) to 10 (highest video quality). Setting the quality level to 10 corresponds to a visually lossless encode of content and will result in the highest bitrate. At this level, only details imperceptible to the human eye are removed. Each level setting below 10 reduces details slightly from the previous level in a just-noticeable fashion and results in a lower bitrate.

QVBR bitrate is capped by setting a maximum bitrate constraint. Depending on the delivery method, this can be set based on network bandwidth, playback device requirements, or storage constraints. QVBR will use up to the set maximum bitrate to achieve the desired quality per the specified quality level. If the quality is met at a lower bit rate, QVBR will not use a higher bit rate, saving bits.

QVBR keeps operations simple because the quality level and maximum bitrate parameters can be set just once, and then stay the same for any type of source content, generating optimal bit savings and consistent video quality across an entire library of different source videos. With file-based encoding, QVBR also supports a multi-pass mode, which is designed for operators who have strict file size constraints to be able to improve video quality. In addition to maximum bitrate and quality level, multi-pass QVBR adds an optional third parameter, Maximum Average Bitrate. The Maximum Average Bitrate can be thought of as a maximum file size. The encoded asset file size will never exceed the Maximum Average Bitrate size regardless of content, but it can be smaller if the content is simpler to encode.

Using Maximum Average Bitrate, operators can improve video quality even if they have a strict CDN egress budget. As an example, an operator running at 3Mbps CBR can improve VQ without extra costs by switching to QVBR and setting the Maximum Bitrate to 5Mbps and the Maximum Average Bitrate to 3Mbps. The operator will not incur additional storage or CDN egress costs as the overall bitrate cap remains 3Mbps. With tough-to-encode scenes, QVBR will use up to 5Mbps to improve the viewing experience, and for easier scenes QVBR will use less than 3Mbps and save bits.

HUMAN PERCEPTION AND CONFIGURING QVBR QUALITY LEVEL

The combination of encoded picture resolution, display size, and viewing distance determines the human perception of video quality (see **Figure 6**). Encoded picture resolution (e.g., 1080p, 720p, 480p) and physical display size (e.g., 65”, 24”, 7”) determine how far apart individual encoded samples are. This is sometimes called encoded sample density. Encoded sample density, combined with the distance from the viewer’s eye to the display, creates the angle of resolution. Human eyes can resolve picture details occupying a viewing angle down to the angular resolution of about 1 arcminute.

Details that are smaller than one arcminute angular resolution are not perceivable. On the other hand, lack of details in an area larger than one arcminute decreases perceived video sharpness, reducing video quality.

Compression errors or artifacts, including blocking, noise, and blurriness resulting from lower bitrates than required by the content complexity, can also impact the viewing experience if they occupy a large enough angle of resolution. The larger the viewing angle, the more noticeable and distracting any artifacts will appear to viewers. Conversely, if any artifacts occupy a small angle of resolution, they are less noticeable and distracting.

The end result is that perceived video quality:

- *Increases* if the video resolution increases, the display gets smaller, or the viewer sits further away

- *Decreases* if the video resolution decreases, the display gets larger, or the viewer sits closer

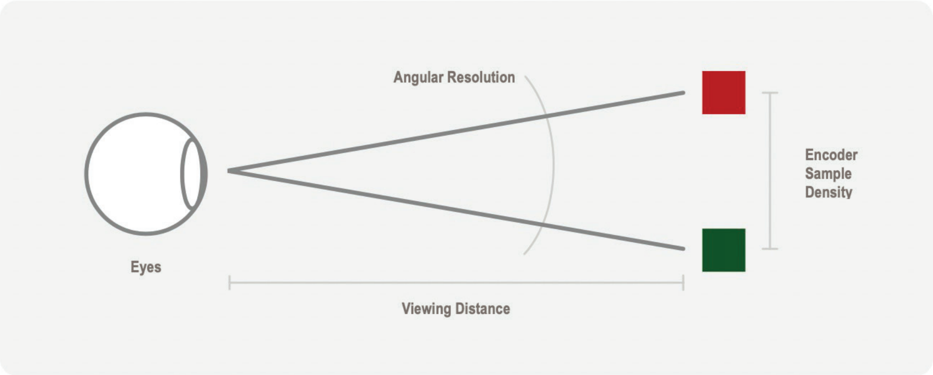


Figure 6. Angle of resolution is the combination of encoder sample density and viewing distance.

Taking advantage of human visual perception characteristics, quality level in QVBR can be set lower to save more bits when the target display is small. On the other hand, quality level should be set higher to retain more detail and reduce artifacts when the target display is large.

An operator can select the quality level according to the viewing devices and typical viewing distance of its most important audience (**Figure 7**). Because most video content is created in HD formats (1080p, 720p), the following table provides recommended QVBR quality level settings:

- Studio usage (visually lossless) - Quality level 10
- Primary screen viewing - Quality level 8 or 9
- PC or tablet viewing - Quality level 7
- Smartphone viewing - Quality level 6

OTT ABR STREAMING

QVBR works with adaptive bitrate (ABR) streaming formats. The QVBR settings for maximum bitrate and quality level can be configured independently for each stream in an adaptive bitrate stack.

Below (**Figure 8**) is an example ABR set encoded with the AVC codec for a target audience that includes primary screen, or large television display, viewers.

IMPROVING QUALITY VIA BETTER INPUTS

Along with improving the quality of your encoding output, it's important to improve the quality of your video input by pre-filtering, giving your encoder better source video. Artifacts in source video typically take two forms: blocking from source content, and mosquito noise. One occurs when the block structure of the previous encode,

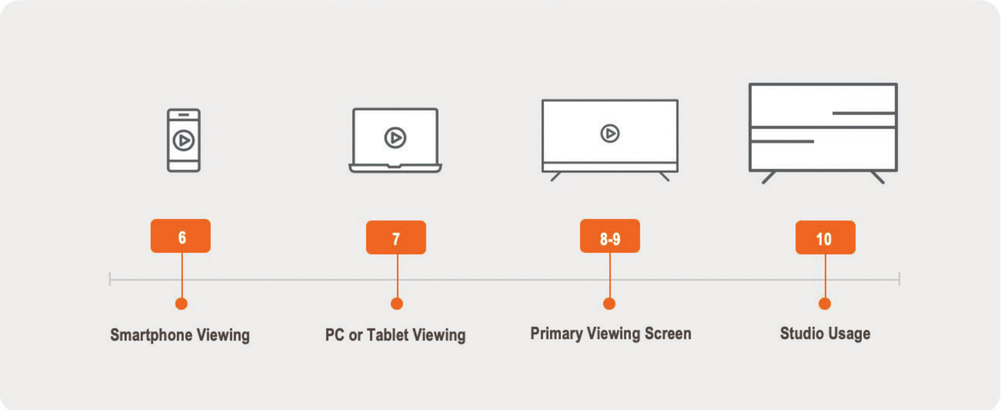
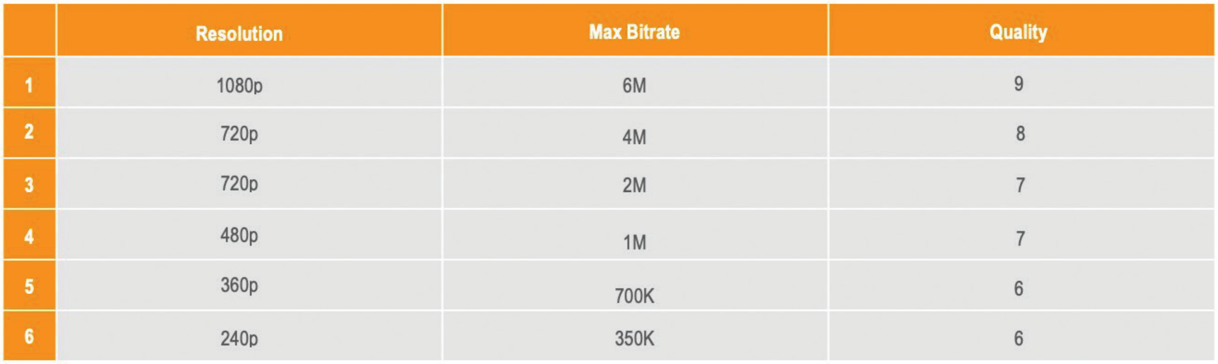


Figure 7. Operators can select different quality levels for different targeted viewing devices.



	Resolution	Max Bitrate	Quality
1	1080p	6M	9
2	720p	4M	8
3	720p	2M	7
4	480p	1M	7
5	360p	700K	6
6	240p	350K	6

Figure 8. Example QVBR quality and max bitrate settings for an ABR set

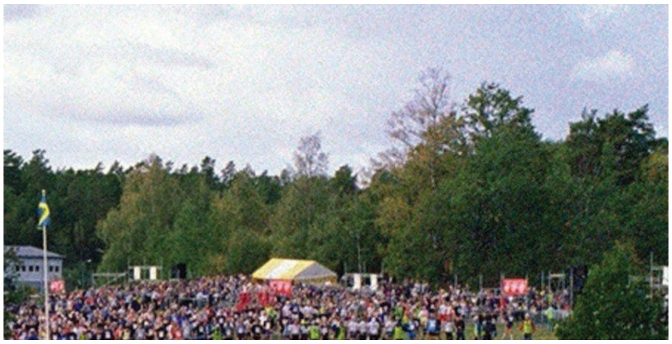


Figure 9. Unfiltered source video

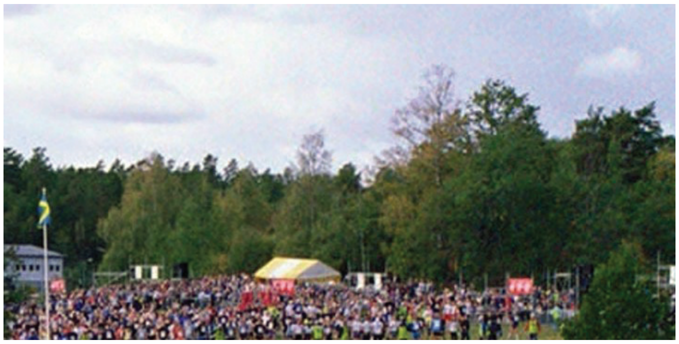


Figure 10. The same source video with the perceptual temporal filter turned on

especially for simpler codecs such as MPEG-2, is revealed in the decoded video. The average viewer sees an array of blocks rather than a continuous video image.

The other issue is called mosquito noise, which is noise from over-quantizing the video. The residual of the discrete cosine transform leaves a kind of noise that is very distracting and clusters around the edges of the frame—like a swarm of mosquitoes flying around the edge of the video.

Both the AWS Elemental Live and AWS Elemental MediaLive products provide you with the ability to activate de-blocking and de-mosquito noise filters. The goal is to remove these artificial structures from the video before it reaches the encoder. Remember that the encoder is simply encoding the actual pixel values that it gets, so it's important to filter out anything in those pixel values that shouldn't be there.

However, pre-filtering can actually go much deeper, and AWS Elemental offers a motion-compensated perceptual filter that removes even more video noise. In testing, we found that removing noise can reduce the delivered bitrate of video by up to 20% without any perceptible loss of detail. Since you are filtering out unwanted aspects of the input content, and since content complexity is variable, it's important that the filter be adjustable, and this perceptual filter does adjust on each frame. It's worth pointing out that some types of noise in the signal are actually appreciated and desirable, most notably film grain. This temporal filter is able to understand the pattern of film grain and avoid filtering it out.

Figure 9 shows a source file with a noticeable amount of visual noise, while **Figure 10** shows the same shot after the perceptual filter has been deployed. You can see that the filtering process removes noise from the clouds and cleans up the trees without losing the sharpness and dynamic colors in the crowd.

FUTURE-PROOFING YOUR ENCODING

Finally, if you're looking to keep your video quality high without breaking the bank, it's important to make sure that any encoding system you are using will be easily upgradeable without requiring additional resources, new hardware, or significant changes in your workflow. AWS Elemental continuously improves VQ across the entire suite of live and VOD encoding hardware, software, and services. In addition to QVBR and the perceptual filter, AWS Elemental has improved high dynamic range (HDR), updated the B frames algorithm, improved down conversion from 1080i, and introduced new color spaces.

By achieving better outputs with QVBR, leveraging the pre-filtering capabilities of encoders for input video, and relying on a software-based approach to all of your video encoding, you can achieve maximum video quality while keeping delivery and storage costs manageable.

QVBR BENEFITS

- Save costs: Lower transfer costs with CDN egress and lower storage costs 10%-50% or more, depending on video content
- Improve video quality (VQ) consistency: Maintain uniformity across all your content, and use a higher peak bitrate only for complex scenes
- Support the AVC and HEVC codec for even greater compression efficiency
- Easy operation: No more content-based bitrates. Instead, set a fixed video quality level and a max bitrate (according to connectivity and device requirements) and the encoder does the rest
- Same performance and no change in latency across the entire suite of encoding solutions from AWS Elemental, and no density loss with AWS Elemental appliances
- Use one-pass encoding for live video (or faster turnaround VOD encoding) or two-pass encoding for additional processing
- Included in AWS Elemental live and file-based appliances, software, and services with no additional cost

AWS
1320 SW Broadway, Suite 400
Portland, OR 97201
USA
Phone: 1 (888) 900-8104
aws.amazon.com/media