LIVE 360° VIDEO DELIVERY

Olie Baumann

MediaKind, UK

ABSTRACT

Today's consumers are increasingly seeking unique and highly immersive ways to experience their favourite live content. By delivering media to a second screen such as a mobile phone or tablet, operators can augment and supplement standard broadcast and on-demand offerings. Capture and delivery of 360º video from live events, including sports and music festivals, represents one such medium which not only offers a sense of being there but also provides a personalized viewing experience.

This paper highlights the advantages of tile-based, viewport-adaptive 360º video delivery and describes how these methods scale with the number of users and the capability of the mobile devices. It also describes experiences gained from MediaKind's deployment of an end-to-end live tile-based 360º video encode as-a-service for live sports events. This includes a 2018 proof of concept event organised and hosted by Deutsche Telekom, which saw a multi-partner collaboration enable the world's first multi-channel, viewport-adaptive, 6K 360º streaming, live from a basketball match in Germany.

INTRODUCTION

Broadcasters, telco operators and content rights holders are constantly seeking new and innovative ways to reach, attract and retain consumers. Recent advances in Virtual Reality (VR) technologies, specifically the bringing of VR experiences to mobile devices, has encouraged significant interest outside the gaming industry. Three-sixty-degree video leverages much of the technology developed for rendering VR experiences but differs in the way it is generated and delivered. Specifically, 360º video capture, delivery and rendering can be performed live, allowing operators to give consumers the sense of being at a sporting or live music event. Indeed, creative producers and the software suites which support these live not-to-be missed events are creating 360º video experiences, which in real life consumers may not have been able to get a ticket to. MediaKind, along with partner Tiledmedia have been working on proof-of-concept demonstrations of 360° video encoding and delivery for some time. Our focus has been on delivering live events to consumers watching on ubiquitous mobile devices, engaging sports and music fans in the 360° experience rather than selling sports to VR enthusiasts. As part of this we worked on delivering content to a "flat app" running on a tablet or phone i.e. not a head mounted device. Moving around the 360 is achieved by swiping or using the device gyros to show a "magic window" effect. This has proved popular and allows 360° video to be a companion to the main broadcast of an event.

Our investigations and demonstrations led to us being invited to take part in a collaborative venture, sponsored and hosted by Deutsche Telekom, which brought us together with 360° video producers Magnum Film, equipment vendor INVR.SPACE, and tiled streaming experts Tiledmedia to deliver multiple, high quality, live 360° video streams from a basketball game in Bonn, Germany in December 2018. Deutsche Telekom customers were able to view and select streams using the Magenta VR app. Streams were delivered using Akamai CDN.

This paper first discusses the challenges associated with delivering high quality 360° video to consumers; introducing the concept of viewport-adaptive 360º video delivery and how the encoding and delivery of viewport-adaptive 360º video naturally lends itself to parallelization and deployment in the public cloud: and finally, we describe our experience of covering live events using this technology.

LIVE 360° VIDEO QUALITY

As with traditional broadcast, the quality of the 360° video experience is affected by every part of the chain from the camera to the display. There are, however, additional challenges to consider such as having multiple lenses and sensors to cover the full 360° field of view, the live stitching of the image data from those sensors and the display of the delivered video on a screen which is often (but not always) mounted close to the viewers eyes. And that's before we even consider what to do with the audio! One great thing about working in this field is that the technology is developing very rapidly. At the frontline, new 360° camera rigs with broadcast quality optics and large sensors are becoming available. New Graphics Processing Unit (GPU) technology and the software which exploits it are allowing higher and higher resolution images to be stitched in real time. At the other end display resolutions, particularly those of mobile devices, are getting higher, most modern devices are now greater than High Definition (HD) resolution. In conjunction, the decode capabilities are getting better, most having hardware HEVC decoders capable of decoding 4K video.

Video Source Resolution

In production, 360° video is often represented as a sphere unwrapped into a rectangular frame, the so called Equirectangular Projection (ERP). At any time, an HD resolution display will render only a small section, known as the viewport: approximately 15%, of the entire ERP video. To ensure that the source 360º image does not limit the resolution of an HD viewport it must have a resolution of at least 8K × 4K, the equivalent of sixteen HD broadcast videos. This presents two challenges, the first being the bandwidth required to deliver such video and the second being the decode capability of the client device. To reach mass consumption, we need to be able to deliver it to mobile devices which, at the time of writing, don't generally have real-time 8K decoders on board. The bandwidth issue is compounded by the need to reduce compression artefacts which, whilst acceptable when viewed on a TV display, can be clearly seen when the video is viewed on a head-mounted device (HMD). Whilst delivery of 360° video as full ERP is possible, it limits the deliverable resolution to that which can be delivered in a reasonable bandwidth and decoded on mobile devices, currently this is 4K which is not enough to address even an HD display. The solution is for the client to only stream and decode the part of the image which is being viewed. Such methods are referred to as viewport-adaptive.

Viewport Adaptive Delivery

The term viewport-adaptive streaming covers any method in which the client device receives only a proportion of the whole 360° video. This includes “encoder-in-loop” solutions in which only the viewport is encoded and streamed to the end user, dependent on the direction in which they are looking. Not only do such methods place significant constraints on the latency of the network but they also fail to scale with the number of clients accessing the video. More scalable solutions include those standardized by ISO MPEG as the Omnidirectional MediA Format (OMAF) (3). These exploit the tile syntax of the HEVC (H.265) standard (1) to allow the full video to be encoded as independent spatial sections, tiles, which can be decoded and recombined in the decoder. The client then selectively downloads only the tiles which are required to fill the viewport, and not the entire image. This has an obvious limitation, however. When the user moves the viewport, by turning their head for example, video data will not yet be available to fill the newly revealed region. Delaying the update of the motion until data is available results in unacceptable motion-to-pixel latency, contributing to the well-known VR sickness as observed by Akiduki (2). The solution is to stream, in addition to the high-quality viewport tiles, a lower quality, low resolution backdrop. This fallback layer is continuously streamed for the entire 360º view meaning that when the user moves the viewport there is always video data available. There may be a short period where the video is lower quality until the client has the time to download and decode the high-quality tiles for the new viewport. In practice, with a reasonable bandwidth connection, the transition from low to high quality is barely perceptible with normal head motion. Specifically, we used the ClearVR implementation of the so-called “late-binding” profile now being added in OMAF version 2.

The requirement to make the video available to the client as independent tiles has advantages for encoding too. Encoding an 8K or higher resolution video in real-time is still the preserve of hardware encoders. By splitting the video up into multiple tiles, we can parallelize the process of encoding across multiple HEVC encoders. This allows the encode workflow to be hosted in cloud environments which, in turn, allows us to take advantage of the associated elasticity. A complete description of the workflow and cloud deployment is discussed in the following section.

LIVE 360° VIDEO WOKFLOW

The following description makes specific reference to two recent events for which Deutsche Telekom were trialling the delivery of viewport-adaptive 360º video streaming via their Magenta VR platform. For the first event, delivery was to a small number of selected “beta” testers. For the second event, this was extended to any customer with the Magenta VR app, an event believed to be the first 6K, viewport-adaptive, 360º video streamed live to end users.

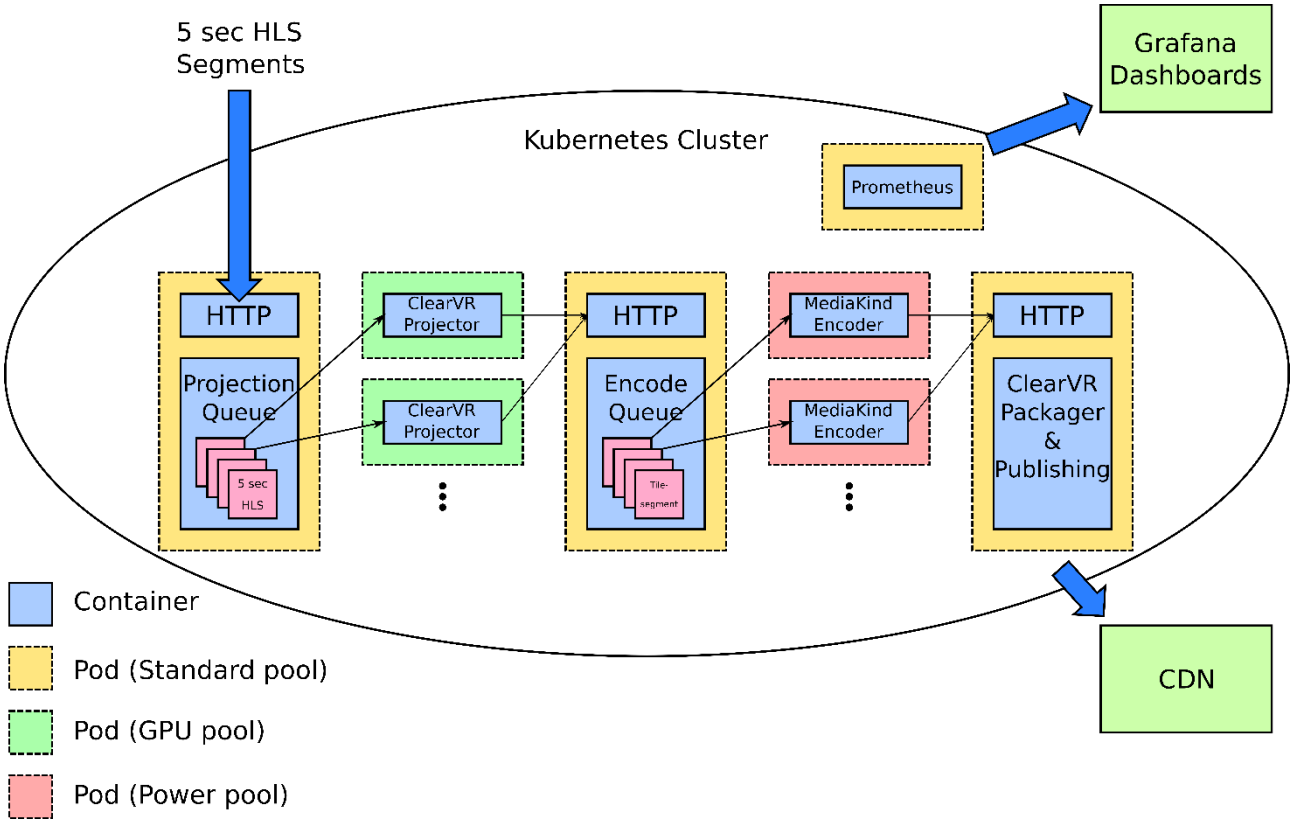

Figure 1 – Schematic of the cloud-based encoder

Production and Cloud Contribution

On-site production was managed by Magnum Film on behalf of the MagentaVR project team who, in conjunction with INVR.SPACE were testing various cameras and production methods. Two camera locations were chosen for live streaming: one on the centre line, just in front of the referees and a second just behind one of the baskets. The stream selection was left to the end user who was able to choose from the in-play menu in the client app. Modified versions of the Nokia OZO and the Z-Cam S1 were used in conjunction with the IMEVE Live software for stitching and production. The output of the production workflow was two 6K × 3K ERP videos with stereo AAC audio streams. For contribution to the cloud-based encoder, the ERP video was encoded using HEVC at 200 Mbps. The audio and video streams were multiplexed and made available as segmented transport stream files of five second duration with an HLS manifest.

The segment files were pushed into the cloud for encoding using a watcher service on the production server which, upon seeing a new segment, sent the file to the ingress node using TCP. By careful selection of the Linux kernel version we were able to use the TCP Buffer Bloat Reduction (BBR) congestion control algorithm to mitigate against packet loss / congestion throttling. Deutsche Telekom provided a dedicated 1Gbps fiber link from the Telekom Dome in Bonn to a peering point in the Deutsche Telekom network coincident with a Google Cloud point of presence. This meant that we were able to utilize in excess of 90% of the 1Gbps line speed, reliably uploading five second segments to the encoder ingress in less than 1.5 seconds despite it being hosted in the Netherlands. By using a push service

and whitelisting the source in the encoder ingress firewall we were able to securely send the segments without exposing the source server to the wider internet.

Encoding and Packaging

The encoding workflow was implemented using the Google Kubernetes Engine (GKE). This significantly simplified the deployment and orchestration process which is ideal for a service which is only required for short, well defined periods of time such as live sports or music events.

The steps involved in encoding are depicted in Figure 1. As far as possible we use a stateless microservice architecture in which each processing step is handled by a Kubernetes “pod” with an HTTP service container and a processing container. The deployment used three node pools, each having different instance types as required for the different pods. Upon ingress, each HLS segment uploaded from the event site is placed on a queue awaiting processing and encoding.

The 6K x 3K ERP video was first converted into a cubemap projection in preparation for tiling. The decode of the mezzanine video and cubemap projection can be performed in real-time using a sufficiently powerful Graphics Processing Unit (GPU). Graphics Processing Unit resources are not available in all Google datacenters; the closest Google region to the venue in Bonn with GPUs was the Netherlands but, as mentioned above, this presented no problems in terms of transferring data for live streaming. Multiple instances of the projector pod exist in the cluster all polling the ingress queue for jobs.

Once converted to a cubemap projection, the image is split into what we refer to as tile-segments according to a defined tiling scheme. In this case the fallback layer was encoded as six 512 × 512-pixel tiles, each one corresponding to a face of the cubemap. The main, high quality layer was encoded as fifty-four tiles each of 512 × 512 pixels. In fact, these high-quality tiles are encoded twice using two different Group of Pictures (GOP) structures. The job of encoding each tile-segment is placed on a queue along with the encode configuration which includes parameters such as bit rate and GOP structure. For the 6K × 3K monoscopic video a total of 114 encode jobs are placed on the queue for each five second source segment. Multiple encoder worker pods then take jobs from the encode queue and produce bitstreams according to the job configuration. Once encoded, the bitstream is posted to the packager pod for packaging and publication. By including the configuration in the job definition on the encode queue, scaling the encoder for higher resolution, frame rate or stereoscopic 360º video is simply a question of increasing the number of encoder workers. Indeed, the number of encoders can be dynamically scaled based on the encode queue length. Another advantage of having stateless parallel encodes is that redundancy, that is the overprovisioning of encode resources to handle certain types of failures, is automatically managed by Kubernetes.

Delivery and Client App

As described in the previous section, the 360º cubemap projection is effectively encoded three times, once as low-resolution fallback tiles and twice as high-quality tiles. This means that the volume of data being packaged and published to the origin server is relatively high. No single client ever receives all this data, but it must of course be made available on the Content Delivery Network (CDN). In these trials the Akamai CDN was used with some specific optimizations for tiled delivery. We were able to load balance the delivery of the content across multiple entry points which enabled the high volume to be managed without

loss. Additionally, some care was taken to ensure that the manifest file was not delivered to the edge of the network in advance of the tile-segments to which it referred, as that might result in the client trying to retrieve data that is not yet available. To enable playback of tiled streams in Deutsche Telekom's existing 360º video platform, the Tiledmedia ClearVR SDK was integrated in the Magenta VR client. Updates were made available to users via the Oculus store.

Monitoring and Administration

Monitoring microservices deployments for live video presents a significant challenge. The objective of any monitoring system is to pre-empt any resource or timing issues which may affect the user experience. For this we use the Prometheus database and Grafana dashboard tools to collect and present data from the entire encoder workflow, from ingress to packaging.

Each process in the workflow registers itself with the Prometheus service as an exporter of several data specific to its function. For example, the encode queue exposes the current queue length which is collected and stored as a time series in the Prometheus database. We also collect and store the times taken for a source segment to propagate through the system. Several Grafana dashboards were defined for the presentation of the time series data. These dashboards range from a high-level overview to detailed data about a specific service or processing function and allow engineers to identify issues and manage the cluster during the live encode.

DEPLOYMENT SPECIFICS

In order to test the entire end-to-end chain prior to the event we arranged for a standard desktop PC to be installed at the stadium and connected to the internet via the dedicated 1 Gbps fibre. We were then able to test the connection to the encoder ingest over the week before the event using standard Linux tools such as iperf. We were also able to host test mezzanine content as HLS streams on this machine and use it as a stand-in for the production server. Since the 360º encoding and packaging solution is hosted in the cloud it could be allocated in advance and tested for resiliency using this content without any representatives on-site. For both events, on-site support in Bonn to oversee the upload service on the production server was provided, but the encode workflow was monitored and administered remotely by an engineer in the UK.

The request from Deutsche Telekom was that the delivery of the 360º streams to their end user should be on average 12 Mbps and should not exceed 15Mbps. This corresponds to the bandwidth available to most Deutsche Telekom subscribers at this time. The nature of viewport-adaptive streaming is such that the delivered bitrate is a function not only of the encoding bitrate, defined as bitrate per tile, but also a function of the viewport motion. When the viewport remains fixed (i.e. no head motion in the case of the HMD) the bitrate was observed to be below 10 Mbps. With significant, and possibly unrealistic, viewport motion the bitrate would occasionally reach 15 Mbps. Knowing that some subscribers will have significantly higher available bandwidth than this, it would be ideal to have multiple high-quality profiles available and a rate adaptation algorithm in the client to dynamically select the most appropriate based on network conditions. This is planned for future events.

Feedback regarding the quality of the streams was generally very positive. A basketball game was, in many respects, an ideal event for coverage in 360º due to the relatively small court, large ball and the proximity of the cameras to the court. That said, at 30 fps the motion of the ball, particularly when close to the camera, was visibly stuttered, in some cases making it hard to track. The need for higher frame rates would be even greater for events such as ice hockey where the puck is small and moves extremely quickly. The need to increase resolution above 6K was known before the event, but currently the limitations on frame rate and resolution are imposed by the cameras and production software. We are aware that new GPU hardware and software updates will permit the live production of 6K × 6K stereoscopic captures at 60 fps in 2019. The scalable nature of the encode workflow means that it will be ready to encode and deliver these data rates as soon as they are made available.

With so many factors contributing to delay in the acquisition, encode and delivery, it was not possible to commit to a glass-to-glass latency ahead of the event. On the day, we made several "clapper-board" measurements and found that it was in the range for thirty to fourty-five seconds. Whilst this is relatively long compared to linear broadcast, there are clear routes to reducing it significantly. Such routes include simply reducing the HLS segment length output from production and implementing the low latency Common Media Application Format (CMAF) tools in the packager. This remains an active area of investigation.

For the second, public, event users of the service spent on average 27 minutes watching 360° video, divided across multiple sessions during the game. Sessions, or periods of continuous watching, were on average 2.9 minutes for the centre line camera and 2.18 minutes for the basket camera. Of the total number of hours streamed, 44% were to the flat version of app which allowed users to view the main broadcast simultaneously, 55% being consumed on head-mounted device.

CONCLUSIONS

We have demonstrated, as part of a collaborative proof-of-concept project, the ability to stream viewport-adaptive 360º video content live to the consumer. The use of a tile-based streaming solution has many advantages. Not only does it meet the bandwidth and decoder requirements necessary to allow consumption on mobile devices, but it also lends itself to cost effective encoding on cloud platforms. This, in turn, means that the encode can be elastically scaled to handle any resolution, frame rate and number of channels which will surely be necessary to keep up with advances in display and head mounted device technology. Importantly, this proof-of-concept demonstration offered DT's MagentaVR customers a significant improvement in their quality of experience and paves the way for a new standard of 360° delivery on the platform.

REFERENCES

1. International Telecommunication Union, "Recommendation ITU-T H.265", https://www.itu.int/rec/T-REC-H.265-201802-I/en, retrieved 17th January 2019

2. Akiduki, H, "Visual-vestibular conflict induced by virtual reality in humans," *Neuroscience Letters,* Vol. 340, No. 3, 17 April 2003, pp. 197-200

3. The Moving Picture Experts Group, “Omnidirectional Media Format”, https://mpeg.chiariglione.org/standards/mpeg-i/omnidirectional-media-format, retrieved 17th January 2019