Whitepaper

Constrained Connectivity

This Dell Whitepaper addresses the concept of constrained environments created over existing connectivity interfaces due to concurrent workloads in an evolving edge computing architecture paradigm, and the new set of problems to solve in these environments.

October 2019

Author(s)

Michael Gatson

Vivek Viswanathan

Table of contents

Introduction 3
Forward and Back Channel 3
Bandwidth Requirements 4
Latency Model 4
Synchronization 6
Conclusion: Solution Space & Areas of Focus 7

Introduction

Substantial changes will take place in information and communications technology as we migrate to the next generation compute platform. This platform will likely include elements of several adjacent technologies. These adjacent technologies include 5G Networks, Edge Cloud Computing, IoT, AI and Machine Learning, and Augmented and Virtual Reality (AR/VR).

As these future technologies mature, new and additional workload requirements will be placed on the information and technology platform to support these new areas of focus.

In an evolving edge compute architecture paradigm, increasing concurrency of workloads creates constrained environments across existing standard connectivity interfaces, while the requirement to increase performance and reduce latency will increase. These needs are further exacerbated by the need for backward compatibility with existing interfaces and conformance to current standards.

Forward and Back Channel

Traditional content consumption models, whether broadcast/multicast or streaming, are largely unidirectional: the primary transmission originates at a host, or sources, and terminates at an end-point sink, or sinks. The direction of this primary transmission from source to sink is typically referred to as the forward channel. An example of forward channel over a network would be streaming of YouTube content from a YouTube https server (source) to an edge client browser or app (sink). Example of a local forward channel are the video, audio, and control output data from an IHS tethered wired or wirelessly with an XR (AR or VR) HMD.

In addition to the forward channel, real-time interactive content experiences generally require the sink to respond with control, data or other ancillary information. The direction of transmission from a sink back to a source is typically referred to as a back channel.

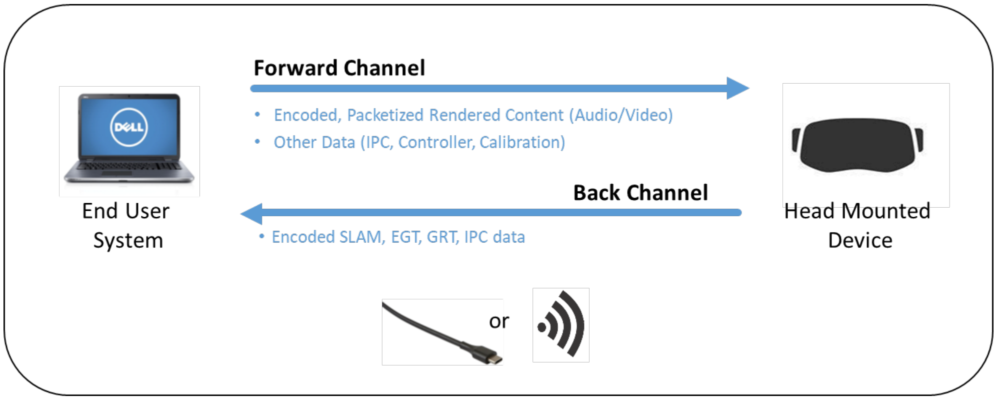
Forward channel transmissions, by themselves, face constraints. In general, the desire to maximize the content transmitted leads to pushing the bandwidth limitation of the communications channel. Numerous techniques are used, including compression, resiliency and error concealment, to maximize amount of content transmitted within transmission bandwidth limitations of the forward channel, while eliminating or minimizing loss or error in environments where transmission bandwidth is constrained or faulty.

For traditional transmissions, the backchannel requirements are usually less than forward channel, in terms of overall bandwidth. However, emerging user experiences, such as Augmented or Virtual Reality, place much greater demand on the backchannel, both in terms of amount of data transmitted and synchronized, to a greater need to reduce latency, or increase responsiveness, of the transmission. The forward and back channel communication, in fact, combine to define an overall requirement for round-trip latency, for many use cases.

A straightforward example of this requirement is an AR or VR head mounted device (HMD) for use with a host or end user system. The forward channel transmission is comprised of audio, video and control information for presenting a virtual image to the headset. This image presented, that is, what the user sees, is based on the user's most recent position and direction within the defined virtual space. As the user changes head or body position, there is a need to update this positional information to the host, or through the back channel, so that future video frames present an updated virtual image based on the user's most recent position. The round-trip latency between updating positional changes from the user to receiving the correctly updated image must be sufficiently low to eliminate motion sickness. This limit is empirically established as roughly 20 milliseconds.[1]

[1] Jerald, J. (2009). *Scene-Motion- and Latency-Perception Thresholds for Head-Mounted Displays*. Department of Computer Science, University of North Carolina at Chapel Hill.

Figure 1: AR example of forward and backchannel transmission.



Bandwidth Requirements

The overall bandwidth requirements for forward channel and back channel are steadily increasing. For the example use case above, for a simple, single wire connection between headset and end user system, the bandwidth requirements for a headset running approximately 2k per eye resolution at 90 frames per second (fps) is over 30 Gbps. This is equivalent to using all four lanes of Display 1.4 alt-mode overall a USB-C connection. In the near future, as screen resolution increases at the same frame rate, the forward channel transmission bandwidth would exceed this limit.

Likewise, if this same headset supported high definition video capture, possible with depth capture, the backchannel requirements could exceed 10 Gbps, equivalent to USB 3.2, independent of supporting tracking and other control data.

Latency Model

The example above represents an extreme case of round-trip latency limits, where a hard, physical limit must be met or an immediate adverse effect, i.e. motion sickness, occurs. Additional techniques, such as predictive rendering, are used in addition to compression and transmission optimizations to mitigate this constraint, but the requirement must be met, even as bit rate requirements increase driven by the need for higher resolution imaging or image sizes increasing to support a wider user field of view.

Extending the example to broader use cases, not all data or information must conform to this extreme limit. In fact, various levels of processing and latency requirements may exist within the same ecosystem.

For example, the closed loop processing required of 20ms—ultra low latency, or ULL-- as discussed for AR/VR and other interactive devices, may need to be maintained between the headset and the end user system, or alternatively, may be offloaded through acceleration within the headset.

Other services, however, may need real time service but at slightly higher latency. For example, even if tracking latency for a headset is handled through offload, the need for a constantly updated graphics stream in addition to other services, would likely require a low latency (LL) path—greater than 20 milliseconds, but likely less than 50 milliseconds between local devices.

In addition, the end user system may have access to edge services, such as machine learning algorithms to support object recognition or other services, that support tasks at the end user, but are offloaded from the end user system. These services may be required to support an intermediate or mid-range latency (ML), possibly greater than 50 milliseconds, but still less than 200ms, given they may be related to specific user interactions, but may be allowed to update in the background.

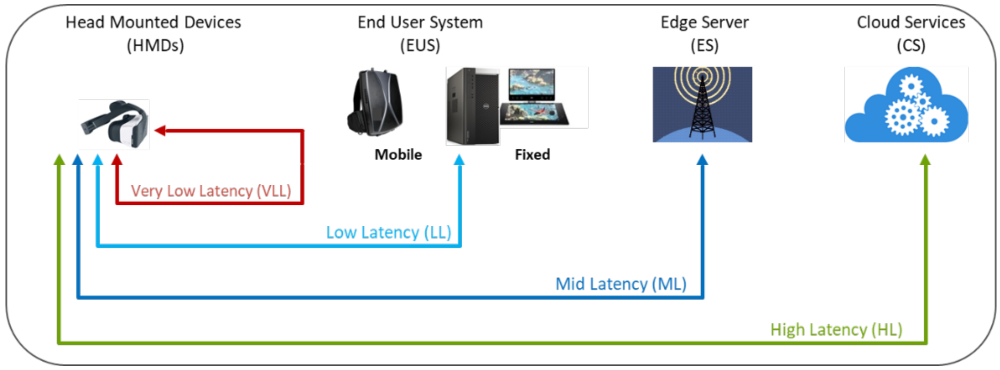
Finally, the user may have access to cloud services that are completely offloaded from the end user system and may tolerate higher latencies (HL) of greater than 200ms.

Backchannel requirements may also support this model, given the amount of video capture, motion-tracking, context and control information backchannel to various levels of compute need to support the latency requirements at each level of processing. E.g. the closed loop ultra-low latency tracking discussed above can be handled locally within the headset through offload or between the end user system and the head mounted device.

Other control and video capture information that is streamed real time back to the end user system may withstand a slightly high latency in support of the 20-50 milliseconds low latency round trip responses. Detailed mapping including depth mapping or upload of video capture in support of edge services such as detail SLAM maps or video archiving may be supported at midrange latencies to the edge. Finally, larger datasets uploaded in support of cloud services may not be sensitive to latency and can be supported by higher roundtrip latencies greater than 200 milliseconds.

The hierarchy described in the examples above is represented below as a multi-prong latency model.

Figure 2: Multi-prong latency model for AR Use Case Example



While the multi-prong model described above accommodates different round-trip latency requirements for different use cases and their resulting workloads, the I/O requirements introduced with services enabled by Scalable Edge Cloud Compute further add to the constrained connectivity problem. The increasing need to share context or transfer context to back-end processes in support of having richer compute capability, coupled with the common desire to reduce compute hardware at the end user device, increases bandwidth demands while straining the ability to maintain low latency paths at every level in the hierarchy.

Therefore, the integration of the new emerging technologies and related, increases requirements for both bandwidth and latency, while in turn enabling the offloading compute needs at the end user.

Synchronization

A related requirement for real time streaming applications requiring higher bandwidth and lower latency in a constrained environment, is the need to synchronize the delivery of specific real time services within and between the various transmission channels.

A few examples of these synchronization requirements and issues:

- Forward Channel transmissions (FC-FC): Audio-video-controller (AVC) synchronization is required for normal video streaming, video conferencing and interactive use cases such as gaming and AR/VR. In cases with constrained or degraded connectivity, data may be lost, delayed or received out of order at the end user, and need to be reconstructed at the receiver.
- Forward Channel – Back Channel synchronization (FC-BC): In some instances, forward channel information, such as IPC control information may need to be sent, with the need to maintain synchronization with position and motion tracking information being transmitted through the back channel.
- Back Channel to Forward Channel (BC-FC): In some cases, such as ultra-low latency closed loop processing, back channel information (e.g. motion tracking in AR/VR or gaming environment) must stay in tight loop synchronization with forward channel frame data. Any delay in transmission may cause will cause dropped frames or incorrect post-processing of frame data.

- Back Channel transmissions (BC-BC): Similar to forward channel transmissions, back channel information such as video capture and motion tracking, may require tight synchronization to other context and control information, such as other sensor data or audio.

Conclusion: Solution Space & Areas of Focus

As discussed, emerging technologies will create new demands for bandwidth and lower latency applications. The emergence of edge and cloud services will enable compute offload for the end user, but with the cost of further increasing requirements placed on forward and backchannel transmissions, including the need to synchronize specific services, resulting in constrained connectivity at various levels of the compute hierarchy.

To improve performance within these constraints, particularly while adhering to existing standards, several methods may be considered for improving overall performance. Using standard forward channel transmission techniques such as selective video encoding, resiliency, and concealment techniques, common in AV transmissions, is one area of focus. Using novel backchannel transmission techniques such as Region of Interest (ROI) tracking, tracking data encoding, resiliency and concealment techniques, is another area of focus. Lastly, keeping the various heterogenous workstreams on forward channel and back channel in relative contextual synchronization in a multi-prong architecture is an area of focus.

About the authors

Michael Gatson is a Distinguished Engineer with the Office of the CSG CTO, Experience Innovation Group (EIG). Michael has over 30 years' experience in the PC and Telecoms Industries. As part of EIG, Michael leads key technology investigations focused on creating new end user experiences.

Vivek Viswanathan is a Software Architect with the Experience Innovation Group (EIG), as part of Office of the CSG CTO, spearheading platform software architecture for Dell's new innovation experiences and platforms. Vivek has 3 years of experience at Dell working closely with Dell's customers, marketing and technology architects, and about 20 years of overall leadership experience in the software industry. Vivek is a Distinguished Technologist in the Technical Leadership Community at Dell.