SCTE CABLE-TEC EXPO 22
SEPTEMBER 19-22 • PHILADELPHIA, PA

2022 Fall Technical Forum
SCTE • CABLELABS • NCTA

Holographics Over 10G

Paving the Way for the Immersive Future

A Technical Paper prepared for SCTE by

Austin Pahl[1]
Software Engineer, Immersive Media Experiences
CableLabs
858 Coal Creek Cir, Louisville, CO 80027
303-661-3867
a.pahl@cablelabs.com

Dr. Abhinav Kshitij[1]
Senior Engineer, Emerging Technologies
Charter Communications
6360 S Fiddlers Green Cir, Greenwood Village, CO 80111
480-376-4115
Abhinav.Kshitij@charter.com

Dell Wolfensparger
XR Architect, Emerging Technologies
Charter Communications

Logan Cho
Software Engineering Intern, Immersive Media Experiences
CableLabs

Thomas Alder
Software Engineer
Formerly OTOY

[1] These authors contributed equally.

Table of Contents

Title **Page Number**

List of Figures

Title **Page Number**

1. Introduction

Depth-based media is an emerging market for users and enterprise that has recently witnessed a sharp uptick in growth and investment. With the rising demand for remote communication in virtual spaces, automation in transportation, maintenance, supply chain, and visualization techniques in healthcare, defense and simulation industry, startups and large companies are competing in this nascent market with the launch of fixed and wearable display units. A host of ecosystems are making efforts to integrate and co-ordinate accelerating development efforts among software professionals and industry experts. Despite growing support, challenges to content generation and transmission include developing an interchange format to support compatibility and an evolved network infrastructure to satisfy bandwidth and latency requirements to reliably and securely deliver immersive content.

The Immersive Digital Experiences Alliance (IDEA) was formed in 2019 to solve the twin problem of *media compatibility* and *media-aware transmission* over a 10G network. IDEA[2] developed 3D Streaming and Intelligent Buffering with the overall objective of enabling optimal immersive content delivery at minimal bandwidth consumption, while preserving viewing experience on multiple classes of immersive display units. The main benefit in bandwidth savings comes through offloading rendering from the core network and moving to the client-side.

The network architecture is robust enough to deliver assets of varying quality to the wide range of available compute resources on fixed and wearable units. The key assumption behind adaptive streaming being that display on a small screen requires fewer details, and therefore lower quality assets would reasonably allow for a satisfactory user experience on mobile and AR glasses. Larger displays however require greater amount of detail for objects closer to the viewer, which is captured and represented in higher-quality assets.

The present work briefly introduces media format interchange and proceed to explain the media-aware network enabled by 3D Streaming and Intelligent Buffering. Section 2 presents current bandwidth challenges to immersive media streaming to a host of different immersive platforms of varying screen sizes. Section 3 explains the 3D Streaming network architecture and Intelligent Buffering over a 10G network, along with a heuristic implementation of asset scheduling contained within the client-side logic. Section 4 evaluates bandwidth usage savings of queue-forming traffic flows of vectorized asset streaming over non-queue forming streaming of rendered frames. Latency measurements on the client-side demonstrate asset scheduling effectiveness in gaining maximum concurrency while fetching assets from remote asset servers. The conclusive sections describe algorithmic improvements made to asset scheduling and integration of current 10G capabilities into the existing network architecture.

2. Challenges to Wider Adoption of Immersive Media

2.1. Immersive Media offers Depth-Based Perception

Immersive media refers broadly to a variety of media that involves depth-based perception and accurately accounting for parallax differences at multiple depth levels. Immersive video allows the viewer to perceive the distance to depicted objects, with the viewer's own eyes as if the objects are physically present in the real world. *Parallax* is one of the main drivers of real-world perception: With viewer perspective shifting, objects nearer to a viewer appear to move relatively faster than objects in the background. VR and AR might be the most well-known examples of immersive media today. In contrast, an image on a computer monitor does not qualify as immersive media because the viewer can only

[2] IDEA Webinar on 3D Streaming, May 4th, 2022 [1]

perceive a flat image on the screen from any perspective and location. The present form of 3D movies also does not qualify as immersive media as parallax is not correctly accounted for by a pair of 3D glasses that stereoscopically superimpose a pair of rendered frames to create an illusion of depth perception.

While immersive media could reasonably be considered an emerging technology, there is a growing support on a host of platforms available in the market today. VR headsets are perhaps the longest running example of an immersive display available today, beginning with the high-profile founding of OculusVR in 2012 [2]. Augmented reality (AR) and mixed reality (MR) are also supported by several products on consumer and enterprise markets today [3], [4]. Together, VR, AR, and MR are collectively known as extended reality (XR).

Volumetric displays, supporting depth-based video, have recently emerged as a new class of display with form factors of a television or a desktop monitor. Volumetric displays are divided into two categories: *eye-tracking displays*, which track a single viewer's eye movements to create the illusion of depth on a 2D screen, and *light field displays*, which send different images out at different angles to produce perceivable depth for multiple viewers at once. Although few volumetric displays have reached the public to date, there are a number of companies already working in this space [5]–[10].

Immersive content generation occurs via digital content creation (DCC), through live captures from the real-world events using advanced camera technologies or using a combination of the two approaches. Digital content creation is the most common approach today, because for the most part it is straightforward to extend existing digital workflows to support immersive displays. Immersive display manufacturers publish free-to-use plugins that integrate with popular graphics toolsets and game engines. In contrast, live capture methods have lagged behind in technological development and adoption for immersive content generation and streaming, although this has been an area of significant innovation in both industry and academia over recent years. Depending on the use case, application requirements, and access to compute and network resources, live capture approaches extend from 2D photo conversion pipelines to depth cameras built into the latest smartphones to high-precision specialized camera systems: Neural Radiance Fields (NeRF) uses a neural representation to achieve exceptionally high fidelity from a sparse set of 2D input images [11]. For a deeper dive, refer to the IDEA white paper [12] on live capture methods and representations.

2.2. Growing industry demand

Over recent years, interest in immersive media has risen significantly. Some have argued that prior to the pandemic, AR and VR reached the "trough of disillusionment" along the Gartner Hype Cycle, which occurs after a product reaches peak inflated expectations and fails to deliver on the hype [13]–[15]. Then during the pandemic, people became acutely aware of the limitations of video conferencing as opposed to face-to-face exchanges. The reasons are numerous – lack of copresence, removal of spontaneous, random encounters, and perhaps worst of all, the newly dubbed "Zoom fatigue" that people experience after extended periods of time spent on video calls [16]–[18]. AR and VR witnessed a significant growth throughout the pandemic, at least in part because of the increased time spent working from home and the prospect of overcoming the limitations of video conferencing [19].

Then the concept of the metaverse came into mainstream attention with Facebook's rebranding to Meta at Facebook Connect 2021[3], while showcasing their progress on building a new platform for rich, immersive social experiences online. Google's Project Starline also investigated immersive media technologies for enhanced telepresence, bringing live 3D video to the video call format [16]. NVIDIA announced its

[3] CEO Mark Zuckerberg's letter to Meta employees [20]

launch of Omniverse platform that allows real-time collaboration on digital twins, architecture, education, and facilities maintenance [21]. Looking Glass Factory recently secured the CIA's venture capital funding to provide immersive displays for intelligence and defense applications [22]. Hollywood movies are increasingly being produced using game engines [23].

Industry giants and startups continue to expand upon XR development – creating applications, utility tools and supporting hardware to allow immersive content generation, streaming and consumption. Display manufacturers, game engine developers, network operators, chip manufacturers and application developers continue to invest capital and participate in evolving ecosystems surrounding immersive technologies. We believe this trend will continue to dominate as demand for immersive content finds an increasing use in improving learning and productivity while enhancing entertainment experiences.

2.3. Rasterized transmission will require prohibitive bandwidths

Newer displays and media pose several challenges that need to be considered to enable the ideal vision of an immersive future. First, the massive variety of methods for capture, encoding, and display of immersive media leads to a "many-to-many" problem when developing workflows and processes related to immersive media. Each method, format, and display has its own advantages and drawbacks, and conversion between representations runs the risk of significant information loss. Even today, conversion among existing 3D scene description formats can lead to problems like missing materials, untranslatable logic, and subtly altered rendering behaviors. This gets worse as the number of features and formats continues to grow.

Bandwidth requirements for immersive displays are expected to grow at an unprecedented rate. State-of-the-art VR headsets today reach resolutions above 4K [24], but light field displays are anticipated to be orders of magnitude higher resolutions than anything available today. To provide a 3D effect without eye tracking, light field displays attempt to mimic the behavior of rays of light bouncing off a real, physical subject. While a pixel on a 2D display unit encodes a single color at 24 bits in total, a *holographic pixel* would need color encoding for each ray emanating from angles discretized in the azimuthal and altitudinal directions. A holographic pixel supporting 90 different angles horizontally (azimuthal) and 90 different angles vertically (altitudinal) could enable viewers to experience a few inches of depth [25], but would require a total of 8100 color encodings. A holographic still image on a UHD-4K display (3840 × 2160) using these 90 × 90 holographic pixels would require 67 gigabytes of uncompressed data. While compression reduces data requirements, their application on compressing immersive media for light field displays is in early prototyping phases, and public data and literature remains scarce.

3. Vectorized Content Delivery over Scalable Networks

3.1. 3D Streaming of Interchangable Media Format

IDEA has published a suite of royalty-free specifications establishing a baseline for interchange of immersive media, known as the Immersive Technologies Media Format (ITMF). The format was initially intended to be used for interchange amongst industry-standard digital content creation (DCC) tools, i.e. for the packaging and creation of 3D synthetic, computer generated, and natural media, including audio and visual media. As a baseline format primarily for use with DCC tools, assets described by ITMF are agnostic to the specific type of device on which they may be presented. For example, visual media will be display-agnostic, so that a subsequent rendering step in a media- and application-aware distribution system can reformat the visual media to match the capabilities of the client display.

While streaming rendered frames require massive bandwidths, real-time streaming of immersive content followed by rendering 3D assets on the client within display units could mitigate the challenges associated with the delivery of immersive media. Local rendering of 3D assets on a game engine runtime has the advantage of asset reuse over multiple scenes, thereby eliminating bandwidth redundancy that comes with streaming rasterized frames. We present ***3D Streaming*** — a system architecture supporting real-time streaming of immersive content to clients by transmitting ITMF scene graphs and associated assets to clients. In addition to reducing light field display bandwidth requirements, 3D Streaming simplifies content distribution to heterogeneous immersive display units. This system also generalizes to other scene graph formats – Universal Scene Description (USD), Graphics Language Transmission Format (glTF) – accommodating diverse application-level requirements and use cases.

The general framework of 3D Streaming finds it form in a network architecture that could be scaled on-prem, in cloud or the edge, dictated by network flows in streaming 3D assets depending on the use case. The key components described here would be considered essential to the overall implementation.

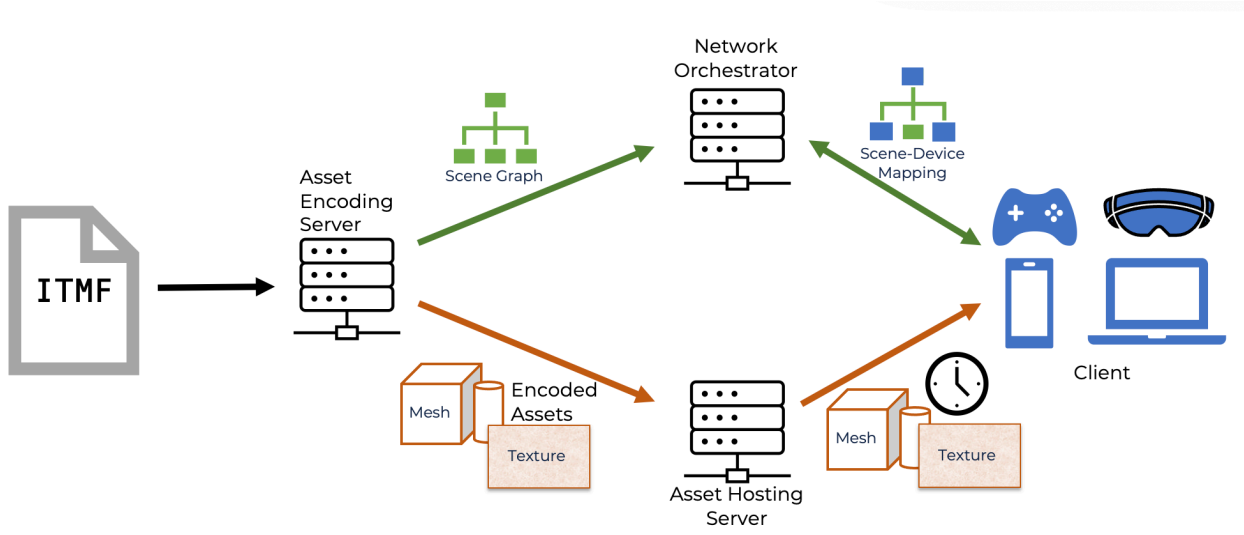

Figure 1 - 3D Streaming Architecture built with an Asset Delivery Pipeline

3.2. System Architecture

Figure 1 describes the architecture of the 3D Streaming demonstrated in IDEA's May 2022 webinar [1]. The end-to-end pipeline of conveying an ITMF scene to an immersive display unit consists of the Asset Encoding Server (AES), the Network Orchestrator (NO), and the Asset Hosting Server (AHS). Initially an ITMF container is ingested by the Asset Encoding Server (AES), which extracts various files from the ITMF container, namely (1) the *assets* (meshes, textures) that spawn in scenes, and (2) the *scene graphs* (XML, JSON) describing the layout and properties of the assets and the scenes (lighting, animation).

The AES pushes the scene graphs to the Network Orchestrator (NO), which is mainly responsible for mapping client requirements to the *asset quality*, and control plane operations related to individual streams. Asset quality (AQ) is defined as an abstraction of mesh compression, texture compression and levels-of-detail (LODs). The AQ-client mapping allows for a real-time adaptive asset streaming to heterogeneous platforms (clients) of widely ranging compute requirements and operating under dynamic network conditions (latency, jitter).

The assets are pushed to the Asset Hosting Server (AHS)[4], defined as an abstraction layer of content delivery network (CDN) hosted on-prem, in public/private cloud and on edge resources. The AHS distributes assets of varying asset quality (AQ) to meet the compute and network requirements of the application (Section 3.3.2). Alternate encodings and/or levels of detail (LODs), i.e., AQ, may be generated by the AES and included in the distribution to the AHS to support performance and hardware constraints on the client.

When a client initiates a 3D Stream, it establishes a connection with the NO, which ensures that any constraints known up front are applied to the scene, such as support for specific asset encodings or display-specific content. The modified scene information is then sent to the client. The client reads the scene information, sends data plane asset requests to the AHS, and finally the client renders the scene in real time on a game engine (Unreal, Unity) runtime. The client makes several asset fetch calls to the AHS at different times during an interactive session (multiplayer gaming, XR application), and a non-interactive session (live streaming, playback) to buffer immersive content on an immersive display unit. Therefore, this *scene awareness* allows for buffering content time-to-time, distributing large downloads over multiple smaller downloads over time, as and when the required assets are relevant to the scene.

3.3. Intelligent Buffering over the 10G Network

Intelligent Buffering refers to the use of network and scene awareness to fetch 3D assets from the AHS such that fetch times and the impact of adverse network traffic conditions are minimized while ensuring best possible QoE for the client. *Network awareness* refers to the consideration of latency and bandwidth and *scene awareness* refers to the consideration of 3D scene properties like asset placement and timing. Together, network and scene awareness enable cloud orchestration for scalable, adaptive streaming of 3D assets of different levels of detail and compression. We consider two different facets of intelligent buffering that can be controlled at runtime to support QoE: (1) Queueing asset fetches according to the time they appear in the scene; (2) selecting asset LODs when prioritization is not sufficient.

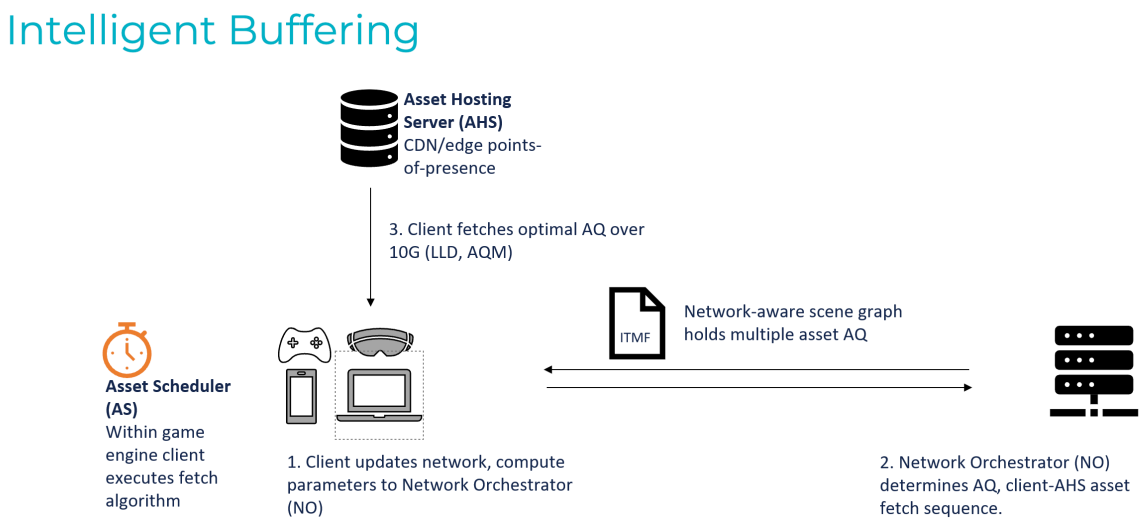

Figure 2 - Intelligent Buffering over the 10G Network

[4] In prior webinars, this was called the "Asset Server".

As noted in Section 3.1, the main advantage of 3D Streaming comes in the form of asset reuse over multiple scenes, therefore mitigating redundant content streaming requirements over several rasterized frames. For example, assets representing natural background objects (forest, rocks, foliage) remain largely static in a scene, and could be fetched once from the AHS and reused on multiple scenes. The uncompressed rendered assets (highest AQ) preserve the content quality under favorable network and compute conditions, much like adaptive bitrate streaming can deliver the highest quality of video content available, and adaptively adjusts frame bitrate as network conditions degrade during video streaming.

The client-AHS connection forms the main network bottleneck of 3D Streaming due to larger bandwidths (compared with client-NO, AES-NO connection) and rendering latency requirements on the client. In most cases, the scene graph is significantly smaller than the assets that fill the scene. During a 3D Stream, it is essential that the client fetches and renders all of the necessary assets on its memory to allow for compute and rendering latencies to display when needed. Failure to display assets during a scene playback would result in loss of necessary details or may temporarily pause the playback while the assets are transferred and loaded (akin to "buffering" on video streams). Such issues arise due to adverse impacts of network congestion, packet loss, and poor memory management.

This critical queue-forming traffic can be optimized by leveraging maturing 10G capabilities of Low-Latency DOCSIS® (LLD) technology with Active Queue Management (AQM) being incorporated into the working design of Intelligent Buffering. While traditional video streaming is download-heavy, immersive traffic would require the increased upstream capability of DOCSIS 4.0 networks to enable live capture and streaming. Mobility considerations include Low-Latency Wi-Fi and Low Latency Mobile Xhaul over converged networks.

The following sections describe intelligent buffering using its runtime variables and metrics, along with an overview of asset selection processes and algorithms as key enabling tools to optimize asset delivery to multiple clients on heterogenous platforms.

3.3.1. Variables and Metrics

In our initial intelligent buffering system, we incorporated the following variables and metrics available at runtime to support effective scheduling decisions. A list of assets from the 3D scene carries the following information:

- Asset type (mesh, texture, animation),
- URLs for multiple AQs of each asset,
- Asset file size associated at each AQ,
- Start and end times of asset visibility in a scene.

While the first three metrics constitute asset metadata available on the AHS and the scene graph, asset visibility time range is not explicitly described in typical scene graph representations. Rather it may be computed in a preprocessing step on the client. Content that is also generated in real-time, such as that from a source game engine, could be instrumented to log and transmit asset visibility, but this is not yet implemented.

From the network connection, the round-trip time (RTT) and download throughput is measured for every asset fetch call from the client to the AHS to determine current network conditions that dictate AQ selection for next fetch call or subsequent fetch calls over a time window.

3.3.2. Asset Quality Selection

The asset quality selection algorithms determine fetch sequencing and AQ. In traditional 2D video streaming, a high bitrate video is typically encoded on the server with multiple bit rates, enabling clients to perform adaptive bitrate: switching between encodings to maximize the content visual quality without experiencing interruptions like buffering. The analog to this in 3D Streaming is asset selection: the AHS provides multiple AQs for data intensive assets like textures and meshes. This way, the intelligent buffering algorithm may react to network conditions to provide similar assurances to those applied by adaptive bitrate.

Most common asset types, especially those that tend to have large file sizes, have existing facilities for lossless and lossy compression. For example, textures are often ingested into game engines using widely used formats like PNG or JPEG. However, the engine may convert image files to specialized, lossy texture compression formats like DXT1 which reduce game package size with minimal or zero impact on decode latency [26], [27]. With respect to meshes, games often include multiple AQs of the same mesh because different amounts of detail are needed when an object is close or far from the camera. In practice, mesh AQs may be generated either automatically or manually based on how much control is needed [28].

3.3.3. Basic Heuristic Example

Here, we demonstrate a simple heuristic approach for network-aware intelligent buffering to illustrate its fundamental application. If a designated latency threshold is met, say, over 45 milliseconds, a “high latency” mode is triggered which means subsequent asset fetches are made using smaller asset variants. In practice, there are many directions that could be taken to improve performance (see Section 5.2 for further discussion).

Consider a client that initiates a 3D Stream of an ITMF file containing a list of N assets. For each asset, a high detail and low detail variant are hosted on the AHS. Once the client receives the scene information, the client sorts the asset list according to time of first appearance. The first asset’s larger variant is fetched, and round-trip time (RTT) is measured. If this time exceeds 45 milliseconds, the high latency mode is toggled so that the next asset fetched uses its smaller variant. Fetch the next asset, measure RTT, and update high latency mode if needed. Repeat this process until all assets have been fetched. Rendering can begin as soon as all assets that appear at the very start of the scene have arrived, at which point rendering and streaming may continue in parallel.

While simple, this approach is primarily intended to illuminate the problem space that is occupied by intelligent buffering. The overwhelming majority of the logic occurs on the client, which facilitates low-cost, high throughput server deployments while achieving the objective of maximal QoE for clients. As previously mentioned, the AHS requires no more than static file hosting, so that component may be deployed on a traditional CDN.

4. Analysis

This section presents early analysis of intelligent buffering, including a description of key network performance metrics and a qualitative comparison to traditional video streaming. Measurements were taken on 3D Streams of small sample scenes.

4.1. Performance Metrics

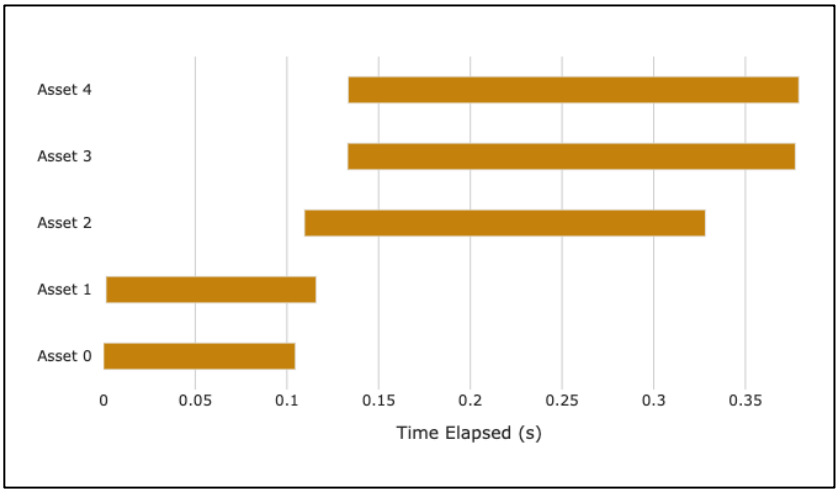

Figure 3 - Timing of Asset Transmissions in a 3D Stream

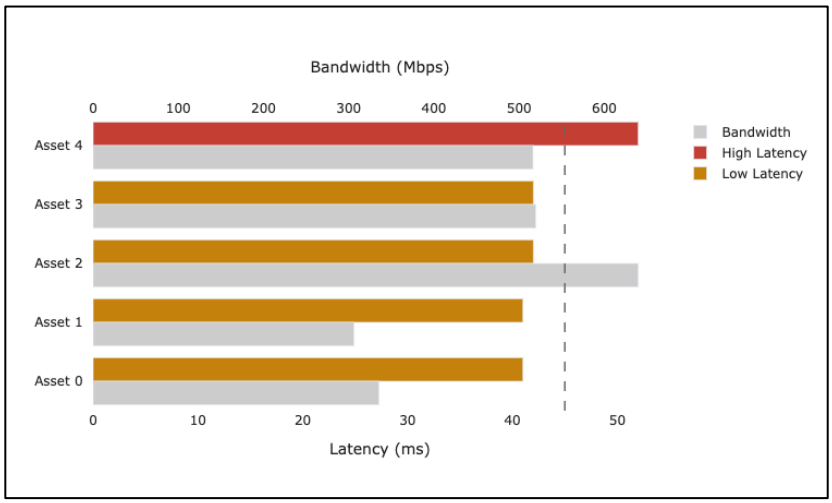

Figure 4 - Measured Latency and Bandwidth of Assets in a 3D Stream

Figure 3 shows a waterfall chart of the asset transmissions that occurred over time during a stream. Assets 0 and 1 were transferred first for the scene to begin rendering, then Assets 2-4 were transferred afterwards. This demonstrates that asset transmissions in a 3D Stream need not occur synchronously: much like a web browser, the client can utilize multiple connections at once to provide a smoother experience where possible. Our goal with intelligent buffering is to develop an approach that minimizes the horizontal length of this plot (total time elapsed transmitting assets).

Figure 4 shows, for each asset transmitted in a stream, the latency and average bandwidth measured from the HTTP response. The vertical dashed line represents the latency threshold from our heuristic example in Section 3.3.3: assets whose latency surpasses 45 milliseconds, Asset 4 in this case, trigger “high latency mode”. Moving beyond the basic heuristic example, bandwidth measurements are a key metric for understanding network conditions to enhance QoE.

4.2. Compared to Conventional 2D Streaming

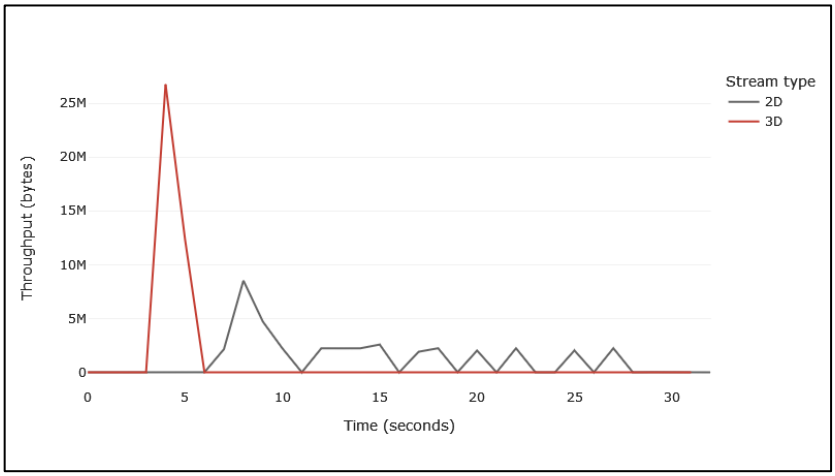

Figure 5 - Measured Latency and Bandwidth of 3D and 2D Streaming

3D Streaming can be considered a complementary solution to traditional video streaming for media delivery: While 3D Streaming does not completely replace the use cases enabled by video streaming, it performs better in various contexts. At a high level, the biggest differences between 3D and video streaming are the use of 3D rendering on the client and the patterns of network transfer. 3D Streaming requires sufficient client resources to perform the real-time render, but this is increasingly common today due to widespread consumer use of graphics processing units (GPUs).

The difference in network behavior is illustrated by Figure 5. This plot shows bytes transferred each second on a 3D Stream and a video stream. In this case, both streams coincidentally transfer about 40 megabytes of data in total. However, the 3D Stream transmits most of that data in one short burst up front, while the video stream transmits small amounts of data steadily over time. This is because the 3D Stream transmits all assets as soon as possible, while the video stream must consistently transmit video frames over the full duration of the content playback.

With the rise of heterogeneous immersive displays, we anticipate that content creators will increasingly need to develop media that supports many different types of displays at once, such as VR headsets, volumetric displays, and 2D displays. Ideally, that media should be tailored as best as possible for each display type. This would be challenging with traditional video: a different version of the content would need to be produced for every single display type, and it would be hard to accommodate the specific advantages of different displays, such as VR's higher degrees of freedom, without a focused, manual effort. Real-time rendering simplifies this process because the display-specific experience can be generated at runtime, often via a specialized plugin or API provided by the display manufacturer. The streaming application may also support custom enhancements like interactivity, which are much more complex to execute in a video streaming context.

As mentioned earlier, immersive displays are leading to increases in effective video resolutions at unprecedented rates, particularly light field displays which increase by an order of magnitude for each degree of freedom introduced. While today's display resolutions are generally well supported by modern network infrastructure and video codecs, it is unclear how long today's systems will remain effective. 3D Streaming is unaffected by this problem because the size of the scene is not correlated with the resolution of the display like a video is. Furthermore, rising adoption of 10G facilitates the delivery of larger scenes faster and more reliably.

As long as the client has sufficient storage, assets only need to be transmitted to the client once each. This means that reuse of assets over time is rewarded with less network chatter. In Figure 5, this is apparent because all the assets were transmitted at the beginning, then nothing else needed to be transmitted for the remaining playback time, keeping the network silent for the rest of the trace. Asset reuse is also useful for scenes that have repetitive content, such as trees, bushes, or buildings in a cityscape. This is already a commonly practiced technique in game development, as it allows the use of *GPU instancing*, which is a performance optimization that renders multiple copies of an object in a scene at one time [29].

5. Discussion

5.1. Related Work

Several online video games and geographic applications have developed systems for the real-time delivery of 3D content over the network [30]–[33]. Similarly, efforts in cloud and distributed rendering have implemented related functionality such as real-time collaboration on shared 3D scenes or scene delivery over the network [34], [35]. All these systems excel at their respective use cases, but do not generalize beyond that. In contrast, 3D Streaming is a general-purpose system for 3D content delivery over the network. In the future, it should be possible to build new networked 3D applications, whether games or content creation tools or otherwise, by leveraging the architecture presented here.

Petrangeli et al. [36] presented a system for streaming AR objects in real time with a mechanism for heuristically adapting LODs according to network condition and scene placement, significantly reducing startup latency and data requirements compared to predownloaded AR scenes. In the context of our work, their approach would be an effective drop-in solution for network and scene awareness in intelligent buffering, likely to be included in future analyses of intelligent buffering methodology. Our architecture also generalizes to broader use cases, including heterogeneous immersive display units with support for tailored experiences.

5.2. Future Work

Enhanced network awareness. We intend to develop a robust analysis of intelligent buffering algorithms. Petrangeli et al.'s work on AR streaming [36] provides one option to analyze, but there may be ways that we can incorporate enhanced network and scene awareness for further improvements.

Device awareness. Another factor to consider in optimizing the QoE of a 3D Stream is the client hardware. We refer to the usage of client hardware conditions and specifications for ensuring the best possible QoE as *device awareness*.

One example of device awareness is to consider the client as a cache comprised of three layers: its GPU memory, CPU memory and storage. In situations where the scene is particularly large or the client is resource-limited, such as embedded or mobile devices, it is possible that the entire scene would not fit on the client at one time. When an asset is delivered to the client over the network, we would store it on an available layer, beginning with GPU memory, falling back to the next when out of space. Coupled with an eviction algorithm that takes into consideration the available space in each layer, along with network and scene conditions, to intelligently free up space, this approach would allow for lower latency, local retrievals of previously seen assets into the scene. Another opportunity for device awareness is to factor in the client's screen resolution into our asset quality selection algorithm. Lower asset quality is less likely to harm QoE on lower resolution displays.

Real-world capture support. We are interested in exploring the deployment of methods for viewing real-world 3D data on the client. Many real-world capture encodings can be embedded into a 3D scene graph [12], [37], and some of them, including NeRF, can render in real time [38]. As real-world 3D capture becomes more accessible, this will likely become a core use case for immersive displays. Embedding these captures into scene graphs will also enable new forms of mixed content: for example, one could imagine a virtual gallery filled with 3D scans of real art.

6. Conclusion

The current paper explores recent progress made on the development of a network architecture for scalable, vectorized content distribution to multiple platforms, rendered on client-side game engines. An end-to-end testing of a content delivery pipeline demonstrates significant bandwidth savings, while preserving content quality during transmission and allowing for asset reuse over multiple scenes. The architecture leverages modularity and scalability to allow for high availability of content over core network and cloud deployment. By streaming content over a 10G network, queue-forming traffic of immersive content can be delivered over reasonable times.

A key challenge for the present architecture lies in compute requirement on the client-side, especially with the growing demand for lighter wearable XR platforms to improve user experience. Future testing of the asset scheduler will determine algorithmic effectiveness in improving AQ adaptability, while *tail-end latency* ranges are expected to be curtailed primarily by deploying 10G capabilities on the existing platform without the need for making significant hardware changes.

As newer versions of game engines feature photorealism with even greater detail, vectorized content streaming could be the preferred choice of streaming immersive content. Although it may satisfy the demands of latency-sensitive applications like gaming, live event streaming (sports, concerts) will deliver live-captures (large bandwidths) but driven by latency requirements. The current architecture presents a general framework that can be adapted for live-capture and streaming by using low-latency techniques, delivering traditional video frames and 3D assets on separate queues.

Abbreviations

2D	two-dimensional
3D	three-dimensional
AES	Asset Encoding Server
AHS	Asset Hosting Server
AQM	Active Queue Management
AR	augmented reality
CDN	content delivery network
DCC	digital content creation
glTF	Graphics Language Transmission Format
GPU	graphics processing unit
IDEA	Immersive Digital Experiences Alliance
ITMF	Immersive Technologies Media Format
LLD	Low Latency DOCSIS
LOD	level of detail
NeRF	neural radiance field
NO	network orchestrator
QoE	quality of experience
RTT	round-trip time
UHD-4K	ultra-high-definition 4K
USD	Universal Scene Description
VR	virtual reality
XR	extended reality

Bibliography & References

[1] *ITMF 3D Streaming Demo Webinar*, (May 04, 2022). Accessed: Jul. 14, 2022. [Online Video]. Available: https://www.immersivealliance.org/videos/

[2] G. Kumparak, "A Brief History Of Oculus," *TechCrunch*. https://social.techcrunch.com/2014/03/26/a-brief-history-of-oculus/ (accessed Jul. 13, 2022).

[3] "Microsoft HoloLens | Mixed Reality Technology for Business." https://www.microsoft.com/en-us/hololens (accessed Aug. 03, 2022).

[4] "Enterprise augmented reality (AR) platform designed for business | Magic Leap." https://www.magicleap.com/en-us/ (accessed Aug. 03, 2022).

[5] "Simulated Reality | 3D Display technology," *Dimenco*. https://www.dimenco.eu (accessed Jul. 13, 2022).

[6] "Sony Spatial Reality Display |Sony US," *Sony Electronics*. https://electronics.sony.com/more/spatial-reality-display/p/elfsr1 (accessed Jul. 13, 2022).

[7] "Leia Inc. – 3D Lightfield Experience Platform." https://www.leiainc.com/ (accessed Jul. 13, 2022).

[8] "Looking Glass Factory: The Hologram Company," *Looking Glass Factory.* https://lookingglassfactory.com (accessed Jul. 13, 2022).

[9] "Light Field Lab." https://www.lightfieldlab.com/ (accessed Jul. 13, 2022).

[10] "Avalon Holographics Inc.," *Avalon Holographics Inc.* https://www.avalonholographics.com (accessed Jul. 13, 2022).

[11] B. Mildenhall, P. P. Srinivasan, M. Tancik, J. T. Barron, R. Ramamoorthi, and R. Ng, "NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis," in *Computer Vision – ECCV 2020*, Cham, 2020, pp. 405–421. doi: 10.1007/978-3-030-58452-8_24.

[12] "Photographic Live Action Capture for Immersive Media," Immersive Digital Experiences Alliance. Accessed: Jul. 14, 2022. [Online]. Available: https://www.immersivealliance.org/download/download-photographic-live-action-capture-for-immersive-media/

[13] M. Brenner, "The Resurgence of AR and VR Content in a Post-Pandemic World," *Marketing Insider Group*, Jul. 27, 2021. https://marketinginsidergroup.com/content-marketing/the-resurgence-of-ar-and-vr-content-in-a-post-pandemic-world/ (accessed Jul. 13, 2022).

[14] J. Pace, "XR and the Self-Inflicted Trough of Disillusionment," *Medium*, Nov. 07, 2019. https://arvrjourney.com/xr-and-the-self-inflicted-trough-of-disillusionment-e2177c6b33fe (accessed Jul. 14, 2022).

[15] "Gartner Hype Cycle Research Methodology," *Gartner*. https://www.gartner.com/en/research/methodologies/gartner-hype-cycle (accessed Jul. 14, 2022).

[16] J. Lawrence *et al.*, "Project starline: a high-fidelity telepresence system," *ACM Trans. Graph.*, vol. 40, no. 6, pp. 1–16, Dec. 2021, doi: 10.1145/3478513.3480490.

[17] C. Morris, "This startup wants to replace your Zoom meetings with holograms," *Fast Company*, Mar. 11, 2022. https://www.fastcompany.com/90730176/startup-matsuko-zoom-meetings-holograms (accessed Jul. 14, 2022).

[18] V. Ramachandran, "Four causes for 'Zoom fatigue' and their solutions," *Stanford News*, Feb. 23, 2021. https://news.stanford.edu/2021/02/23/four-causes-zoom-fatigue-solutions/ (accessed Jul. 14, 2022).

[19] S. Vardomatski, "Council Post: Augmented And Virtual Reality After Covid-19," *Forbes*. https://www.forbes.com/sites/forbestechcouncil/2021/09/14/augmented-and-virtual-reality-after-covid-19/ (accessed Jul. 21, 2022).

[20] M. Zuckerberg, "Founder's Letter, 2021," *Meta*, Oct. 28, 2021. https://about.fb.com/news/2021/10/founders-letter/ (accessed Jul. 14, 2022).

[21] "NVIDIA Announces Omniverse Open Beta, Letting Designers Collaborate in Real Time — from Home or Around the World," *NVIDIA Newsroom*. http://nvidianews.nvidia.com/news/nvidia-announces-omniverse-open-beta-letting-designers-collaborate-in-real-time-from-home-or-around-the-world (accessed Aug. 01, 2022).

[22] L. Fang and J. Poulson, "The Brooklyn Hologram Studio Receiving Millions From the CIA," *The Intercept*, May 27, 2022. https://theintercept.com/2022/05/27/metaverse-cia-military-hologram-looking-glass-factory/ (accessed Aug. 01, 2022).

[23] M. Seymour, "Art of LED Wall Virtual Production, Part One: 'Lessons from the Mandalorian,'" *fxguide*, Mar. 04, 2020. https://www.fxguide.com/fxfeatured/art-of-led-wall-virtual-production-part-one-lessons-from-the-mandalorian/ (accessed Aug. 01, 2022).

[24] "Varjo XR-3," *Varjo.com*. https://varjo.com/products/xr-3/ (accessed Jul. 21, 2022).

[25] "SO, WHAT IS A HOLOGRAPHIC DISPLAY?," *Avalon Holographics Inc.* https://www.avalonholographics.com/resources/what-is-a-holographic-display (accessed Jul. 21, 2022).

[26] "Textures in Unreal Engine." https://docs.unrealengine.com/5.0/en-US/textures-in-unreal-engine/ (accessed Jul. 17, 2022).

[27] "Texture Format Support and Settings." https://docs.unrealengine.com/5.0/en-US/texture-format-support-and-settings-in-unreal-engine/ (accessed Jul. 17, 2022).

[28] "Geometry Best Practices for Artists." https://developer.arm.com/documentation/102496/0100/Level-of-Detail---LOD (accessed Jul. 17, 2022).

[29] "Unity - Manual: GPU instancing." https://docs.unity3d.com/Manual/GPUInstancing.html (accessed Jul. 22, 2022).

[30] "3D Tiles," *Cesium*. https://cesium.com/why-cesium/3d-tiles/ (accessed Jul. 18, 2022).

[31] "VRChat," *VRChat*. https://hello.vrchat.com (accessed Jul. 18, 2022).

[32] F. Brown, "Microsoft wants to bring back Flight Simulator to show it supports PC," *PC Gamer*, Jun. 11, 2019. Accessed: Jul. 18, 2022. [Online]. Available: https://www.pcgamer.com/microsoft-wants-to-bring-back-flight-simulator-to-show-it-supports-pc/

[33] "Second Life - Virtual Reality, VR, Avatars, and Free 3D Chat." https://secondlife.com/ (accessed Jul. 18, 2022).

[34] "Omniverse Platform for Virtual Collaboration," *NVIDIA*. https://www.nvidia.com/en-us/omniverse/ (accessed Jul. 18, 2022).

[35] "The Render Network." https://render.x.io (accessed Jul. 18, 2022).

[36] S. Petrangeli, G. Simon, H. Wang, and V. Swaminathan, "Dynamic Adaptive Streaming for Augmented Reality Applications," in *2019 IEEE International Symposium on Multimedia (ISM)*, Dec. 2019, pp. 56–567. doi: 10.1109/ISM46123.2019.00017.

[37] Q.-A. Chen, "nerf_Unity." Jul. 21, 2022. Accessed: Jul. 21, 2022. [Online]. Available: https://github.com/kwea123/nerf_Unity

[38] A. Yu, R. Li, M. Tancik, H. Li, R. Ng, and A. Kanazawa, "PlenOctrees for Real-time Rendering of Neural Radiance Fields," in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada, Oct. 2021, pp. 5732–5741. doi: 10.1109/ICCV48922.2021.00570.

