The evolution of delivering immersive media over 5G/Cloud

Mauricio Aracena, VRIF President/Standardization Manager

Virtual Reality Industry Forum (VRIF)/ Ericsson AB.

Bill Redmann, Director of Standards, Immersive Media Technologies

InterDigital

Dr. Du Ho Kang, Senior Specialist

Ericsson AB

Dr.-Ing. Louay Bassbouss, Senior Project Manager R&D

Fraunhofer Institute for Open Communication Systems (FOKUS)

Dr. Sebastian Schwarz, Research Leader

Nokia

Written for presentation at the
SMPTE 2023 Media and Technology Summit

Abstract. *Aspects of the 5G ecosystem, now being deployed in 3GPP releases 17/18/19, include split processing and edge computing resources. These can be enlisted to offload manipulation and rendering of immersive datasets, thereby reducing the burden on the mobile device. Instead, the mobile device receives only the rendered video, for one or both eyes, ready-made for display by the headset.*

Key design criteria for 5G connectivity to near-edge compute resources have been established based, in large part, on augmented and mixed reality use cases that rely on network slicing and quality of service (QoS) management, impose limits on bidirectional communication latencies, and establish minimum requirements for the compute resources themselves. Additionally, 5G system is becoming "XR aware" with a specific set of features for XR offloading. These features provide not only the proper network requirements, but also additional intelligence to consider device power and capacity considerations critical to scaling the deployment of XR services.

The authors are solely responsible for the content of this technical presentation. The technical presentation does not necessarily reflect the official position of the Society of Motion Picture and Television Engineers (SMPTE), and its printing and distribution does not constitute an endorsement of views which may be expressed. This technical presentation is subject to a formal peer-review process by the SMPTE Board of Editors, upon completion of the conference. Citation of this work should state that it is a SMPTE meeting paper. EXAMPLE: Author's Last Name, Initials. 2022. Title of Presentation, Meeting name and location.: SMPTE. For information about securing permission to reprint or reproduce a technical presentation, please contact SMPTE at jwelch@smpte.org or 914-761-1100 (445 Hamilton Ave., White Plains, NY 10601).

This paper describes the evolution of XR applications from merely delivering media over 5G to how to best utilize 5G cloud/edge infrastructure to process and distribute advanced immersive experiences and content to lightweight, wireless, head-mounted displays. Critical use cases impose clear network requirements and how those are met in 5G is examined. Lastly, the shape of potential relationships within the stakeholder ecosystem are presented, all with the goal of guiding content providers (developers and service providers) through the paradigm shift from device-centric to network-centric XR services.

Keywords. *5G, virtual reality, augmented reality, mixed reality, VR, AR, MR, immersive, XR, network slice, edge compute, edge render, cloud gaming, hyperscale cloud provider, communication service provider.*

The authors are solely responsible for the content of this technical presentation. The technical presentation does not necessarily reflect the official position of the Society of Motion Picture and Television Engineers (SMPTE), and its printing and distribution does not constitute an endorsement of views which may be expressed. This technical presentation is subject to a formal peer-review process by the SMPTE Board of Editors, upon completion of the conference. Citation of this work should state that it is a SMPTE meeting paper. EXAMPLE: Author's Last Name, Initials. 2022. Title of Presentation, Meeting name and location.: SMPTE. For information about securing permission to reprint or reproduce a technical presentation, please contact SMPTE at jwelch@smpte.org or 914-761-1100 (445 Hamilton Ave., White Plains, NY 10601).

Introduction

With the availability of more Augmented Reality (AR) and Virtual Reality (VR) headsets, people are starting to experience more realistic and interactive immersive services. Thanks to the advanced technology embedded into the headset we are getting more powerful devices, able to compute and render images of increasing resolution and quality. Yet development of longer and more realistic experiences is progressing slowly, limited by battery consumption, device form factor, and heat dissipation constraints. Many service providers have started to deploy services in the cloud to address these issues. However, running the application in the cloud imposes additional challenges: latency, bandwidth, reliability, and availability of the service. 5G cloud architecture can overcome those issues with solutions that can be applied incrementally, each differently affecting the complexity of the application, but each improving the ultimate experience for the user. Additionally, the ultimate vision for 5G architecture as applies to immersive experiences calls for new relationships among the ecosystem members – the consumer, communications service provider, hyperscale cloud provider, and developer/service provider. This paper examines key aspects to launch an immersive service using 5G cloud infrastructure. First, reviewing recent offerings and developments, then walking through a set of use cases each exploiting more and more offload to the cloud. We follow with a description of 5G technologies that satisfy the use cases, and finally, reflect on the evolution of the stakeholders' ecosystem in relation to their technical and commercial relationships to establish an immersive service using 5G.

Existing Media Offloading Techniques and New Use Cases

Cloud Gaming

Schmidt provides an overview of cloud gaming and its recent advancements in [1]. Cloud gaming technology enables users to play video games without the need for a local device having the graphics processing capabilities necessary to run the game. Instead, the game is executed on a remote server and streamed to the user's device via the internet. This capability allows users to play high-end games on devices that may not have the necessary hardware to run the game locally, such as low-end laptops or smartphones.

One of the principal benefits of cloud gaming is the ability to play games on devices with lower hardware specifications, as the game runs on the more powerful hardware of a remote server, rather than the local device. This allows a user access to a wider range of games and the ability to play them at higher rendering settings, even on devices that would otherwise be incapable of running them.

Another advantage of cloud gaming is the ability to play games on different devices. As the game is running on a remote server, it can be accessed from any device having an internet connection and compatible software. Thus, users can play such games on various devices, including laptops, smartphones, tablets, even those with no dedicated gaming platform, such as smart TVs.

Popular cloud gaming platforms include:

1. **Google Stadia:** Launched in 2019, as one of the first cloud gaming services. Stadia offered a wide selection of games that could be played on devices having a Chrome

browser. It could provide 4K resolution gaming and allowed users to purchase games individually or subscribe to the Stadia Pro service for free games and better streaming quality. The service went offline permanently in January 2023. Despite Stadia's discontinuation for failing to raise sufficient user engagement, the technological advancements made since its launch were substantial. However, other cloud gaming platforms achieved greater popularity, generally for offering a larger game catalogue.

2. **Microsoft Xbox Cloud Gaming (formerly xCloud):** Launched almost a year after Stadia as part of their existing social gaming service Xbox Game Pass Ultimate, Microsoft's cloud gaming service allows players to stream games from a vast library on Xbox consoles, Windows PCs, and Android devices, with support expanding to other platforms.
3. **NVIDIA GeForce Now (aka NVIDIA Now):** Released in 2020, following betas beginning in 2016, the current model lets users stream their existing game libraries from popular digital platforms. Supported devices include desktop computers running Windows or macOS, mobile devices running Android, iOS, and others. It offers a free tier with limited playtime and a premium subscription for extended sessions and priority access.
4. **Sony PlayStation Now (now PlayStation Plus):** Allows players to stream and download a wide range of PlayStation games on their PlayStation consoles or PC. Unlike other cloud gaming systems, a dedicated hardware architecture matching the PlayStation game consoles is used. Both subscription-based access and individual game rentals are offered.
5. **Shadow:** Shadow provides a full Windows 10 PC in the cloud, allowing users to access their favorite PC games (and other Windows apps) on devices running Windows, macOS, Android, iOS, and Linux through a dedicated app.
6. **Amazon Luna:** Launched in 2022, Amazon's service streams games to devices running Windows, macOS, Android, iOS, plus Amazon Fire TV. A growing library of games is offered, which run on Amazon Web Services servers hosting NVIDIA graphics cards.

All these services leverage cloud computing and streaming technology to offer video game streaming to users. When a user plays a cloud-rendered game, it executes on a powerful computer, in some cases having special or dedicated graphics hardware, in a data center and the video and audio output are streamed to the user's device, to be displayed and played in real-time. The objective behind the design of the technology is to provide an uninterrupted and engaging gaming experience, allowing users to access high-quality games on a variety of devices without the need for costly hardware. However, a fast and stable internet connection is necessary to ensure optimal streaming quality.

5G Edge Computing

5G technology offers new opportunities for high-speed communication between users and computing machines. 5G offers a low latency configuration that can revolutionize data exchange. By bringing cloud-like capabilities physically closer to the end user, that is, to the “edge” of the network, 5G hardware can reduce transmission distance and increase transmission speed. This approach to computation allows for more efficient performance of critical tasks when compared to centralized cloud computing, where data must be transmitted to a more distant central server.

One important feature of 5G is configurability, enabling it to satisfy diverse requirements that depend on the intended use. The design of 5G addresses three broad application areas, shown in Table 1: Enhanced Mobile Broadband (eMBB), Massive Machine Type Communication (mMTC), and Ultra Reliable Low Latency Communication (URLLC). Thanks to the flexibility and configurability of the 5G infrastructure, it can enable far more advanced XR scenarios.

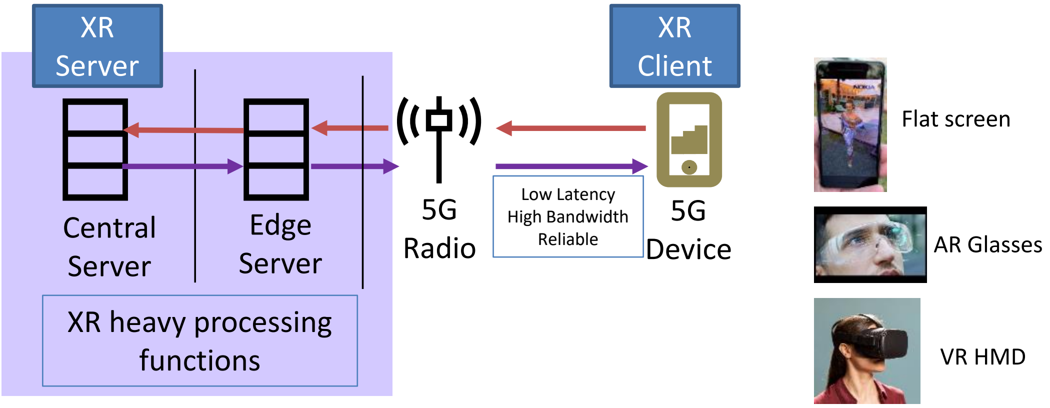
Table 1: Three main type of 5G configurations[1].

5G eMBB	5G mMTC	5G URLLC
• Peak data rate: 10 to 20 Gbps • 100Mbps whenever needed • 10000 times more traffic • Supports macro & small cells. • Allows high mobility - 500 km/h • 100x Network energy savings	• High device density (10^6 devices per km^2) • Low data rate (1 to 100 Kbps). • 10-year battery life (reduced complexity) • Asynchronous access.	• Less than 1 ms air interface latency. • 5 ms end to end latency between User Equipment (UE) (mobile) and 5G eNB (base station). • 99.9999% availability • Low to medium data rates (about 50 kbps to 10 Mbps).

5G Cloud XR Infrastructure

Improvements that 5G edge processing brings to the quality of experience can enable immersive experiences on “thin” devices, i.e., devices with low processing power.

Figure 1 shows the generic 5G Cloud XR infrastructure [2] in which an XR application has computational support from an XR application server wherein the XR functionality may be wholly or partially computed and/or rendered by the server before transmission to the XR client. Since the XR use case may have strict latency, reliability, and bandwidth requirements, the XR server executes its portion of the XR functionality and serves the XR client using appropriate 5G traffic characteristics (e.g., specifying a variant of the 5G URLLC configuration). Depending on the XR use case requirements, the XR server may be implemented on an edge server, to satisfy latency requirements.



[1] These numbers are peak (extreme) values, but can be configured to achieve specific use case requirements.

Figure 1: Generic 5G Cloud XR infrastructure

The 5G Cloud XR architecture enables:

- Enhanced usability: Devices are lighter weight with reduced processing load (less heat)
- Decreased battery use, providing longer usage time
- Reduced local storage requirements since content is streamed, not stored.

The significance of edge computing in Metaverse-related applications, where cloud-based solutions come with high latency, is discussed by Dhelim et al. in [3]. They propose an edge-based architecture that has demonstrated a 50% reduction in latency. Edge computing appears to be a potential solution for Metaverse applications, but they concluded further maturation was necessary to determine the best architecture for specific applications.

From Device to Cloud: Offloading Scenarios

Modern immersive media collects under the banner of extended reality (XR), a set of technologies including virtual reality (VR), augmented reality (AR), and mixed reality (MR). VR is the oldest of these, its goal being for a user's sensory experience, sight, sound, and preferably touch, the real world is replaced by computer-supplied stimuli. AR allows its user to perceive the real world, whether directly or through an apparatus (e.g., a camera and attached display), but then adds annotations and objects appearing to overlay the world, thus injecting information and other content. MR is a mashup of these: A user sees the real world, but the machine supplied annotations and objects are realistically integrated, not merely overlaying the view.

Each immersive increment, from VR to AR to MR, represents a substantial increase in complexity. Such complexity is further increased with each incremental degree of quality. As the corresponding computational and other burdens grow, so does the challenge to engineers designing devices to mediate such interactions.

For example, if a headset is to be lightweight, its battery must be small. If a battery is to operate for meaningful durations, the power demanded of it must be modest. However, complexity begets power consumption, which increases power demand and seems to beg for a larger battery. Providing sophisticated XR experiences seems, at first glance, to be at odds with designing the lightweight, long-lasting, XR devices to deliver them.

The 3rd Generation Partnership Project (3GPP) identified a solution in version 16 of their Technical Report 26.928 [4] which is, fundamentally, to offload the heaviest XR computations to a system that isn't battery-constrained, leaving the user device to substitute one or more less-complicated communication tasks. As simple as that sounds, the innovation adds a whole new set of challenges. Communication introduces power consumption and overhead of its own. Further, to be usable, latencies in an XR system, such as between the time something is detected and the time an XR presentation reacts, have to be kept low, ideally below thresholds of human perception, otherwise the sense of immersion and acceptability of the XR system is degraded. Communication links and buffering add to latencies and must be closely managed. Constraints such as these, apropos to a specific immersive experience of interest, were considered by 3GPP when adding 5G XR support and inspired a variety of capabilities relating to the use of compute resources located at or near the edge nodes of the network. Having such resources tightly engaged by a user's immersive media device, can make extraordinary XR experiences possible.

Core Use Cases for XR Delivery

3GPP collected relevant, well-understood, XR use cases and consolidated those that were related or imposed similar technical requirements. The core use cases follow an evolutionary progression, which is useful to aid understanding.

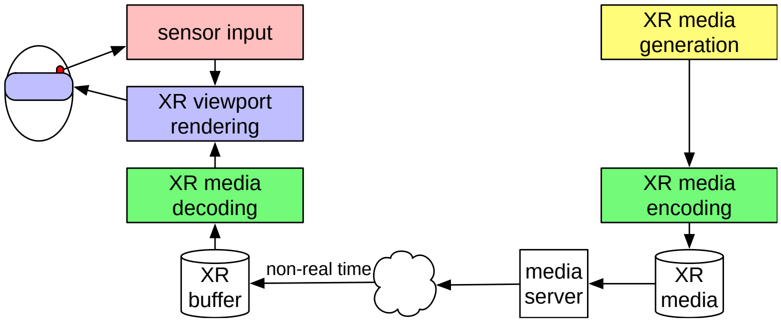


Figure 2: Viewport Independent Delivery (offline)

The simplest interaction between an XR device and the network is just downloading XR media, rather than being completely self-contained. An XR device appears on the left half of Figure 2. While illustrated as a head-mounted display, a handheld device such as a tablet or smartphone could be shown instead. A sensor to detect the user's position in either three or six-degrees of freedom informs the rendering of the XR viewport, the user's view into the XR presentation. That rendering draws on XR media, decoded from an XR buffer. XR media comes in many forms, including both proprietary and standardized ones. For example, 3D point cloud geometries with corresponding textures, whether static or animated, can be conveyed using codecs developed by the Motion Picture Experts Group (MPEG) 3D Graphics Coding group (3DG), such as their Video Point Cloud Compression (V-PCC) as described in ISO/IEC 23090-5:2021 [5]. Depending on the type(s) of XR media conveyed, the sensor input can allow a user to look about a surrounding environment, or to study and move about a nearby object.

In this use case, the XR buffer is pre-populated by a connection to services on the right side of Figure 2 through which the user device obtains a network download of XR media through a media server. The XR media generation and XR media encoding can be performed in advance, and the download needn't be too aggressive. In this scenario, the download operates in non-real time.

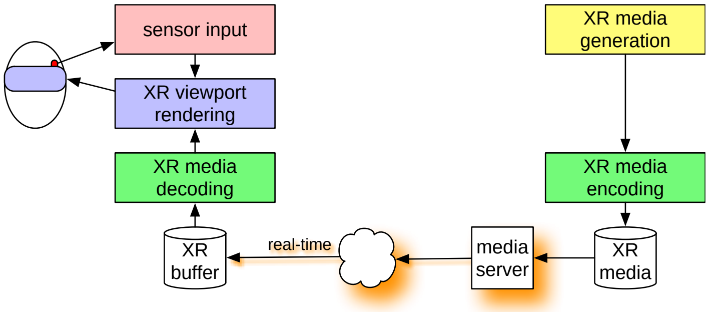


Figure 3: Viewport Independent Delivery (streaming)

By imposing a significant requirement on the network, namely that the content can be streamed over the network in real-time, the amount of memory needed on the user device can be dramatically reduced, though the necessary communications bandwidth is increases. This is shown in Figure 3. Here, and in Figures 4-7, the orange drop-shadows indicate additions relative to the prior figure(s). Further, XR media generation and encoding becomes real-time as well, as the media source can be from another XR device and the sending and receiving of XR media is symmetrical between the two devices, though only one direction is illustrated.

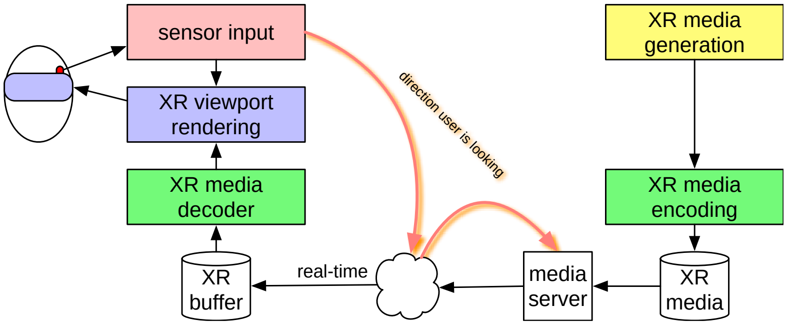


Figure 4: Viewport-dependent Streaming

Note that in Figure 2 and Figure 3, the server has no idea of the position or orientation of the user device relative to the XR media and where it goes in the scene. That changes in Figure 4, where the sensor data is sent to the server and the server selects which portions of the XR media to send and in what level of quality. For instance, the portion of XR media corresponding to a user's gaze direction could be sent at a maximum quality level, while portions corresponding to the user's peripheral vision can be sent at much lower quality levels using less bandwidth, e.g., by using lower resolution and/or more a higher degree of compression. Portions that would appear behind the user can't be seen at all, even if the user were to turn quickly, and might not be sent at all.

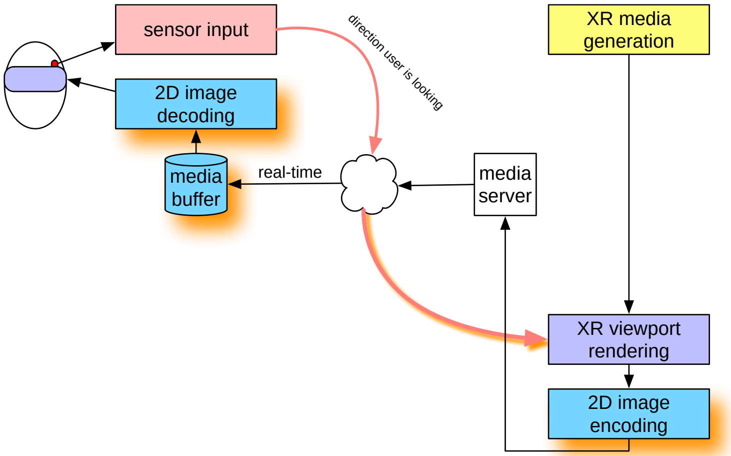


Figure 5: Viewport rendering in Network

The advantage of viewport-dependent rendering is that only XR media that is likely to be seen is sent, saving not only communications bandwidth, but also the complexity of decoding and processing media that would not be used. The tradeoff is that by the time XR media is received, the user orientation may have changed such that the XR media is no longer of a quality or coverage appropriate to the new position. Ideally, the latencies are such that the updated position information is sent, and the quality and coverage are restored before the user can perceive that any compromise has been made.

An even greater energy savings is achieved in Figure 5, where the responsibility for running the XR engine that renders the XR viewport is offloaded to the server. Here, the sensor information produces a request for a view in a particular direction, but instead of selecting what portion of the XR media is to be sent and at what quality level, the XR media is rendered by the server, resulting in a 2D image that is lightly compressed, to save bandwidth without adding much latency, and sent to the user device. Upon receipt, the user device decodes the 2D image and displays it directly. In this configuration, if the XR media generation is live, then no encoding or decoding of the XR media is needed.

The transfer of the XR rendering responsibilities to the server-side drastically reduces the complexity of the work being performed on the user device. 2D image decoding is a well-understood technology and, in many instances, occurs in specialized chips, thereby reducing the power requirements even more. Even so, care must be taken that latencies of the system are controlled to remain small and the rendering framerate constant.

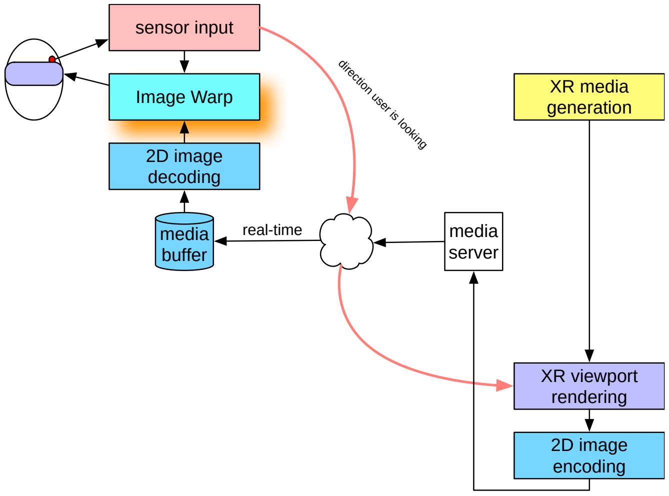


Figure 6: Split Rendering with Asynchronous Time Warping (ATW) Correction

The tight requirement for low latency and constant framerate is relaxed somewhat by the configuration shown in Figure 6. This comes from the addition of a late-stage 2D rendering adjustment called Asynchronous Time Warping (ATW) correction. Here, not only is the sensor data sent to the server for use in rendering the XR viewport, but when it is time to display the resulting decoded 2D image, any intervening changes detected by the sensor is used to warp the 2D image to achieve a better apparent alignment. For example, if in the intervening time, the user's head has turned to a little to the right, then the 2D image is displayed offset a corresponding amount to the left. Similarly, if the user's head has tilted or moved forward, the image can be rotated or zoomed accordingly. ATW adjustments can be made more frequently than once per rendered frame, hence “asynchronously”, and there needn't be a uniform number of adjustments per frame. In some degree, ATW reduces the need for consistency in the framerate at which the XR viewport rendering takes place.

There are cases where even the advantages of ATW are not sufficient to maintain an acceptable XR presentation. This occurs where 3D XR objects appear close to a user, and the appropriate perspective changes from motion parallax, whether caused by the user's motion or the motion of the XR object relative to the user, are more severe.

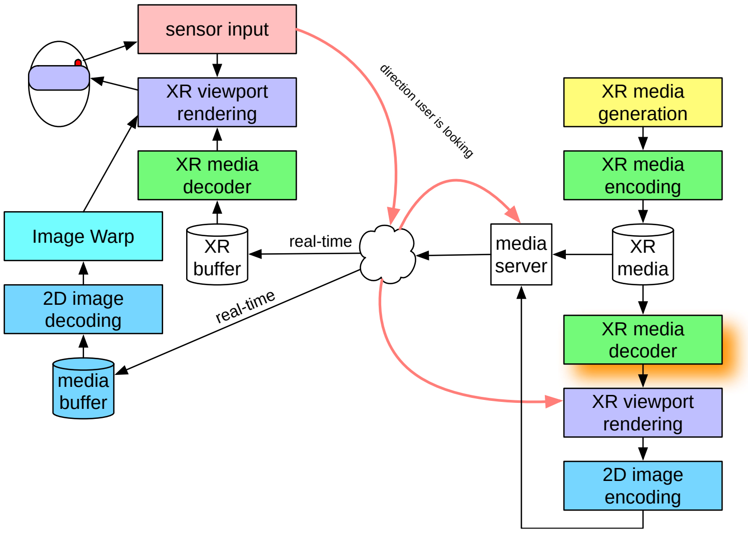


Figure 7: VR Split Rendering with XR Viewport Rendering in Device

The solution to the motion perspective issue involves using a hybrid approach shown in Figure 7. The advantages of the 2D split render of Figure 6 is combined with the improved accuracy of a XR viewport render on the user device, where the sensor data is as current as possible. Since the server side is informed about the user's position and orientation, it is possible for a pruning of the XR content is able to select only the likely-relevant portions, as in Figure 4. It is further possible that different portions of the XR media are treated with different update rates. For example, changes in the perspective of distant elements, though 3D, is small and thus less noticeable, that of elements that are nearer. The user device performs the final XR viewport rendering which composites all the locally rendered XR media elements with the post-ATW elements that were rendered by the server.

In still more advanced scenarios, the sensors of user devices acquire not only the position and orientation of the user, but potentially more elaborate information about the pose of the user, (e.g., to properly pose an avatar of the user for self-viewing or viewing by others) or video capture of the environment (e.g., for simultaneous location and mapping, i.e., SLAM, and/or to capture the environment for reproduction for other users). Figure 8 shows a more general distributed computing architecture for XR, where processing of such sensor information may occur locally on the user device, remotely on the XR server, or both.

Both the server and user device of Figure 8 can generate an XR scene from the sensor data. Where the video is processed to represent an XR object, corresponding metadata describes that object's position relative the user and/or other objects allowing proper placement and assembly of the scene. This allows the server to perform the XR viewport rendering, offloading the bulk of that computation burden as in Figure 5, while at the same time, the user device is able to keep its own rendition of certain items in tight registration with real-world anchors, and all rendered items in consistent positioning relative to each other.

The distributed computing architecture is particularly flexible for the user device, as tasks can be offloaded as needed, or, if simple enough, kept locally for responsiveness.

A final core use case, shown in Figure 9, represents an overlay architecture which could be used with various of the configurations discussed in Figure 2 to Figure 8, and is particularly directed to providing conference services among two or more users.

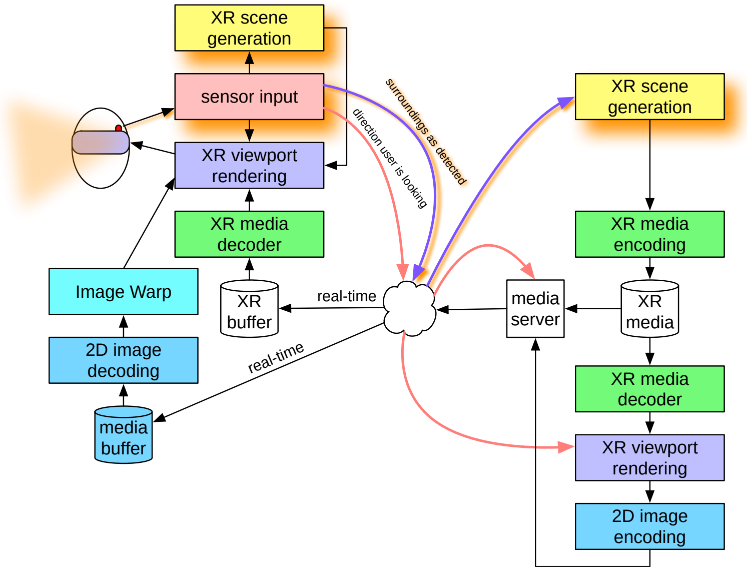


Figure 8: XR Distributed Computing Architecture

The XR Conference Server responds to requests for session initiation, while individual clients request or accept connections. Unsurprisingly, session control based on 3GPP's IP Multimedia Core Network Subsystem (IMS) and doesn't require any extraordinary resources.

Any commonly shared resources, for example, the XR media elements assembled to present the conference room or other environment that is being shared virtually by the users, are available for transfer and referenced by the scene configuration.

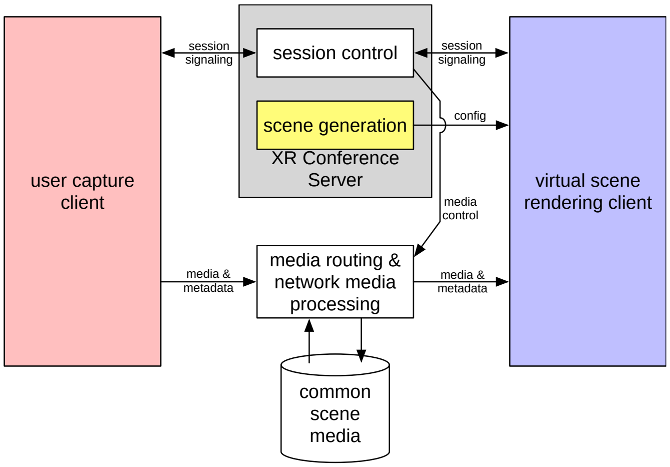


Figure 9: General Architecture for XR conversational and conference services

In this architecture, the sensor apparatus of the user device incorporates a user capture client. The user capture client can determine the user's pose, which can be used to render an avatar not only for others, but for rendering a self-view for the user, too. The capture client can capture or otherwise provide a 3D model of the user's face, perhaps including eye tracking. This allows the user to be rendered without appearing to be wearing an HMD, making communication more natural. If and as needed, the image processing and scene understanding tasks necessary for generating a user avatar, whether for self-view or viewing by others, can be dispatched to the network media processing block, relieving the user device of that computational burden.

3GPP analyzed each of the elements in these core use cases to establishment acceptable requirements for bandwidth, latency, packet error rates, and the like, separately for inbound and outbound paths, and accumulated those to define aggregate network characteristics necessary to satisfy those use cases.

XR Offloading Deployment

There are many ways to deploy immersive services using the 5G cloud to support the use cases described above. Service providers can choose how they would like to offload their applications. As Alriksson et al. describe in [6], one can deploy different XR edge-processing architectures to support different degrees of offload: Low, Mid, and High offload for an XR user case providing SLAM (Simultaneous Localization and Mapping). Figure 10 shows how eight significant XR functions required for SLAM can be divided among an XR device and the network edge.

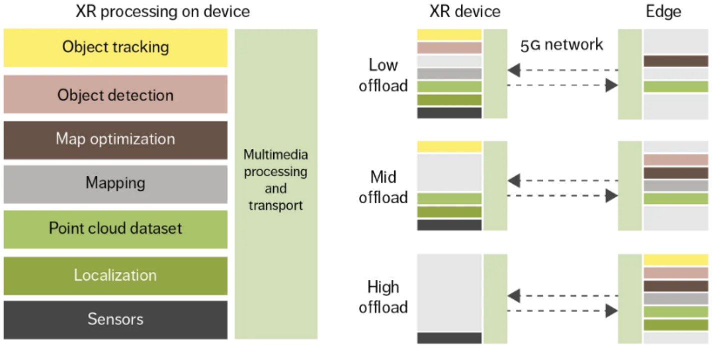


Figure 10: Example degrees of offloading (low, mid, high) for a scenario requiring SLAM and object detection

In the low-offload architecture, almost all processing is done on the device: The sensors provide data to populate the point cloud dataset, which is used for spatial map generation and localization. As portions of the point cloud and spatial map are built, they are compressed and transmitted to the 5G edge to be merged into existing global spatial map data, with the edge performing updates in the map optimization process. Pertinent portions of the global map can be returned. Object detection and tracking are performed on the device. Outside of the SLAM process, rendering can occur locally on the device or at the network edge.

In the mid-offload case, localization and object tracking are performed on the device. Point cloud data sets are created from sensor data by the device and sent (using video codecs) to the edge for object detection, spatial mapping, merging with the global point cloud datasets and spatial map. Here, any overlay rendering occurs at the network. The edge-rendered video and audio content are encoded into video and audio streams and transmitted to the device along with the rendering data.

In the high-offload case, only sensor data is sent over the uplink. Sensor data includes camera data. The image data is encoded using video compression such as Motion Pictures Experts Group (MPEG) High Efficiency Video Coding (HEVC) or Versatile Video Coding (VVC).

It is important to recognize that higher degrees of offloading lower power consumption in the device, but require higher data rates, since more communication is needed between the edge processing and the device.

Likewise, offloading functions – partially between the cloud and the device – impose additional challenges for the application developer. Designing and developing "network-aware applications," is necessary, meaning that an original application, intended to run locally on the device, is not trivially reused in scenarios where some functionality is migrated to the cloud. An example might be using remote GPUs, where the application developer must use a method to send GPU instructions and retrieve rendered images from the cloud.

One way to overcome such challenges is for the developer to implement or use a third-party's platform to separate and offload specific functionalities (as in the case of low and mid offload scenarios of Figure 10).

Completely offloading the application to the cloud can be simpler. This way, only sensor and user interface information (graphics, video, audio, etc.) are sent back and forth, reducing the complexity of application development and maximizing the reusability of application software in both device-only and fully cloud-offloaded scenarios.

Network Requirements

As suggested above, 5G connectivity requirements depend heavily on the XR use case and what functions are being offloaded. Some general requirements were developed by Alriksson et al. in [6] and are summarized in Table 2.

Table 2: 5G network requirements by use case

Use case	Download (Mbps)	Upload (Mbps)	One-way latency (ms)	Frame reliability (%)
Cloud gaming	8 - 30	~0.3	10 - 30	≥ 99
VR	30 - 100	< 2	5 - 20	≥ 99
AR	2 - 60	2 - 20	5 - 50	≥ 99

The bitrates and latency requirements of Table 2 can serve as a baseline for XR services, but there are some new traffic characteristics observed:

- Video traffic dominates: The three use cases require sending and/or receiving image information, mostly in the form of video; hence, high bandwidth is required.
- Multiple Traffic Flow: There are several data streams such as video data, audio data, different types of sensor data, and control information that are transmitted. It is common practice to use the user datagram protocol (UDP) for its lightweight, minimalist characteristics to transmit such data streams sharing the same UDP connection. This is a challenge for network optimization, since it is hard for the network to distinguish the different data streams. When these data streams share the same connection, they might not be synchronized with the corresponding video streams, e.g., with video at 60 fps, audio packets every 20ms, haptic information refreshing at above 100Hz, etc.

- Rate adaptation dynamicity: Periods of excessive buffering buildup inevitably cause disruptions in the application since buffering causes latency. Increases can result from a reduction in channel capacity or other disruptions such as handovers. When channel capacity is lowered, applications must adapt to the lower bitrates, but each application might do this in different ways.
- Latency variation and application dependency: Variations in network latency are much harder to handle and can impact the user experience. This has notoriously seen in cloud gaming use cases. Latency variations are best handled at the user device. Therefore, network latency jitter should be limited (or at least minimized) to a range that the jitter buffer of the application can manage.

Evolution – "Enhancements for XR" in the 5G Releases

Network Traffic Characteristics

With Release 17, 3GPP undertook to analyze a variety of XR use cases and corresponding offloading scenarios to understand the resulting 5G network traffic characteristics, from which they could identify specific radio access network (RAN) requirements. Their report [7] determines what 5G features are needed to satisfy the XR use cases as well as what enhancements are required for future releases with regards to XR capacity (how many user devices per cell), coverage, mobility, and power consumption necessary for communication by the user device.

5G Specific XR Enhancements

3GPP is constantly updating the 5G standard to provide better quality of service (QoS) for XR applications. A high-level table summary is presented in Table 3, highlighting important XR-related enhancements.

Table 3: XR-related enhancements by 3GPP release

Rel. 17	Motivation and Evaluation of Traffic characteristics: TR 38.838	Work finished in 2022
Rel. 18	XR Awareness in RAN - Core features (TR 38.835) [8] ⊖ In a PDU Session, a logical connection between the UE and a data network that can support one or more flows, support new semi-static Packet Data Unit (PDU) set QoS parameters per flow. ⊖ Support core signaling for dynamic PDU "Set Information" and "Identification" in a given flow - RAN features (RP-230786) [9] ○ Signaling of semi-static/dynamic PDU set information between RAN nodes ○ Provisioning by UE of XR traffic assistance information e.g., periodicity, uplink traffic arrival information	Work to be finished by end 2023

	XR specific capacity enhancements o Allows multiple Configured Grant (CG) physical uplink shared channel (PUSCH) transmission occasions (TOs) in a period of a single CG PUSCH configuration o Dynamic indication of unused CG PUSCH occasions based on Uplink Control Information (UCI) by the UE o Buffer Status Report (BSR) enhancements including at least new Buffer Status Table(s) o Delay reporting of buffered data in uplink o Discard operation of PDU Sets for DL and UL XR specific power saving enhancements Reduce UE[2] power without sacrificing QoS (latency, bit rate, reliability)	
Rel.19	Further enhancements (to be decided December 2023)	

XR Awareness in RAN

The RAN[3] obtains information about an XR application and its traffic characteristics. If known by the RAN, this information assists in increasing capacity and achieving power-saving gains. For example, knowing the application PDU (set) size or Packet Delay Budget (PDB) for a given PDU set can allow the network to schedule resources more efficiently, in turn increasing capacity. Similarly, obtaining the periodicity of different flows and their jitter information can assist the network to select suitable Discontinuous Reception (DRX) configurations. Across the studied use cases [7], the network's requirements significantly depend on the traffic characteristics, offloading scenarios, and user-targeted quality. Traffic characteristics and network requirements vary from application to application, even varying within a session of the same application. Thus, knowing important characteristics of the applications and their traffic allows the network to manage its own resources well, while meeting the requirements for each specific application.

For 5G Networks to utilize the "XR awareness in RAN" feature, the XR application informs the RAN, via the Core Network, about its traffic characteristics. For instance, this may include video resolution, framerate, audio and haptic information, desired bandwidth, latency, as well as other information that may shape traffic. This challenges an application developer, since the application must communicate directly with the 5G Core. To alleviate this, a high-level Application Programming Interface (API) integrated in a software development kit (SDK) will simplify software development and boost the adoption of XR applications using 5G.

[2] User Equipment (UE)

[3] RAN: 5G Radio Access Network

Quality of Service Configuration in 5G

An important challenge is to ensure a coherent mapping of the above-discussed XR application flows into RAN parameters, with the most important ones being:

1. **5G Quality of Service Identifiers (5QI):** Indicate, for specific data flows, an appropriate set of QoS characteristics, e.g., priority level, packet delay or packet error rate, etc. The set of QoS characteristics can be standardized or not.
2. **Data Radio Bearers (DRB):** Logical channels set up to request allocation of radio resources from the radio scheduler, consistent with the 5QI for one or more flows.
3. **PDU Session:** Data connection within a flow, allowing RAN partitioning; efficient bundling of DRBs and QoS-flows; impacted by operator configuration;
4. **Slices:** data network name (DNN) partitioning; efficient way to reach different DNNs as per Network Slice Selection Assistance Information; impacted by application use case.

A suitable configuration across these dimensions must respect current 3GPP system limits as well as individual handset, network vendor, and operator constraints, resulting in specific values for the parameters described above. Three example XR 5QI | DRB | PDU | Slice configurations are illustrated in Figure 11. The first, A, has XR traffic differentiated via dedicated QoS flow / DRB shared with other QoS flows within the same PDU session. B differentiates its XR traffic with a separate PDU session, while C offer complete traffic separation in a separate slice.

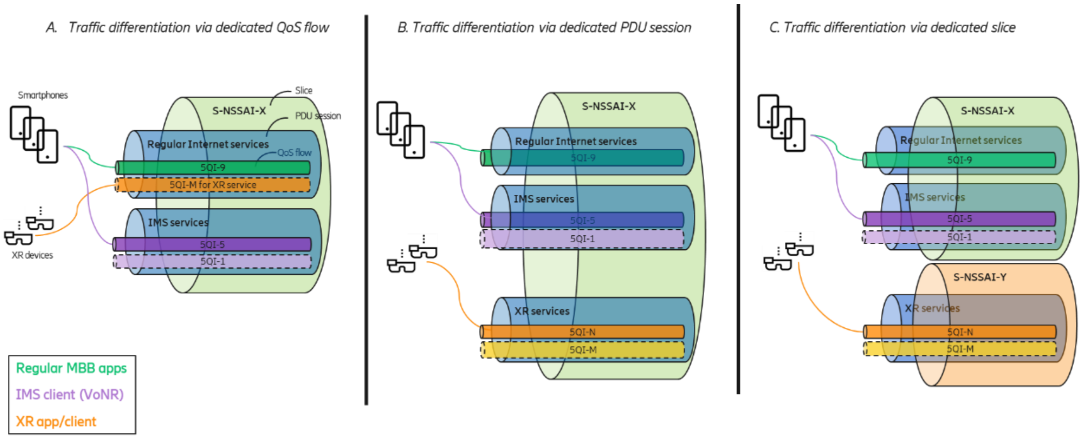


Figure 11: Possible XR 5G network configuration options

User Equipment Route Selection Policy (URSP)

As shown in Figure 11, a network slice can provide the connectivity requirements needed for XR services. Network slicing will play a crucial role in enabling service providers to deliver advanced applications, since slices are easily customized to support specific requirements. This differs significantly from enhanced Mobile Broadband (eMBB), which only provides best-effort connectivity.

User Equipment Route Selection Policy (URSP) is a 3GPP standard that enables application steering with network slices. URSP functionality is already being implemented in 5G iOS and Android devices and in 5G network infrastructure to allow devices multiple network slices on the same device with traffic detection and steering capabilities.

At a high level, URSP works as follows:

- Network operators or communication service providers (CSPs, also known as telco providers) offer specific network slices, such as low latency, high bandwidth, etc., by subscription.
- When launching an application that requires specific connectivity, a check with the operator determines whether the device is entitled to use a specific slice.
- If no, the operator can offer an additional new consumer or enterprise subscription from a list of adequate configurations.
- With a subscription in place, the application gains access to the appropriate network slice.

Network APIs

Network exposure, or service exposure, makes network capabilities available for CSPs and third parties, such as service providers or application developers [10]. Policies for security and data integration, and accessibility of network data and resources can be assigned for different ecosystems, preparing the way for the creation of innovative applications.

The architecture of 5G has two components, the Network Exposure Function (NEF) and Policy Control Function (PCF), playing important roles for network exposure. NEF is a 3GPP entity that exposes network capabilities to third parties, and PCF to provide a framework for policy control. However, using these interfaces in 3GPP requires a deep understanding of the 5G system. Network Application Programming Interfaces (NW APIs) have been introduced to simplify use of 5G features. In contrast to URSP, where there are well-defined, pre-built network slices offered directly to consumers, NW APIs provide a means to customize network capabilities specifically for a particular service, configured directly by service providers. URSP is “QoS device initiated,” while using NW APIs is “QoS network initiated”. Service providers establish a Service Level Agreement (SLA) with CSPs to enable the use of NW APIs for a service provider. Network vendors may provide their own NW APIs. Initiatives are underway to boost adoption of NW APIs through well-defined and harmonized APIs across network vendors and CSPs.

CAMARA[4]: Telco Global API Alliance

CAMARA is an open source project within the Linux Foundation aimed to define, develop and test (NW) APIs [11]. The goal is to provide an abstraction from Network APIs to Service APIs. These developer-friendly APIs do not require telco expertise, yet satisfy data privacy and regulatory requirements, thus facilitating application-to-network integration from various network vendors and CSPs. CAMARA works in close collaboration with the GSMA Operator Platform Group to align API requirements and publish APIs.

CAMARA not only provides QoS APIs but also offers a mechanism for Edge cloud discovery and application onboarding to the cloud. At the moment of writing this paper, some the CAMARA APIs categories are listed here[5]:

- **Quality on Demand**: Set quality for a mobile connection

[4] https://camaraproject.org/

[5] https://github.com/camaraproject/WorkingGroups/blob/main/APIBacklog/documentation/APIBacklog.md

- **Edge Cloud**: Provide and manage application images to be deployed on resources within the operator network
- **Identity and Consent**: Protect user privacy and comply with regulations

Evolution of the XR Stakeholders Ecosystem and its Performance

For XR experiences, we can identify the following stakeholders:

- Consumer: our “user”, the person using (if not wearing) an XR device to consume an XR experience
- Communication Service Provider (CSP): provides 5G infrastructure and QoS connectivity
- Hyperscale Cloud Provider (HCP): provides cloud infrastructure close to the 5G network (cloud edge)
- Service Provider: provides the XR services (the XR application)

As the applications are moved to the cloud, the roles of the stakeholders in this ecosystem evolve. For example, in some situations CSPs might further offer cloud services as part of their own infrastructure.

For B2B use cases, such as XR services for major enterprises and specific industries, a cloud infrastructure might be located on the premises and could be owned entirely by the enterprise or the industry. On the other hand, a typical small business is likely to rely on an HCP for their cloud support.

Figures 2 through 7 showed many ways, with varying degrees of benefit, to offload an XR application to the cloud and the selection depends entirely on the service provider. To illustrate the evolving relationships among the stakeholders, we examine three such selections.

Figure 12 illustrates the application configuration similar to the minimalist offloading of Figure 2, with the XR application and render (XR App 1) taking place on the XR device, the cloud supplying only a stream of XR content.

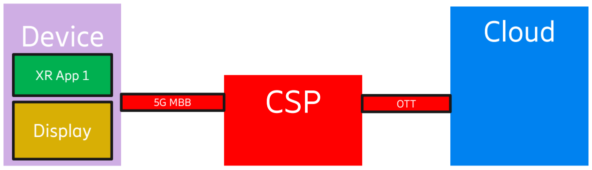


Figure 12: App 1 running entirely on the device, with cloud providing some data via OTT

Figure 13 is akin to Figure 5, but also cloud gaming, where the entirety of the XR application and render (XR App 2) resides in the cloud, with only the 2D rendering transmitted for display on the, possibly stereoscopic, XR device.

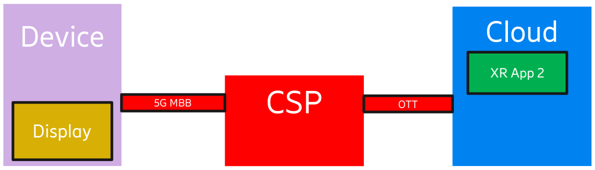


Figure 13: App 2 is completely offloaded, but it is still an OTT service

Figure 14 is also akin to Figure 5, with the XR application and render (XR App 3) in the cloud but employing advanced XR features.

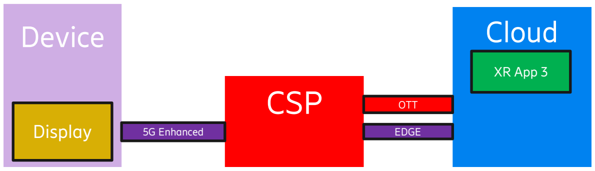


Figure 14: App3 is completely offloaded to the edge and uses 5G enhanced features for XR

XR Apps 1, 2 and 3 all provide similar services, but come from different service providers having made different selections. Being deployed differently, the three apps may provide different user experiences. The comparisons are summarized in Table 4.

Table 4: XR App-related enhancements by service configuration

Configuration	App 1	App 2	App 3
XR Offloading	App 1 runs entirely in the device	App 2 runs in cloud and offers XR performance at least equal to App 1	App 3 runs in cloud and offers highest XR performance
Connectivity	Based on an OTT service Some connectivity is required for fetching some data from cloud Uses 5G MBB (best effort)	Based on an OTT service Connectivity is required for rendered video streams. 5G MBB (best effort) but hopes for high bandwidth and low latency	Exploits 5G Enhanced Features. Secured QoS for the required bandwidth and latency Well characterized requirements allow RAN configuration to reduce battery load
Type of Cloud	Central cloud	Central cloud	Cloud edge close to CSP -or- Central cloud if no edge agreement with CSP
5G CSP Agreement	Regular 5G subscription	Regular 5G subscription	Special subscription or agreement (SLA): URSP – consumer-initiated connectivity subscription -or- Service Provider has SLA with CSP specific to XR App 3

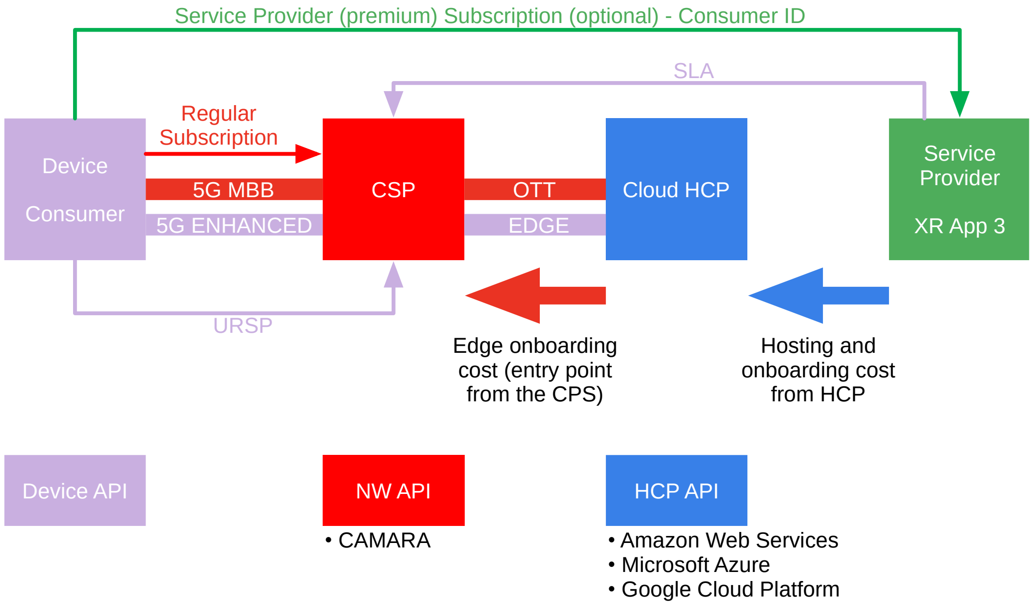


Figure 15: Stakeholder ecosystem for 5G cloud XR applications

In Figure 15, we diagram the relationships among stakeholders in the ecosystem.

The consumer, through their device,

- has a regular subscription for 5G eMBB connectivity
- may have a URSP subscription to access advance services using 5G Enhanced
- may have a subscription, possibly for a premium, with a Service Provider for advanced XR services in the cloud

Communication Service Provider (CSP)

- Offers regular 5G subscriptions
- May offer advanced network slices via URSP
- May offer specific connectivity directly to service providers via Service Level Agreement (SLA) to support 5G Enhanced to consumer, in which case the URSP may not be needed if the Consumer pays for the connectivity requirement via the premium subscription directly to Service Provider

Hyper Scaler Provider (HCP) – Cloud provider

- Hosts the XR application (XR App 3) in the Cloud
- Service providers may use specifics HCP APIs
- May be present at the network edge, where the CSP provides an entry point to 5G network.
- Agreements are needed between the cloud provider and CSP, and between the service provider and cloud.

Service Provider

- Provides the XR service to the consumer
- Offers a subscription relationship to the consumer, whether regular or premium

- May have an SLA agreement with CSP to support cloud edge and 5G Advance features via NW APIs

Conclusion

This paper describes the evolution of immersive media applications offloading to the cloud. Offloading an application can be done in many ways, from partially to completely offloading the entire application, and depends entirely on the service provider. Each setup imposes specific challenges in term of application design and portability of the applications. The simplest for application development and reusability is to maximize offloading to derive the maximum benefit in terms of XR device power consumption reduction and best quality of experience. As soon as the application relies on any degree of offloading, there is increased dependency on connectivity. 5G Cloud infrastructure is designed to meet the connectivity requirements for a wide range of XR applications, and 5G can be configured in many ways to fulfill these requirements. The main challenge is creation of an ecosystem where service providers (the application developers) can easily develop applications that exploit the 5G cloud infrastructure. A key aspect is using URSP to offer the consumer specific slices having better and more reliable connectivity than the default best-effort MBB. Network APIs offer the next step on 5G QoS customization and provide a better link between service providers and CSPs and further provide a harmonized way to onboard edge cloud applications to CSP networks. Finally, the success of the 5G cloud offloading ecosystem will depend on how relationships evolve among stakeholders in the ecosystem; once it is well-defined and understood by all stakeholders, particularly the service providers on how to use 5G, and CSPs on how to monetize 5G, we will see advanced immersive applications running in the cloud.

Bibliography

[1] S. Schmidt, "Assessing the quality of experience of cloud gaming services," Springer International Publishing, 2022

[2] M. Aracena, M. O'Doherty, O. Schreer, S. Schwarz, "Solving the Challenge of Volumetric Video Production and Streaming: An End-to-end Perspective of Technologies and Device Ecosystem Capabilities and Performance," *SMPTE 2022 Media Technology Summit*

[3] S. Dhelim , T. Kechadi, L. Chen, N. Aung, H. Ning, L. Atzori, "Edge-enabled Metaverse: The Convergence of Metaverse and Mobile Edge Computing," 2022, arXiv preprint arXiv:2205.02764.

[4] "ETSI TR 126 928 V16.0.0 (2020-11) - 5G; Extended Reality (XR) in 5G (3GPP TR 26.928 version 16.0.0 Release 16)"

[5] "ISO/IEC 23090-5:2021(en) Information technology — Coded representation of immersive media — Part 5: Visual volumetric video-based coding (V3C) and video-based point cloud compression (V-PCC)"

[6] F. Alriksson, D. Kang, C. Phillips, J. Pradas, A. Zaidi, "XR and 5G: Extended reality at scale with 5G networks - Ericsson," [Online]. Available:

https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/xr-and-5g-extended-reality-at-scale-with-time-critical-communication

[7] "3GPP ETSI TR 38.838 Study on XR (Extended Reality) evaluations for NR," [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3736

[8] "3GPP ETSI TR 38.835 Study on XR enhancements for NR," [Online]. Available: https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=4048

[9] "3GPP ETSI RP 230786 Updated WID on XR Enhancements for NR," [Online]. Available: https://www.3gpp.org/ftp/TSG_RAN/TSG_RAN/TSGR_99/Docs/RP-230786.zip

[10] "Enable innovation with open network exposure," Ericsson. [Online]. Available: https://www.ericsson.com/en/core-network/network-exposure

[11] "CAMARA: Telco Global API Alliance," GSMA. [Online]. Available: https://www.gsma.com/futurenetworks/ip_services/understanding-5g/camara-telco-global-api-alliance/

[12] F. Alriksson, D. Kang, C. Philips, J. Pradas, A Zaidi, "Technology Review: Extended Reality and 5G," Ericsson [Online]. Available: https://www.ericsson.com/4a492d/assets/local/reports-papers/ericsson-technology-review/docs/2021/xr-and-5g-extended-reality-at-scale-with-time-critical-communication.pdf