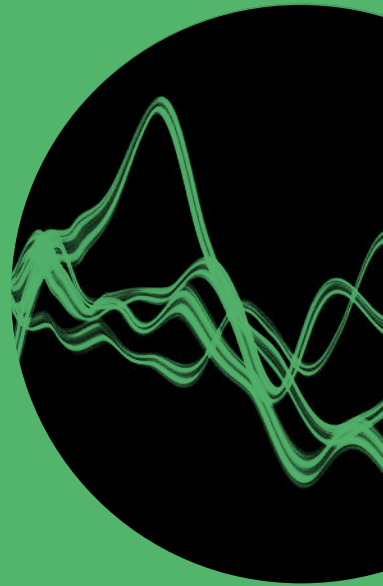


Review

ERICSSON
TECHNOLOGY



FUTURE NETWORK
REQUIREMENTS FOR
EXTENDED REALITY



ERICSSON

Future network requirements for extended reality applications

Many experts believe that new and emerging extended reality technologies will lead to the next major paradigm shift in telecommunications, with lightweight XR glasses ultimately overtaking smartphones as the dominant device type in mobile networks. This evolution has major implications on the requirements for future networks.

FREDRIK ALRIKSSON,
OSKAR DRUGGE,
ANDERS FURUSKÄR,
DU HO KANG,
JONAS KRONANDER,
JOSE LUIS PRADAS,
YING SUN

The availability of discrete, attractive and high-performing extended reality (XR) devices is key to the mass-market success of XR applications, and the process of creating them is well underway. Device developers are working intensively to evolve the device form factor from today's bulky head-mounted displays (HMDs) to lightweight and stylish glasses that will depend on computation offloading to a much greater extent.

■ It is of the utmost importance that future networks are designed and dimensioned to meet the requirements of XR applications. Low latency, high reliability and high data rates will be essential to sustain user quality of experience (QoE) in XR applications while offloading computation services. XR applications are also going to generate a significant amount of additional traffic, which networks must have the capacity to handle.

To gain a better understanding of XR network

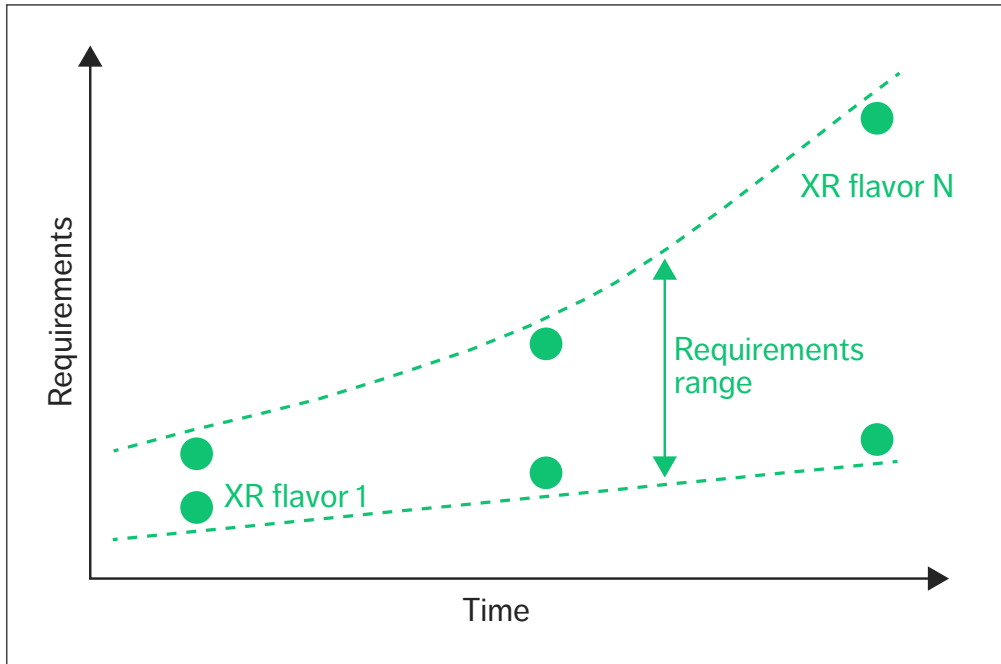


Figure 1 XR requirements with stringency increasing over time

requirements, Ericsson has derived a set of XR operating points as concrete samples from the requirements space, as illustrated in *Figure 1*. For the sake of simplicity, we refer to the selected samples (represented by the dots in *Figure 1*) as XR “flavors.” We have used the flavors to make projections regarding how the network requirements of XR applications, devices and computation offloading variants are likely to evolve over time, and what implications the evolving requirements will have in terms of network capabilities.

To start, we identified which XR applications are

most likely to be used and how advanced we expect them to be. In the next step, we considered computation offloading and its impact on connectivity. The XR flavors result from a combination of these two aspects, together with device capabilities, that enables us to derive the requirements on the network for these specific examples.

XR applications and user behavior

Extended reality is an umbrella term that covers immersive technologies ranging from virtual reality

Terms and abbreviations

AR – Augmented Reality | **DL** – Downlink | **HMD** – Head-Mounted Display | **MBB** – Mobile Broadband
| **Mbps** – Megabits per Second | **ms** – Milliseconds | **QoE** – Quality of Experience | **RAN** – Radio-Access
Network | **SLAM** – Simultaneous Localization and Mapping | **UL** – Uplink | **XR** – Extended Reality

●● DAILY XR USAGE IS EXPECTED TO INCREASE SIGNIFICANTLY AS XR DEVICES BECOME SLIMMER AND MORE CONVENIENT TO WEAR FOR LONGER PERIODS ●●

to mixed reality and augmented reality (AR) [1]. XR devices are expected to bring immersive experiences that merge digital objects, information and overlays with the real world. XR represents three major shifts:

1. from flat screens and 2D content to immersive 3D content displayed and spatially located in the real world
2. from graphical user interfaces to natural interaction within the real world
3. from context switching (between screen and real world) to context awareness.

Application areas that have a high potential for wide uptake include gaming and entertainment, retail and shopping, social communication and virtualized work. These comprise consumption of XR content in a single user setting as well as communication services such as holographic communication that include two or more XR users [2].

XR applications differ in terms of complexity of graphics, as well as on how aligned the graphics need to be to the real environment. The range of XR applications includes everything from a simple navigation service with an arrow pointing toward a destination to advanced holographic communication with moving 3D objects perfectly aligned with the real environment. Unsurprisingly, the most advanced applications with the highest degree of computation offloading have the greatest impact on network requirements, and the simplest ones with the least amount of computation offloading have the smallest impact. It should be noted, however, that a

single application can be elastic in the sense that it can operate with different resolutions, for example.

Network requirements are also impacted by usage location and the distribution of use over the time of day. Consumer studies indicate that the introduction of XR may not lead to abrupt changes in user patterns – that is, consumers will likely continue to use the same type of services at the same locations and times as now. In this sense, XR is “only” a new type of user interface. Over time, XR is expected to create additional value for consumers across a broad range of application areas [3, 4].

Daily XR usage is expected to increase significantly as XR devices become slimmer and more convenient to wear for longer periods. But even when it becomes feasible for large numbers of people to wear XR devices all day, it is reasonable to expect the types of XR services that are used to vary over the course of the day, similar to the usage patterns of smartphone services today. Therefore, we expect it would be adequate to dimension future networks to support heavy XR usage for as little as one hour, or just a few hours, per day.

XR functions and computation offloading

Many of the functions needed to realize XR experiences are computationally heavy. Rendering and spatial computation are the two most important. Rendering refers to the process of generating graphics to be shown to the user in the HMD. Depending on the complexity and resolution of the graphics, rendering can become very computationally demanding. Spatial computation functionalities include simultaneous localization and mapping (SLAM), object detection and object tracking. These are primarily used to gain an understanding of the local environment.

Rendering and spatial computation are both good candidates for offloading to a remote server – that is, an edge or cloud server with adequate computation capability. Computation offloading enables more powerful processing through cloud computation and reduced HMD complexity in terms of a smaller form factor, increased battery life and reduced heat generation. Cloud computing allows for more

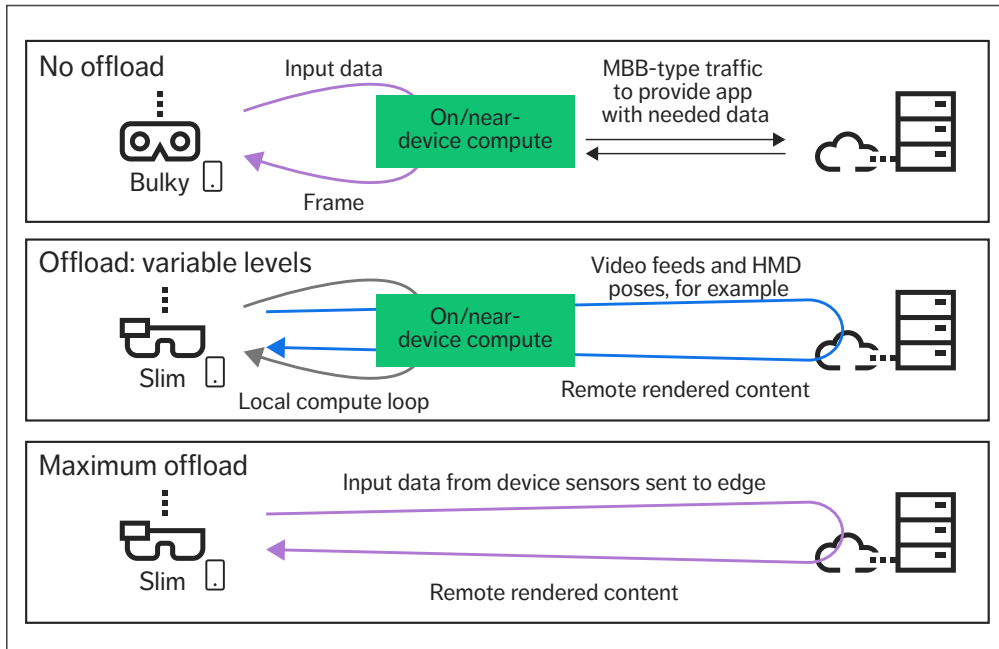


Figure 2 Three approaches to XR computation processing

complex (and faster) computations that enable more immersive experiences with higher resolution, quality and frame rates, as well as supporting multi-user experiences in dynamic environments.

Figure 2 illustrates the three main approaches to XR computation processing. In the “no offload” scenario at the top, all the computation occurs in the device itself or with the support of a tethered companion device nearby (such as a smartphone). In the “maximum offload” scenario at the bottom, all computation occurs remotely, in the cloud. Previously referred to as a “high offload” option [1], maximum offload has extreme requirements on connectivity and is not expected to reach mainstream adoption in the foreseeable future.

In the “offload: variable levels” scenario in the middle, some computation occurs in or near the device, while other computation occurs in the cloud. This split may be user-specific and possibly dynamic during the service period depending on the device

capabilities, cloud capabilities, network functionalities and conditions, and user preferences. This scenario corresponds to the “low offload” and “mid offload” options defined in a previous article [1] and represents the set of offload scenarios we expect will gain market traction.

Offloading as a mechanism implies data transmission between device and the processing unit on the network side, where the resulting bit rate and latency requirements vary depending on many factors.

In the case of remote rendering, the application graphics are rendered at a remote server (an edge server, for example) and sent to the HMD through a 5G (or other) network. Cloud gaming is a contemporary example of this. Remote rendering requires high downlink (DL) bitrate with bounded low delay to provide good QoE. Typically, two or more 2D video streams are sent in the DL. When foveated rendering is used, four 2D streams of

different resolutions will be sent to the HMD. Another version of remote rendering is to render volumetric content remotely and let the HMD make the final 2D rendering based on the most up-to-date pose data. In this case, HMD-local, latency-hiding techniques such as space warp may use the 3D data streams to enable the user to look at an object from other angles without requiring more data in the DL.

When offloading spatial computation functions, the HMD uploads compressed sensor data (video, lidar and so on) or detected objects to a server that builds a map of the environment and positions the HMD within it. Offloading SLAM enables shared immersive multi-user experiences, as the map of the environment can easily be shared between users in the remote server. These maps can make the positioning task of an HMD that is entering a previously mapped environment more efficient and allow for object persistence. There is a variety of offloading options for SLAM, and complete offload implies very stringent latency requirements. The uplink (UL) bitrate depends on the resolution and level of compression of the sensor data (or feature data) and can be large.

For object detection and tracking, a video feed is typically uploaded to a server for image recognition and tracking of previously identified objects. The location of the objects, and possibly other metadata, is sent to the HMD. Object tracking places stringent requirements on the latency if the function is offloaded. Offloading only object detection could be less sensitive to latency, as the corresponding information can be sent to the HMD when it becomes available. Then the latency-sensitive

object-tracking functionality on the HMD takes over to ensure that the visual information presented to the user stays aligned with the real world.

In cases where processing is done in the device and/or in a tethered companion device, the communication over the network is limited to application data, with characteristics that are similar to mobile broadband (MBB) traffic. The real-time processing power is mainly limited to what is available in the glasses and the companion device. The device capabilities are much more limited when running without a companion device with local processing only in the HMD. The processing in slim-form-factor AR glasses is expected to only be sufficient for basic SLAM and rendering, while more of the SLAM needs to be performed remotely for immersive and interactive experiences. For the cases with complete offload of all XR application processing functionality, the HMD will always have to run some processing locally, including overlay imaging, localization and latency-hiding processing.

In short, smaller XR device form factors, longer battery life and advanced XR services will require more cloud computing due to the increased need for computation offloading. The impact on network requirements will vary depending on which XR functions and/or their various functionalities need to be offloaded. It makes sense to offload XR functions that enable the highest device power savings or the greatest QoE improvements, as long as the connectivity requirement in terms of bitrates and latencies can be met by the available networks. Offloading will then be progressive as networks evolve and become more capable to meet the various

Key terms

Foveated rendering is a technique that uses eye-tracking to identify where a user is looking and render higher resolution images in those directions, while rendering lower resolution images for peripheral vision. The purpose is to save bandwidth and reduce the rendering complexity.

Object persistence means that a virtual object remains in the same location, even if the user moves out of the environment. A placed object could be stored in a server along with a precise position in a stored map. The map can be tagged with its location in the real world. The storing of detailed maps of scanned environments comes with privacy concerns and is a topic of ongoing research and regulation.

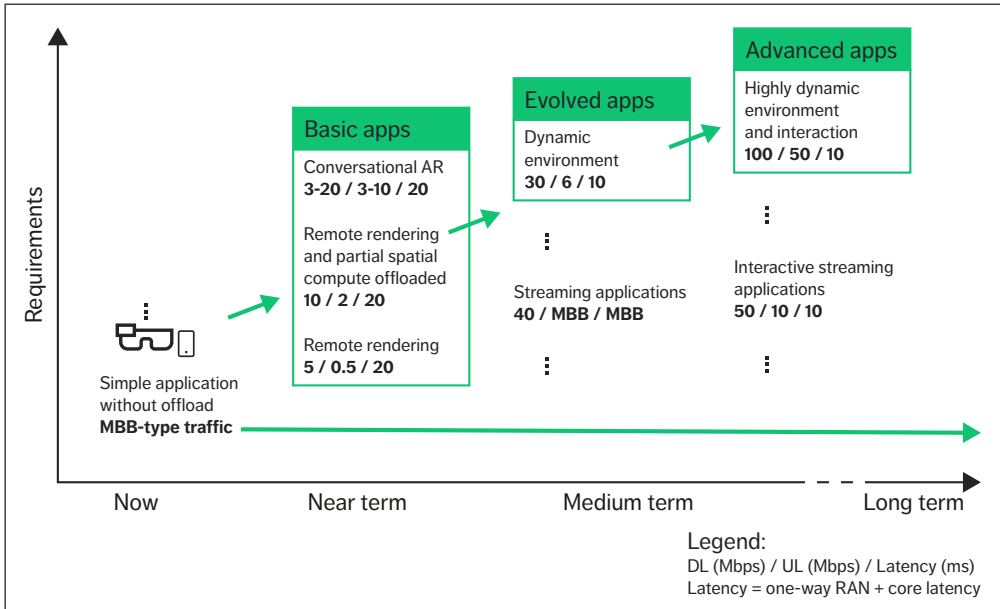


Figure 3 Evolution of requirements over time

and changing requirements of XR services and offloading levels.

Example operating points – XR flavors

A large multi-dimensional space results from the combination of a broad range of more and less advanced XR applications and various XR device capabilities that are associated with different levels of computation offloading. In theory, any point in this space is a possible XR flavor with its own requirements for network support. To simplify the analysis of required network solutions we narrow our focus to a smaller sample. The sample of XR flavors that we present here is based on concrete experiments with various partners [5, 6] and our own research and has been chosen to represent a potential evolution of slim AR glasses. The time dependency of the dimensions is a key consideration.

The XR flavors that we have chosen to analyze should be understood as a limited selection of samples in a wide range. They are examples rather

than forecasts, and the conclusions we draw cannot be directly extrapolated to all XR applications and all cases. The requirements that we have obtained from the different XR flavors are simplified and intended to provide general guidance. In reality, XR requirements will be very elastic in both DL and UL bitrates, as well as in latency, at least to some degree. Thus, it is expected that XR requirements will be defined by a range of values as opposed to a single value.

In Figure 3 we present the XR flavors along a time axis and a strictness-of-requirements axis. For each point in time – now, near term, medium term and long term – there are flavors with different characteristics, resulting in different requirements on the network. Increasingly advanced XR applications will become available over time, which will, together with the device capabilities, place increasingly stringent requirements on the connectivity. Still, at each time instance, a wide range of applications will be used, some of which will be very simple, while others will be more complex. The

THE CONVERSATIONAL AR FLAVOR THAT WE HAVE CONSIDERED IN THE NEAR TERM INVOLVES NO COMPUTATION OFFLOADING

bitrate requirements shown in Figure 3 are aggregated over the flows, whereas the one-way delay requirements are for the most sensitive flow.

Now – simple XR applications without offload

In the simplest XR applications, for which offload provides limited benefit, all processing is done locally on the HMD or on a tethered device. This is similar to how AR applications work on mobile phones today. The requirements are much like those of MBB, and there are no stringent requirements on latency. Example applications include simple navigation services and the ability to extend the touch-screen user interface of smartphones and tablets beyond the bounds of their physical screens with virtual screens controlled by hand gestures. Relevant parts of the global representation of the world are downloaded without real-time requirements. The global map used by these applications is updated on a relatively slow timescale. The time lapse before a newly-created AR object – a new restaurant in a navigation app or a user-added static virtual object in an AR geocaching app, for example – becomes visible for other users depends on how often the global map is updated based on user/actor input and how often the local map is refreshed from the global map.

Near term – basic XR apps

For simple XR applications with remote rendering, we assume that SLAM is run on the HMD. The remote-rendered content would be 2D video content or very simple volumetric content sent in the DL to the HMD. The DL bitrates vary depending on the resolution, frame rate and level of compression. The UL traffic consists of the HMD pose and user input,

which is sent to the remote server to ensure proper rendering. This category of devices would require a relatively low UL rate of 0.5Mbps. A slightly higher DL rate would be needed depending on the size and resolution of the virtual content created in the edge cloud. We chose a sample point of approximately 5Mbps. The one-way latency for such a use case would be estimated at around 20ms, including the radio-access network (RAN), the core and the transport network. In this flavor, exchanging anchor points between users enables the alignment of coordinate systems and local multi-user applications such as board games.

In simple XR applications that offload both rendering and partial spatial computation, localization occurs in the HMD, while spatial mapping, map optimization and object detection are offloaded. This means that 3D objects are rendered on the remote server, and 2D images for the displays are rendered in the HMD. This setup provides the HMD with 3D content and allows for local adjustments in the rendering to account for the user's head movements as late as possible in the rendering chain. The DL throughput depends on the compression and resolution of the 3D content. We use 10Mbps as an estimate of what could be realistic for low-resolution 3D content.

The UL data is composed of the HMD pose, user input and data representing the detected environment in the form of UL video and sensor data, for example. The UL bitrate depends on the resolution and frame rate of the video and sensor data. A more detailed representation of the detected environment requires better maps and more precise localization. Parts of the spatial mapping and remote rendering have an impact on latency requirements. The offloaded spatial computation functions of this flavor enable an increased understanding of the local environment.

The conversational AR flavor that we have considered in the near term involves no computation offloading. The requirements focus on sending a volumetric data stream generated by one device to receiving HMD(s). At one end, a person is continuously scanned and a volumetric data stream

is generated and sent to the receiving HMD(s) for display. In this scenario, we assume that the rendering from volumetric data to the HMD is performed on the HMD. The UL and DL bitrates depend on the resolution of the point cloud, the frame rate and the level of compression. A reasonable operating range in the near term would be up to 20Mbps in the DL and 10Mbps in the UL. The DL bitrate will depend on the number of participants in the conversation.

Medium term – evolved XR apps

The requirements to support more advanced applications will be more challenging, particularly in dynamic environments where AR objects must adapt to changes in the physical world in real time. One example would be to have a virtual AR character, such as a Pokémon, walk alongside a user on a sidewalk and avoid colliding with anyone or anything around it. Both remote rendering and the offload of spatial mapping will likely be required to achieve this. It will be essential to update the digital representation of the physical environment quickly enough for the application engine to respond with a timely and appropriate rendering of the AR object. The latency requirements in both the UL and DL for this flavor are much tighter than for previous ones, because the consequences of failing to detect and map certain parts of the environment with sufficient quality are greater. For example, the AR experience would be seriously undermined if an object in front of the Pokémon were left undetected and the Pokémon walked through it rather than jumping over it.

To avoid such error events, the representation of the detected environment needs to increase in size (in terms of higher sensor resolution and higher sensor frame rates, for example), which leads to an increase in the UL bitrate compared with previous flavors, based on the assumption that the content sent in the DL is remotely rendered, high-quality 2D video streams.

In the medium term, we also expect to see streaming applications evolve to use XR. For example, content-providing apps such as Netflix, TikTok and Snapchat may provide volumetric video

streams that offer more immersive 3D-viewing experiences such as the Star Wars movie scene where Princess Leia is projected by R2D2 as a hologram. In this flavor, SLAM is handled on the HMD, along with decoding to 2D. The DL requirements would vary depending on the video quality and frame rate. A point cloud stream of 40Mbps could represent two digital objects at a given resolution, for example. The UL bitrate and latency requirements for this flavor are similar to MBB traffic requirements.

Long term – advanced XR apps

In the long term, we expect that interactive streaming applications will have the ability to both stream volumetric video and enable user interactivity – that is, the user's behavior will affect the content that is streamed from a server. In this flavor, we assume that SLAM will be offloaded. Rendering of the volumetric video stream, based on user interactions, will be performed at a remote server. The DL bitrate will depend on the complexity, resolution and frame rate of the content. The UL will be dominated by the HMD sensor data that is needed to offload SLAM. Low latency will be required both for SLAM offload and for user interactions with the content.

In future advanced XR apps that support highly dynamic environments and interactions, AR objects will be expected to adapt to the dynamic environment, moving as if they were physically present in it. It will also be possible for multiple users to perceive and interact with the AR objects simultaneously. The AR object casts shadows, is occluded by physical objects and is shown correctly

●● THE AR OBJECT CASTS SHADOWS, IS OCCLUDED BY PHYSICAL OBJECTS AND IS SHOWN CORRECTLY AS A NATURAL PART OF THE PHYSICAL WORLD ●●

as a natural part of the physical world. In this flavor, we assume a dynamic multi-user application with several users in the same environment that interact with each other and the AR objects. The digital world is shared in the cloud.

This XR flavor will require the streaming of high-quality volumetric content in the DL, which could become very large. Alternatively, high resolution and high frame rate remote-rendered 2D video streams could be considered, with or without foveated rendering. To enable correct rendering and spatial understanding, large sets of high-resolution HMD sensor data would be sent in the UL to the cloud. Spatial mapping would be offloaded and correspond to roughly 10-20Mbps in the UL. In addition, high-resolution 3D point-cloud data would be sent in the UL to enable an enhanced experience. This could correspond to 30-40Mbps. Latency requirements would be as strict as those of the dynamic environments flavor in the medium term.

Conclusion

The use of extended reality (XR) technologies is expected to rise significantly in the years ahead, generating new network requirements with varying degrees of stringency depending on device capabilities, how much computation offloading is required, and which specific applications will be used. While these future requirements are heterogeneous and uncertain, they are essential inputs to the process of developing new network capabilities to support future XR use cases. To assist us in this work, we have defined a set of XR flavors to serve as samples in the expected requirements range. Our detailed analysis of these flavors demonstrates that they will require networks that can support bitrates of tens of Mbps together with latencies of 10-20ms to be achieved with high reliability – a task that is significantly more difficult than that of supporting today’s mobile broadband services.

References

1. **Ericsson Technology Review, XR and 5G: Extended reality at scale with time-critical communication, August 24, 2021, Alriksson, F et al., available at:** <https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/xr-and-5g-extended-reality-at-scale-with-time-critical-communication>
2. **Ericsson Technology Review, Holographic communication in 5G networks, May 17, 2022, Essaili, A E et al., available at:** <https://www.ericsson.com/en/reports-and-papers/ericsson-technology-review/articles/holographic-communication-in-5g-networks>
3. **Ericsson, 5G: The next wave, available at:** <https://www.ericsson.com/en/reports-and-papers/consumerlab/reports/5g-next-wave>
4. **Ericsson, Ten Hot Consumer Trends in 2030, available at:** <https://www.ericsson.com/en/reports-and-papers/consumerlab/reports/10-hot-consumer-trends-2030>
5. **Ericsson blog, Network performance and the metaverse: Can 5G deliver what's needed?, November 30, 2022, Kang, D H and Pradas, J L, available at:** <https://www.ericsson.com/en/blog/2022/11/network-performance-metaverse-5g>
6. **Ericsson case study, Building the future of entertainment with 5G, June 2022, available at:** <https://www.ericsson.com/en/cases/2022/5g-xr>

Further reading

- » **Ericsson blog, Welcome to XR immersive experiences, available at:** <https://www.ericsson.com/en/blog/2021/11/welcome-to-xr-immersive-experiences>
- » **Ericsson blog, How 5G and edge computing can enhance virtual reality, available at:** <https://www.ericsson.com/en/blog/2020/4/how-5g-and-edge-computing-can-enhance-virtual-reality>
- » **Ericsson, Immersive technology, available at:** <https://www.ericsson.com/en/5g/immersive-technologies>
- » **Ericsson, Imagine possible perspectives, available at:** <https://www.ericsson.com/en/about-us/new-world-of-possibilities/imagine-possible-perspectives/>
- » **Ericsson, Varjo&One Reality video, available at:** <https://www.ericsson.com/en/about-us/new-world-of-possibilities/imagine-possible-perspectives/videos/digital-twins/article01-vr-and-ar-devices-demo-video.mp4>

THE AUTHORS



Fredrik Alriksson

◆ is an expert in system architecture for the Internet of Things (IoT) and new services at Business Unit Networks, where he leads strategic technology and concept development related to the IoT and industry verticals. He joined Ericsson in 1999 and has worked in R&D with architecture evolution, covering a broad set of technology areas including RAN, Core and IP Multimedia Subsystem. Alriksson holds an M.Sc. in electrical engineering from KTH Royal Institute of Technology in Stockholm, Sweden.



Oskar Drugge

◆ is a system designer at Business Unit Networks. Since joining Ericsson in 2004, he has worked with a wide range of wireless technology development

from device side radio access algorithms to network architecture. His current focus is on concept development within the IoT and industry verticals. Drugge holds an M. Sc. in electrical engineering from Luleå University of Technology, Sweden.



Anders Furuskär

◆ joined Ericsson Research in 1997 and is currently a senior expert focusing on radio resource management and performance evaluation of wireless networks. He holds an M.Sc. in electrical engineering and a Ph.D. in radio communications systems, both from KTH Royal Institute of Technology.



Du Ho Kang

◆ is a senior specialist at Ericsson Research who joined the company in 2014. His expertise is in 5G-and-beyond concept

developments and performance evaluation toward diverse international standardization and spectrum regulation bodies. His current interest is developing future RAN concepts for emerging services. Kang holds a Ph.D. in wireless infrastructure and deployment from KTH Royal Institute of Technology.



Jonas Kronander

◆ has been with Ericsson Research since 2007. He is currently a research leader and manager for a section focusing on radio networks for machine-type communication. His current research interests include extended reality and its impact on radio networks. Kronander holds an M.Sc. in engineering physics and a Ph.D. in theoretical physics from Uppsala University, Sweden.

Jose Luis Pradas

◆ is a master researcher at Ericsson Research whose current focus is on RAN enhancements to support XR services in 5G networks. He joined Ericsson in 2007 and has worked in research,



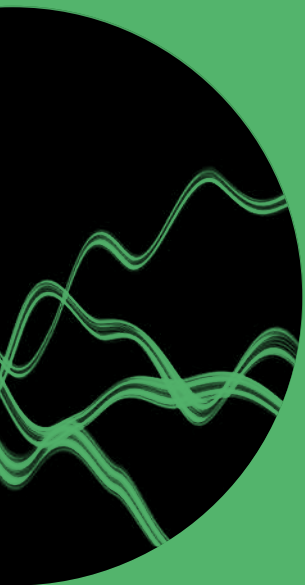
performing concept development in architecture and RAN protocols, as well as driving 3rd Generation Partnership Project standardization. Pradas hold an M.Sc. in telecommunication from the Universitat Politècnica de València, Spain, and an M.Sc. in communications from the Helsinki University of Technology, Finland.



Ying Sun

◆ is a senior specialist in scheduling. Since joining Ericsson in 2001, she has served as a system engineer, a research engineer and a local product manager. Her current focus is on concept and product development within time-critical communications. Ying holds an M. Sc. in electrical engineering from the Delft University of Technology, the Netherlands.

We would like to thank Håkan Olofsson, Göran Eneroth, Fredric Kronstedt and Yashar Nezami for their valuable contributions to this work.



ISSN 0014-0171
284 23-3391 | Uen

© Ericsson AB 2023
Ericsson
SE-164 83 Stockholm, Sweden
Phone: +46 10 719 0000