**5GPPP Technology Board Working Group**
**5G-IA's Trials Working Group**

# Edge Computing for 5G Networks

## Version 1.0

Date: 29-01-2021 Version: 1.0

DOI 10.5281/zenodo.3698117
URL https://doi.org/10.5281/zenodo.3698117

## Table of Contents

# Executive Summary

The EU-funded research projects under the 5G PPP initiative[1] started back in 2015, when the so-called Phase 1 of research activities was launched to provide the first 5G concepts. This was followed up with the second phase in 2017 where the first mechanisms were designed, and significant technological breakthroughs were achieved. Those projects posed the basis for the architecture and services of the 5G and beyond systems. With Phase 3[2] a new set of projects was launched in 2018, starting with the three Infrastructure projects, followed up with the three cross-border automotive projects, the advanced validation trials across multiple vertical industries and the projects dealing with the 5G longer term vision. 5G PPP is currently on boarding the latest projects, the latest of which are expected to start in January 2021 and deal with smart connectivity beyond 5G networks.

It is therefore a good time to review how 5GPPP projects have been using and enhancing Edge Computing for 5G and beyond systems, based on the information shared by the projects themselves. But before delving into that analysis, this whitepaper presents a rationale on why Edge Computing and 5G go hand by hand, and how the latter can benefit most from the former.

Section 1 of this whitepaper presents a brief intro to the Edge Computing concept with some perspective linking it to the explosion of data usage driven by other technologies like Artificial Intelligence (AI) and the relevance of Data Gravity. It also elaborates on how Edge Computing helps the 5G Value proposition. It then goes over Edge locations and how an Edge deployment could look like, to finalise with the Edge Cloud ecosystem introducing the roles of the main actors in the value chain.

Section 2 presents an exhaustive technology review of concepts with a 5G network perspective, focusing on four categories: virtualisation, orchestration, network control, and operational frameworks. As Edge Computing is always deployed within a wider communication system, this section presents several scenarios for connecting the Edge Computing to other technologies such as Cloud federation (connecting the Edge Cloud to other Clouds), End to End Slicing (where Edge Compute resources are part of some Network Slice), Radio Access Network (in particular the Open RAN model that can leverage Edge Computing resources), Inter Edge Border connectivity (to show how Edge resources can move between Home and Visited Networks), and finally the connection to Satellite Networks.

Section 3 analyses the role of Security in Edge Computing, reviewing key security threads and how they can be remediated, and how some 5G PPP projects have addressed these problems.

Section 4 presents the so-called Battle for Edge that many companies are currently fighting, trying to gain the best possible position in the ecosystem and value chain. It describes the different actors and roles for these companies, and then describes the

[1] https://5g-ppp.eu/

[2] https://5g-ppp.eu/5g-ppp-phase-3-projects/

"Coopetitive Landscape", analysing both scenarios where one actor can take the dominant role and other more collaborative scenarios.

These sections of the whitepaper provide the context on motivation on using Edge Computing for 5G, the technology and security landscape and the options for building an Ecosystem around Edge Computing for mobile networks, preparing the reader for the main section of the whitepaper.

Section 5 enters in the main focus of the whitepaper, describing 5GPPP projects approach to Edge Computing and 5G. This analysis has been based on 17 answers from Phase 2 and Phase 3 5GPPP projects to an Edge Computing Questionnaire created specifically for this whitepaper. The questionnaire asked about the type of infrastructure deployed, the location of the Edge used in the project, the main technologies used for these deployments, the Use Cases and Vertical Applications deployed at the Edge, and what drivers were used to select those. As the reader will see, Edge computing solutions have been extensively used by many 5G PPP projects and for diverse use cases. The analysis of the received answers provides some useful insight to the reader about the usefulness of Edge Computing in real networks.

We are confident that this whitepaper will be of interest for the whole 5G research community and will serve as a useful guideline and reference of best practises used by 5G PPP projects.

# 1. Introduction - Why Edge Computing is key for 5G and beyond

## 1.1 What is Edge Computing

There are many definitions for the term Edge Computing. The Linux Foundation has created an Open Glossary and under Edge Computing [3] one can read the following definition:

*The delivery of computing capabilities to the logical extremes of a network in order to improve the performance, operating cost and reliability of applications and services. By shortening the distance between devices and the cloud resources that serve them, and also reducing network hops, edge computing mitigates the latency and bandwidth constraints of today's Internet, ushering in new classes of applications. In practical terms, this means distributing new resources and software stacks along the path between today's centralized data centers and the increasingly large number of devices in the field, concentrated, in particular, but not exclusively, in close proximity to the last mile network, on both the infrastructure and device sides.*

So, Edge Computing reduces the distance between Users (Applications) and Services (Data). But the question remains: “Why has Edge Computing become such a popular technology trend during the past years?”

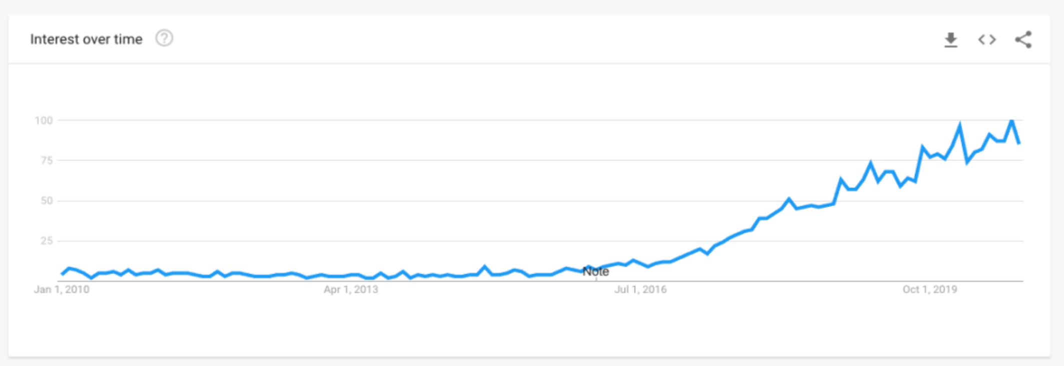
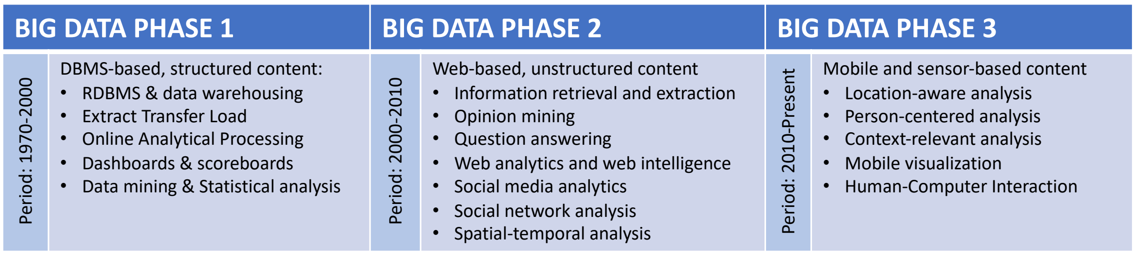


**Figure 1: Number of Searchers of “Edge Computing” from Google Trends**

We can explain this explosion of interest by looking at Big Data and AI evolution.



| BIG DATA PHASE 1 | | BIG DATA PHASE 2 | | BIG DATA PHASE 3 | |
|---|---|---|---|---|---|
| Period: 1970-2000 | DBMS-based, structured content:<br>• RDBMS & data warehousing<br>• Extract Transfer Load<br>• Online Analytical Processing<br>• Dashboards & scoreboards<br>• Data mining & Statistical analysis | Period: 2000-2010 | Web-based, unstructured content<br>• Information retrieval and extraction<br>• Opinion mining<br>• Question answering<br>• Web analytics and web intelligence<br>• Social media analytics<br>• Social network analysis<br>• Spatial-temporal analysis | Period: 2010-Present | Mobile and sensor-based content<br>• Location-aware analysis<br>• Person-centered analysis<br>• Context-relevant analysis<br>• Mobile visualization<br>• Human-Computer Interaction |

**Figure 2: Big Data major phases from the Enterprise Big Data Professional Guide**[4]

[3] https://github.com/lf-edge/glossary/blob/master/edge-glossary.md

[4] https://www.bigdataframework.org/short-history-of-big-data/

While the beginning of Big Data can be set in the 90s, it is really in the last decade that Data explosion took place.
The application of AI to Big Data increased the need for larger Data sets to train inference models. Public cloud has played an instrumental role in this space, but the more the data set grows, the more difficult is to move the data.

That´s why Dave McCrory in 2010 introduced the concept of "Data Gravity"[5]. The idea is that data and applications are attracted to each other, similar to the attraction between objects as explained by the Law of Gravity.

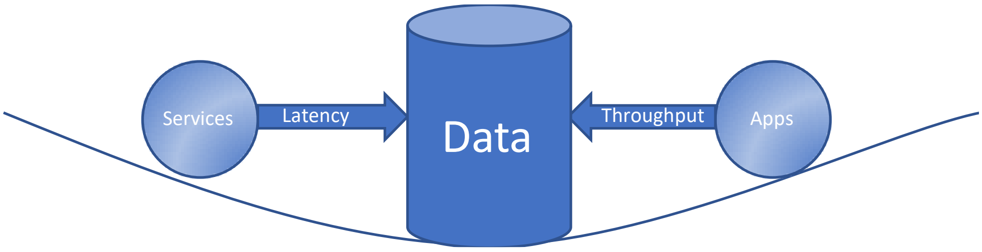


**Figure 3: The Data Gravity concept introduced in 2010 by Dave McCrory**

In mobile networks, Applications (Apps) run in smartphones, whereas Services run in the Operator´s Core Network (IMS Services) or in Internet (commonly in Public clouds). Apps and Services are therefore very 'far away' from each other as perceived from a time point of view (e.g., typically more than 50-100 ms). This is because exchanged data have to travel through a set of networking entities and devices (e.g., aggregation points, IP routers, Peering routers, Interconnection hubs). It is not uncommon that the links to these devices can get congested, and therefore it is impossible to guarantee any end-to-end Quality of Service (QoS) or throughput.

In such an environment Edge Computing plays a key role as the enabling technology to shorten the distance between Users (Apps) and Services (Data) and enable guaranteed Latencies and Throughputs, as required by services and applications. These requirements have become apparent especially with the digitization of Verticals such as Industry 4.0, Collaborative and Automated Driving, E-Health etc.[6]

## 1.2 Why is Edge Computing critical for 5G?

The 5G Network is the most recent Mobile Network generation defined by 3GPP. Looking back at the evolution of Mobile Networks, before the introduction of a new generations it has always been a problem to predict which use cases would have been the ones mostly valued by Users:

- 3G Networks were designed mainly for Voice (Circuit Switched) and limited Internet browsing. However, Smartphones appearance in 2007 revealed Apps as the main use case: people used to spend 90% of their mobile usage time with Apps[7].

[5] https://datagravitas.com/2010/12/07/data-gravity-in-the-clouds/

[6] 5G PPP, White paper, "Empowering Vertical Industries, Through 5G Networks", https://5g-ppp.eu/wp-content/uploads/2020/09/5GPPP-VerticalsWhitePaper-2020-Final.pdf

[7] https://buildfire.com/app-statistics/

- 4G Networks were designed for Data services, modelling Voice service as Data (VoLTE), while most of the traffic in 4G Networks is Video (Video will represent 82% of all IP traffic in 2021)[8].

If the Telco Industry would have known that Video was to account for 80% of traffic, most probably the design of 4G Networks would have been different, e.g., introducing Content Delivery Network (CDN) in the architecture.

The reality is that it is impossible to predict how users are going to drive the usage of newly introduced mobile networks. Therefore, for 5G Networks, 3GPP has taken a Service Oriented approach, introducing new key concepts, such as Network Slicing, or a Service Bus Architecture for Microservices, to offer the possibility to create a Virtual Network for a specific Service to deliver the best user experience to customers.

The 5G Network value proposition relies on three pillars or capabilities, usually displayed like in Figure 4, associated to most relevant use cases:

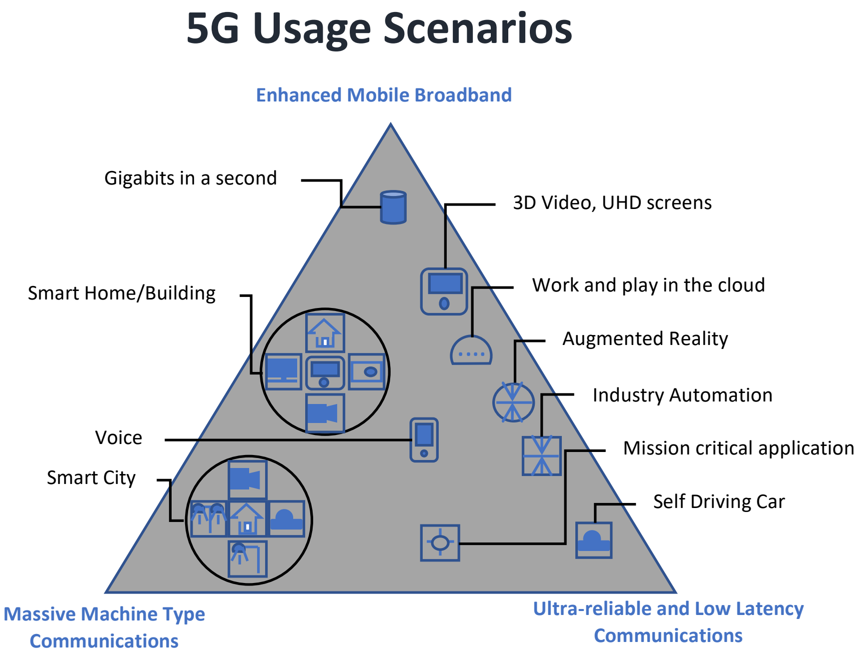

5G Usage Scenarios
Enhanced Mobile Broadband
Gigabits in a second
3D Video, UHD screens
Smart Home/Building
Work and play in the cloud
Augmented Reality
Industry Automation
Voice
Mission critical application
Smart City
Self Driving Car
Massive Machine Type Communications
Ultra-reliable and Low Latency Communications

**Figure 4: 5G Usage Scenarios (Source International Telecommunications Union[9])**

- **Enhanced Mobile Broadband (eMBB):** aims to service more densely populated metropolitan centers with downlink speeds approaching 1 Gbps (gigabits-per-second) indoors, and 300 Mbps (megabits-per-second) outdoors.
- **Ultra-Reliable and Low Latency Communications (URLLC):** addresses critical communications where bandwidth is not quite as important as speed - specifically, an end-to-end (E2E) latency of 1 ms or less.

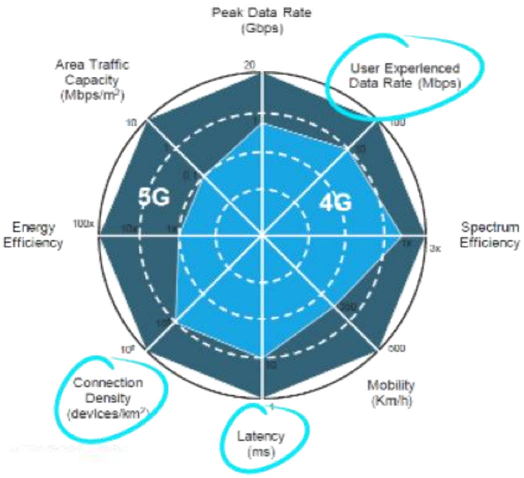
[8] https://www.businessinsider.com/heres-how-much-ip-traffic-will-be-video-by-2021-2017-6?IR=T

[9] https://www.itu.int/en/ITU-D/Conferences/GSR/Documents/GSR2017/IMT2020%20roadmap%20GSR17%20V1%202017-06-21.pdf

- **Massive Machine Type Communications (mMTC):** 5G enables an 1000X increase of devices connected to the Network, moving from 1K devices per Km2 in 4G to 1M devices in 5G[10].

In order to deliver the above mentioned above value proposition, Edge Computing plays a fundamental role, as Compute resources are critical to enable those three capabilities to the Network, so to be able to finally deliver a satisfactory E2E experience.

Figure 5 elaborates on what the main enhancements to some key system capabilities are, when moving from a 4G network to a 5G one.



**eMBB**: increasing Data transfer in Radio interface is not enough. Content needs to be closer to customers in order to sustain high data transfers rate with no congestion.
**URLLC**: reducing Latency in Radio interface is not enough. We need to move Services closer to customers in order to deliver a reduced and guaranteed E2E Latency.
**mMTC**: increasing the number of connected devices to the network needs to be accompanied by processing the signalling and data from these devices at the edge of the network to digest the volumes of information generated by a huge number of Things connected to the network.

**Figure 5: 5G capabilities vs. 4G capabilities (ITU-R[11])**

Moving content, services and signalling processing closer to customers requires moving compute resources closer to the devices consuming the content, running the Apps, or sending signalling coming from sensors. That is where Edge Computing not only meets 5G, but allows it to fully deliver its promised enhancements: 5G cannot be conceived just as a set of focused technical enhancements, e.g., a new radio technology, but also as a completely new paradigm for Mobile Networks, where Edge Computing plays a significant role.

## 1.3 Where is the Edge of the Network

There is no unique location, or range of locations, where Edge Computing must be deployed. Edge nodes can be included in network routers, cell or radio towers, WiFi hot spots, DSL-boxes, and local data centers. As described in Section 1.1, Edge Computing

[10] Massive Machine-Type Communications: An Overview and Perspectives Towards 5G (https://www.fpz.unizg.hr/ikp/upload/RCITD_2015_Massive%20Machine%20Type%20Communications%20An%20Overview%20and%20Perspectives%20Towards%205G.pdf)

[11] https://www.itu.int/en/ITU-T/Workshops-and-Seminars/standardization/20170402/Documents/S2_4.%20Presentation_IMT%202020%20Requirements-how%20developing%20countries%20can%20cope.pdf

is the concept of placing computing resources closer to users´ locations. Almost any device with computational power that is near or at the user's location can act as an Edge Computing device, as long as it can process a computational workload.

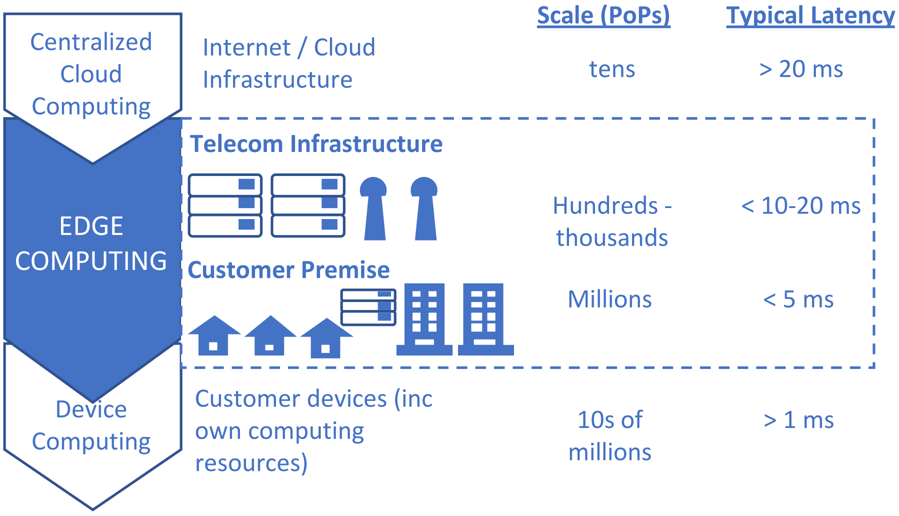


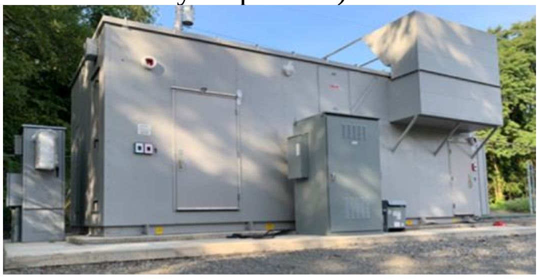
**Figure 6: Edge Computing Location**

Edge Computing is typically placed between users´ Devices and Centralized computing datacenters whether they are Public clouds or Telco Cloud facilities.

Device computing resources are hard to manage because of their heterogeneity and the network environment where they are connected to (typically LAN environments).

We can mention several Edge Computing deployment examples that help us to identify different Edge Computing Locations:

- **On Premise:** Companies deploying 4G/5G Private Networks deploy a full Network Core in the premise infrastructure connected to business applications[12]
- **RAN/Base station:** some companies are deploying infrastructure collocated with RAN in the streets, using Cabinets / MiniDatacenters (e.g., see Figure 7 5GCity/Vapor.io[13])



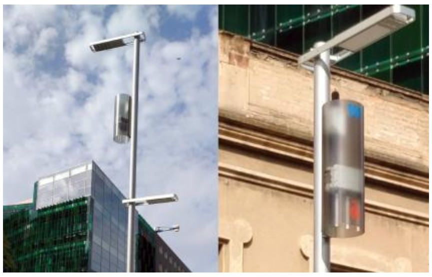


**Figure 7: Vapor.io Edge module and 5GCity Multifunctional Post**

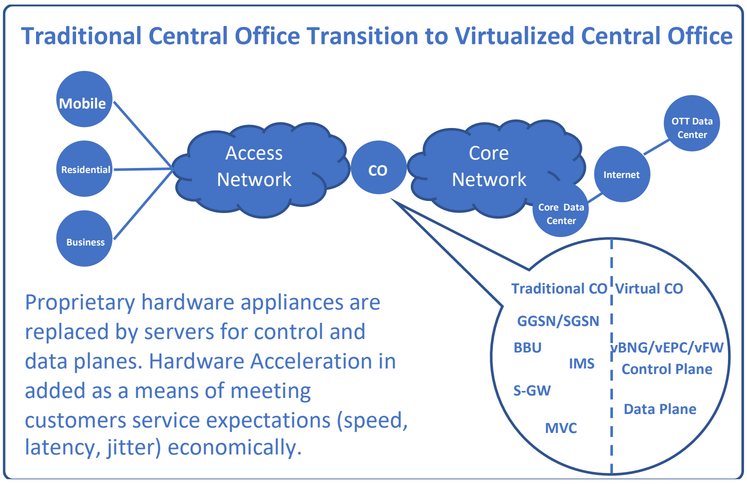
- **Central Offices (COs):** COs are at the Cloud Service Provider (CSP) network edge, which serves as the aggregation point for fixed and mobile traffic to and from end user. All traffic is aggregated to the CO, which creates a bottleneck

[12] https://www.daimler.com/innovation/production/factory-56.html

[13] https://www.vapor.io/36-kinetic-edge-cities-by-2021/

that can cause a variety of problems. Throughput and latency suffer greatly in the traditional access network, essentially cancelling out much of the gain from technologies such as optical line transfer (OLT) and fiber-to-the-home (FTTH), and 5G networks.

To address this issue an ongoing transformation has been initiated. A promising solution is to deploy a virtualized, distributed network at the Edge. Central Office Re-Architected as a Datacenter by CORD[14] and followed by OPNFV[15] and other projects, have started a process where the economies of a data center and the agility of Software Defined Network (SDN) applied with cloud design and network disaggregation principles will tackle the aforementioned problems.



Proprietary hardware appliances are replaced by servers for control and data planes. Hardware Acceleration in added as a means of meeting customers service expectations (speed, latency, jitter) economically.

**Figure 8: Virtualization of the CO principles: Cloud and Network Disaggregation**

- **Private Datacenters:** Telcos and other companies are deploying Private Datacenters to host Edge Computing infrastructure. This approach requires these Datacenters to be interconnected with Mobile Network Aggregation Point of Presences (POP) to get traffic from users.
- **Hyperscalers Edge Locations:** Public cloud companies define their own Edge locations. The AWS Edge solution is called AWS Cloudfront, and is typically deployed in one or two physical points per country in Europe[16]. The Azure solution for Edge is Azure CDN, mainly for content distribution, and is similarly distributed as the AWS Cloudfront[17].

While Edge can be located in different locations, they are not exclusive, and there can be several Edge locations used in a network deployment.

The term Fog Computing as defined by the National Institute of Standards and Technology[18], states that Fog Computing is a layered model for enabling ubiquitous

[14] https://www.opennetworking.org/cord/

[15] https://www.opnfv.org/wp-content/uploads/sites/12/2017/09/OPNFV_VCO_Oct17.pdf

[16] https://aws.amazon.com/cloudfront/features/

[17] http://map-cdn.buildazure.com

[18] https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-325.pdf

access to a shared continuum of scalable computing resources. The model facilitates the deployment of distributed, latency-aware applications and services, and consists of fog nodes (physical or virtual), residing between smart end-devices and centralized (cloud) services.

## 1.4 How does the Edge look like?

An Edge Computing infrastructure may be implemented in many different ways, depending on several parameters. It can go from a Raspberry Pi device to a several racks Datacenter footprint. Different Industry initiatives such as ONF, Broadband Forum and OPNFV have come up with similar architectures for Edge Computing infrastructure to be deployed at a CO level. The ONF design is called CORD, the Broadband Forum (BBF) design is called Cloud CO, and the OPNFV, from the Linux Foundation, is called Virtual CO.

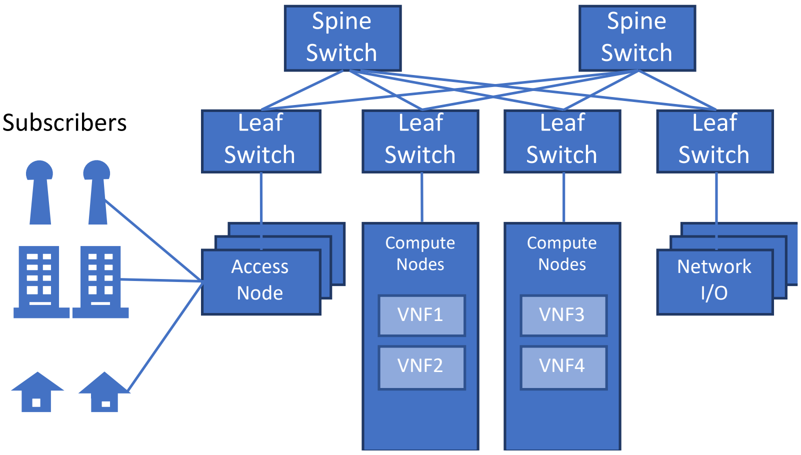


**Figure 9: ONF CORD & Broadband Forum Architecture**

In these architectures, the Edge Computing infrastructure is composed of:

- **Compute nodes:** these are the servers where Compute loads are executed.
- **Switching Fabric/SDN:** switching infrastructure in Leaf-Spine configuration (any leaf is connected to two Spines) managed by a SDN controller. Internal leaves act as Top of the Rack switches for severs to connect them.
- **Access Network:** whether it is a fixed or a mobile access network, connected to one border leaf of the switching fabric
- **Transport Network:** connected to the opposite leaf of the switching fabric.

These solutions are typically designed for full 42Us racks. Smaller footprint solutions are recently available from open organizations like the Open Compute Project, where the Open Edge project has released the Open Edge specifications with 2U and 3U form factors[19].

[19] http://files.opencompute.org/oc/public.php?service=files&t=32e6b8ffca7e964ec65de17ec435a9fc&download

## 1.5 Introduction to the 5G Edge Cloud Ecosystem

The previous sections provided an overview of the motivation, technologies, high-level architectures and deployment aspects that will drive the further development and evolution of the industry and markets for Edge Computing.

Looking at the evolution of Public cloud solutions and Services, they have been driven by a few actors that have grown into big global players, now often called over-the-top (OTT), providing services "on top of" today's Internet, or hyperscalers. The latter referring to their capability of seamlessly provision and adding compute, memory, networking, and storage resources to their infrastructure and make those available as scalability properties of the services offered. In addition, local IT and cloud providers have provided more tailor-made solutions and services that have properties and added value beyond commodity services.

With the emergence of Edge Cloud Services (leveraging Edge Computing technologies and solutions) we anticipate a richer set of actors entering the market, at the same time competing and collaborating. The illustration below identifies this wider set of players.

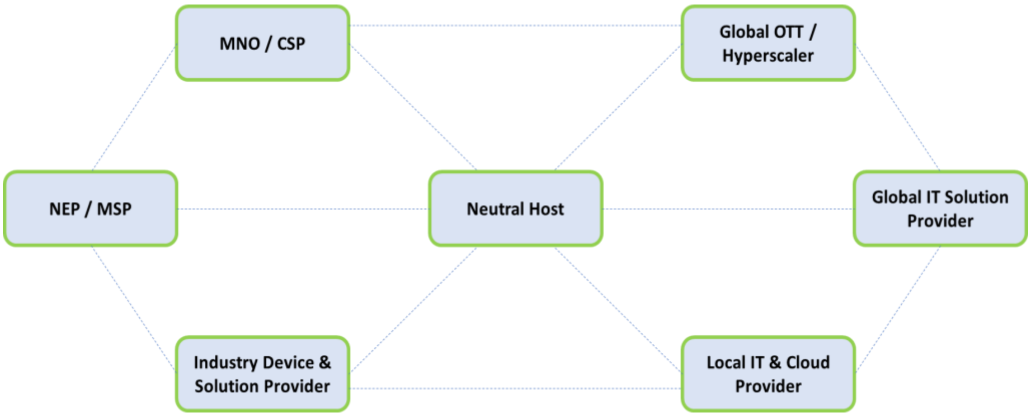

MNO / CSP
Global OTT / Hyperscaler
NEP / MSP
Neutral Host
Global IT Solution Provider
Industry Device & Solution Provider
Local IT & Cloud Provider

**Figure 10: Key players in edge cloud competitive / collaborative landscape**

In such "coopetitive" (cooperative and competitive) landscape we can identify the Global OTT or Hyperscalers and the Local IT & Cloud Providers. On the same side we can also highlight the Global IT Solution Providers such as IBM, Oracle, HPE, etc. On the other side we have Telecom Operators, e.g., Telcos/Mobile Network Operators (MNOs), and Communication Service Providers (CSPs). Moreover, telco vendors, e.g., Network Equipment Provider (NEP), are increasingly also offering managed services, thus acting as Manage Service Providers (MSP). With 5G and network capabilities addressing various Industry 4.0 use cases, the global industry device & solution providers (e.g. Siemens, Bosch, ABB, etc.) will as well address the Edge Computing and Edge Cloud Services space.

In the midst of these actors, we also point out the so-called Neutral Host (provider), potentially managing local or private spectrum and offering services to allow physical or virtual assets to be shared by multiple service-providers, and in this way improving the economic efficiency at locations where other actors acting individually do not see an effective business case.

To introduce some example configurations, functional roles and potential actor positions the following illustration is provided.

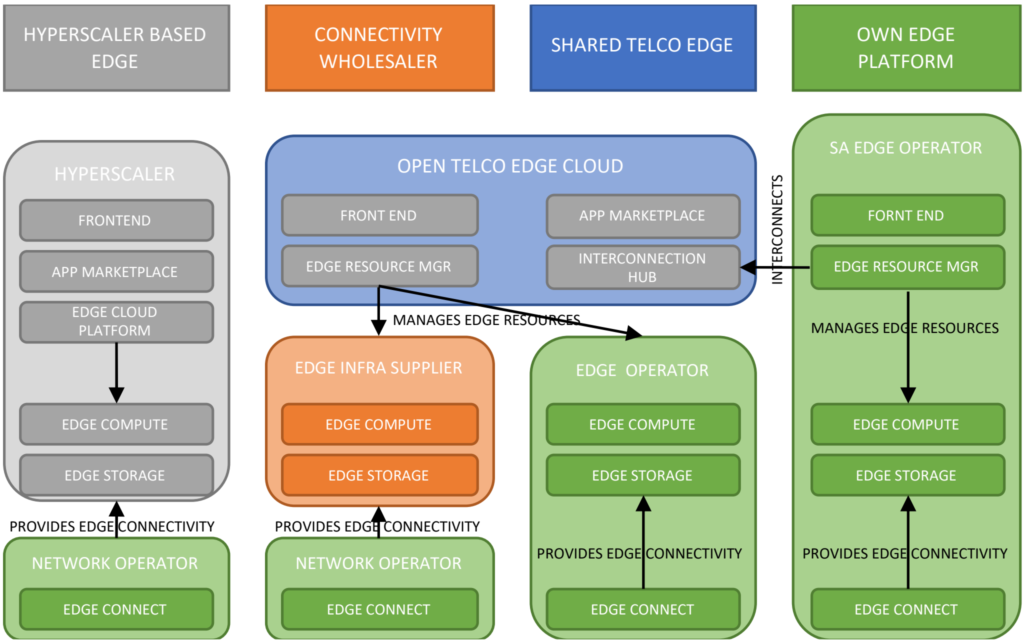


**Figure 11: Example configurations, functional roles and potential actor positions for Edge Cloud**

The “Hyperscaler-based edge” to the left shows how an hyperscaler provides most of the Edge Cloud stack, just building its infrastructure on top of the transport network infrastructure offered by a network operator.

The next three example stacks show various configurations where the Telcos play a significant role.

In the “Connectivity Wholesaler” case the Telco wholesale role is adapted to become a provider of Edge infrastructure such as Edge Cloud datacentre resources. On top of this a layer of Open Telco Edge Cloud (OTEC) capabilities is provided.

Both stacks in the middle of the Figure 11 can be considered as various ways for Telcos to share Edge Cloud resources. The Edge Cloud capabilities can be offered to the vertical enterprise customer by a specialized services provider.

Finally, the right-most stack shows an individual Telco providing the full stack by itself.

# 2. Key Technologies for 5G on Edge Computing

As discussed in the previous chapter, Edge Computing can be seen as an open platform where the core capabilities of networks, computing, storage, and applications converge. It provides intelligent services at the network Edge near the source of the objects or data to meet the critical requirements of real-time services, data optimization, application intelligence, security and privacy protection of industry digitization. To address these requirements, the frameworks of virtualization, orchestration, networking and operations should be designed and adapted to the distributed nature of the Edge services and applications, the ephemerality of data generated and the scaling needs.

In this section, we introduce some key technologies into four areas: the virtualisation, the orchestration, the network control and operational frameworks. In the last sub-section, we introduce some typical Edge Apps and Services through the description of some Edge Connectivity scenarios.

## 2.1 Resources Virtualization framework

The rise of Edge Computing has brought about a shift in system architecture requirements and considerations. As applications demand lower latency and reduced bandwidth, deployment method decisions are increasingly critical. This section examines the differences between using virtual machines (VMs) vs. containers vs. serverless functions in the context of Edge Computing.

### 2.1.1 Virtual Machines and Containerization

Microservices is a powerful architectural design pattern where the system is composed of small granularity, highly cohesive and loosely coupled services. Each of these services fulfil a specific functionality and is self-contained. Interactions between services implement standard light-weight interfaces (e.g., RESTful principles[20], etc.). At the service granularity level, microservices are small, i.e., they contain typically one, two or three modules focusing on one purpose. They have a bounded context where the service components are bounded to own data and to own implementation. Following the Cloud-native Computing Foundation's technologies[21], it appears that the containerization is one of the pillars of this transformation based on micro-services.

According to Docker, a container is a unit of software that packages up code, library and all its dependencies so the Apps run quickly and reliably from one computing environment to another[22]. A developer should create from scratch or starting from another container image, a standalone and lightweight package of software containing the operating system environment, libraries, tools, configurations and code needed for running the specific service. All these lines of code should be compiled and packed up in a Docker container image.

The advantages of a container software package against microservices architecture as containerized network functions are various, but most important it provides isolated

[20] https://ninenines.eu/docs/en/cowboy/2.3/guide/rest_principles/#_rest_architecture

[21] https://www.cncf.io/

22 https://www.docker.com/resources/what-container

environments for running software services and security by design. Containers can provide better service agility, performance, time to run, quick deployments and updates, scaling when necessarily, portability and better security[23].

Service agility and performance of a software container are put in place by the possibility to run directly on host. A software container runs on different namespaces or different parts of namespaces, the only things that are shared being some kernel features which are not completely isolated. Regarding resources, it does not use quota management resources, being protected from the noisy neighbour problem that is present in virtual infrastructures with VMs in place.

Another good characteristic of containers related to production infrastructure is their operational model that is easy to implement and work with, thanks to engines and resources managers with all built-in functions such as: scaling (up or down) according to deployment needs and self-healing, which takes action every time a container is not responding or crashed and service or VNF is partially or totally unavailable.

Many types of containerization technologies are available, for instance:

- **Docker containers** as mentioned earlier
- **Java containers:** those types of software packages enable standalone functioning of Java applications or parts of them. Examples: Springboot, Jetty, Tomcat.
- **LXD containers:** represent Linux Containers software technology that is very similar to various Linux distributions. These are created by Canonical Ltd. and are integrated with the OpenNebula EDGE platform.
- **OpenVZ containers:** Open Virtuozzo[24] is a dedicated container-based virtualization technology specially created for Linux operating systems.
- **RKT containers:** rocket containers and rkt container engine developed by CoreOS for the majority of Linux distributions in a cloud-native environment. This type of container is composed of a pod (like in the Kubernetes model and concept) with one or more applications inside.
- **Hyper-V containers:** they constitute a different type of containers because they create their own copy of the Windows OS kernel and are completely isolated, having incorporated both kernel space and user modes. They could be easily associated with a VM.

### 2.1.2 Lightweight virtualization

Unikernel is an alternative to both VMs and containers for lightweight virtualization of resources that has gained attention over the last few years. It emerged due to the idea that the majority of the functions running either in the cloud or at the Edge do not require many of the services inherent to OSs, and thus those services can be excluded. Unikernels are single-purpose appliances that are specialized at compile time into standalone kernels[25]. They are constructed with the minimal necessary libraries, modularly, compiled

[23] An Analysis of Container-based Platforms for NFV: https://www.ietf.org/proceedings/94/slides/slides-94-nfvrg-11.pdf

[24] https://openvz.org/

[25] A. Madhavapeddy et al., “Unikernels: Library Operating Systems for the Cloud,” ACM SIGPLAN Notices, vol. 48, no. 4. 2013, pp. 461–72.

together with the application code into an image (no division between kernel and user spaces) that can be run on top of a hypervisor or directly on a hardware layer. Different library OSs (e.g., IncludeOS, UKL, MirageOS, OSv, Rumprun, runtime.js) can be used to develop unikernels, with slightly different security profiles, programming languages (some of them aiming to avoid programming directly in C), and legacy compatibility.

Among other advantages, unikernels improve security over other virtualization paradigms since (i) they have no other functions/ports apart from the specific application they were built for, thus the attack surface is minimal, and (ii) they achieve a degree of isolation similar to VMs and much higher than containers, since the latter share a common kernel. Besides, due to their specialization, unikernels come with the benefit of faster boot times and lower images size than containers, as well as similar degree of memory consumption when running.

Still, unikernels have some drawbacks that come mainly from their immaturity. The most critical one is related to the high development times, as (i) kernel functionalities have to be carefully selected and configured for the specific application, (ii) there is a lack of tools designed for debugging unikernels, and (iii) to be updated they have to be shut down, updated, recompiled and instantiated, a set of operations that is not possible to run on the fly. Besides, their performance shows room for improvement, as initial tests have shown that time for (some particular) processes completion is higher in unikernels due to lower efficiency of memory management and hypervisor overhead[26]. This technology is more powerful in applications with high context switching between kernel and user spaces[27].

The nature of unikernels make them suitable for deploying stateless, high-response low-latency VNFs located at Edge nodes. General algorithms (e.g., compression, encryption, data aggregation) and specific functions for Vehicular Edge Computing (VEC), Edge Computing for smart cities and Augmented Reality (AR)[28] are use cases in which unikernels can be of utility. The UNICORE project[29], which aims at providing a toolchain for facilitating the development of secure, portable, scalable, lightweight and high-performance unikernels, foresees their potential application in 5G-RAN, vCPE and serverless computing, among other fields. As current Virtualized Infrastructure Managers (VIMs) support unikernels, some H2020 5G-PPP projects (such as 5G-MEDIA[30], 5GCity[31], Superfluidity[32], 5G-Complete[33], etc.) are using them jointly with VMs and

[26] R. Behravesh, E. Coronado and R. Riggio, "Performance Evaluation on Virtualization Technologies for NFV Deployment in 5G Networks," 2019 IEEE Conference on Network Softwarization (NetSoft), Paris, France, 2019, pp. 24-29.

[27] T. Goethals, M. Sebrechts, A. Atrey, B. Volckaert and F. De Turck, "Unikernels vs Containers: An In-Depth Benchmarking Study in the Context of Microservice Applications," 2018 IEEE 8th International Symposium on Cloud and Service Computing (SC2), Paris, 2018, pp. 1-8

[28] R. Morabito, V. Cozzolino, A. Y. Ding, N. Beijar and J. Ott, "Consolidate IoT EDGE Computing with Lightweight Virtualization," in IEEE Network, vol. 32, no. 1, pp. 102-111, Jan.-Feb. 2018.

[29] http://unicore-project.eu

[30] http://www.5gmedia.eu

[31] https://www.5gcity.eu

[32] http://superfluidity.eu

[33] https://5gcomplete.eu

containers within their 5G deployments, being leveraged in tandem for conforming services thus benefiting from their respective advantages.

On the other hand, serverless computing is a paradigm for virtualized environments that appeared during the past decade and has attracted great interest among services customers and providers. In this paradigm, developers have to focus on writing the code of their applications as a set of stateless event-triggered functions, in a Function-as-a-Service (FaaS) model, without having to manage aspects related to infrastructure (e.g. resource allocation, placement, scaling) since the platform is in control of those tasks. Despite the fact of being a novel concept, most major vendors have a FaaS offering, AWS Lambda being one of the most popular one. Still, there are different open source solutions for developing a serverless computing platform based on Kubernetes cluster on any public/private cloud or bare metal. Among them, one can find solutions such as Apache OpenWhisk[34], OpenLambda[35], Knative[36], Kubeless[37], Fission[38] and OpenFaaS[39]. Apart from the computing service, serverless architectures usually require other services like data storage or Application Programming Interface (API) gateways to be functional.

The advantages of serverless computing can be summarized in three aspects:

(i) increase of resource efficiency, as these are allocated/deallocated and scaled up/down depending on actual demand, thus getting rid of both idling and over-provisioned resources,

(ii) simplification of deployment and auto-scaling, and

(iii) decrease of development times, since developers do not have to manage infrastructure aspects.

However, as other virtualization paradigms, serverless computing is not without drawbacks. The time needed for the underlying virtualized environment (usually a container) to be allocated before running a triggered function is one of the most constraining ones. Other aspects, such as the increase of attack surfaces (vulnerabilities), potential need of an external state and increased integration testing complexity[40] have to be taken into account as well.

Edge Computing can benefit from some of the aspects provided by serverless paradigm, although it may not be an optimal choice for some services of the virtualized networking domain such as packet flow management or firewalls[41], since the required start-up latencies can affect their overall performance. An option to minimize this drawback is to make use of unikernels as underlying runtime engines, but as aforementioned, this technology is still immature and most serverless architectures work now with containers. In any case, serverless computing can be considered at Edge nodes for performing anomaly detection or data processing services. ETSI foresees its utility for 5G mMTC in

[34] https://openwhisk.apache.org

[35] https://github.com/open-lambda/open-lambda/blob/master/README.md

[36] https://knative.dev

[37] https://kubeless.io

[38] https://fission.io

[39] https://www.openfaas.com

40 Kratzke, N. A Brief History of Cloud Application Architectures. Appl. Sci. 2018, 8, 1368.

41 P. Aditya et al., "Will Serverless Computing Revolutionize NFV?," in Proceedings of the IEEE, vol. 107, no. 4, pp. 667-678, April 2019

MEC deployments[42], and the 5G-PPP 5G-MEDIA project has adopted this paradigm for developing VNFs for immersive media, remote and smart media production in broadcasting and CDN use-cases. We remind here an important distinction between Edge Computing and MEC: Edge Computing is a concept, and MEC is an ETSI standard architecture.

Typical architectures of VMs, containers and unikernels are depicted in Figure 12. Serverless functions would leverage these architectures, transparently to end users, although in the case of unikernels the provider should bake the function code with the minimal required OS services and then deploy the resulting unikernel on top of a hypervisor. It should be mentioned that depending on the type of hypervisor, they can work either with or without an underlying host OS.

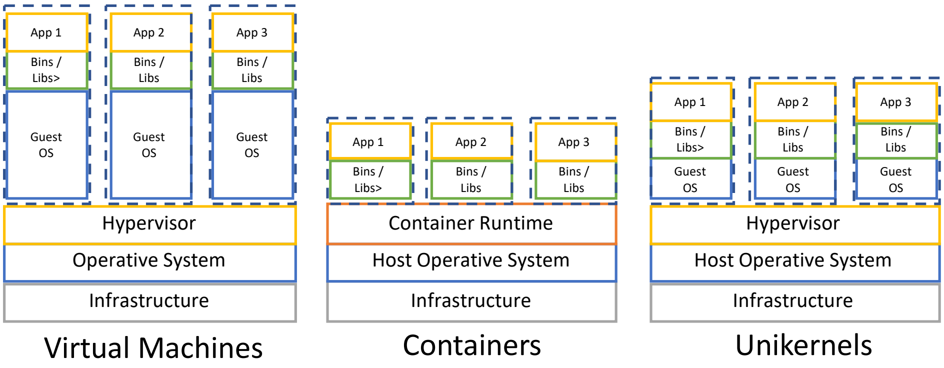


**Figure 12: Comparison of VMs, containers and unikernels system architectures**

## 2.2 Orchestration framework

Cloud technology is moving towards more distribution across multi-clouds and the inclusion of various devices and heterogeneous infrastructures. Virtualisation as the key enabling technology of cloud consists in abstracting the infrastructure hardware resources to run multiple independent instances of an application. Different virtualisation techniques exist today to implement such abstraction: hypervisors, containers and unikernels, as described in the previous section.

The orchestration framework for VMs, containers and also hybrid platforms are on huge demand. Many Platform as a Service (PaaS) use Docker, some have their own container foundation for running platform tools and provide orchestration. Three different PaaS generations can be distinguished:

- The first one was composed of fixed proprietary platforms such as Azure[43] or Heroku[44].
- The second one was made of open-source solutions such as Cloud Foundry[45] or

[42] https://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp28_mec_in_5G_FINAL.pdf

[43] Microsoft Azure, https://azure.microsoft.com

[44] Heroku, https://www.heroku.com

[45] Cloud Foundry, https://www.cloudfoundry.org

OpenShift[46] that allow users to run their own PaaS (on-premise or in the cloud).

- The current third generation of PaaS includes platforms like Flynn, and Tsuru[47], which are built on Docker from scratch and are deployable on own servers or on public IaaS clouds.

In the following we introduce the three main orchestration platforms for both VMs and/or Containers suitable to Edge domain.

### 2.2.1 Kubernetes

Over the last few years Kubernetes[48] (noted as K8s) has become a de facto standard for container orchestration. An important thing to recognize about Kubernetes is that it is a very smart intent-based orchestration engine, a fact that is overlooked by the current standard approach named Management and Network Orchestration (MANO), which treats Kubernetes as “dumb” NFV Infrastructure (NFVI). Essentially, the common approach is to provide a Kubernetes VIM that is used by an orchestration engine “brain” to interact with Kubernetes. A short-term advantage of this approach is clear: providing a low effort standard way of integrating existing MANO frameworks with Kubernetes. However, the long-term advantages of this approach are much less clear.

First, insulating developers and operators from Kubernetes Native Infrastructure (KNI) prevents them from acquiring cloud-native skills and state of mind, which are required to drive innovation in the telecom industry. As container transformation unfolds in the telecom industry, VM based VNFs give way to Container Network Functions (CNFs). These are a natural fit for Kubernetes based orchestration. In fact, CNFs are the primary motivation for shifting the management and orchestration plane centre of gravity to Kubernetes itself. However, it should be noted that by virtue of the Custom Resource Definition (CRD) mechanism, non-Kubernetes resources can be easily added to the Kubernetes ecosystem. Thus, a control and management plane grounded in Kubernetes can orchestrate not just containers, but also resources in other NFVIs (VMs and PNFs alike). At the same time, it is straightforward to reuse legacy orchestration, such as Heat templates, triggering them from Kubernetes.

Second, important Kubernetes projects, such as KubeVirt[49] are poised to disrupt VM based NFVIs and attract VNF migration to Kubernetes. While currently KubeVirt might not be a mainstream option (as of today, we are aware about only a handful of large scale KubeVirt deployments), this technology should be considered by MANO, because it can disrupt the approach it follows now. Indeed, a wide adoption of KubeVirt would obviate Kubernetes as a uniform, portable management and orchestration plane.

Thirdly, treating Kubernetes as just one more NFVI does not allow to use very strong features such as intent driven management that continuously reconciles an observed state of a service with a desired one (i.e., an intended declared state). A best practice to consume this intent management mechanism is via the Operator pattern[50]. This pattern

[46] OpenShift, https://www.openshift.com

[47] Flynn, https://flynn.io. Tsuru, https://tsuru.io

[48] https://kubernetes.io

[49] https://kubevirt.io/

[50] https://kubernetes.io/docs/concepts/extend-kubernetes/operator/

can be used to develop Kubernetes native S(pecialized)-VNFM for network services. That same pattern can be used to develop G(eneric)-VNFM and NFVO.

Finally, MANO today is rather workflow oriented than Operator oriented. While Operators and workflows are radically different patterns, Kubernetes-native workflow orchestration engines, such as Argo[51] use the operator approach to reconcile an actual state of a workflow execution with the desired execution state (i.e., the required workflow steps). Thus, Kubernetes also natively provides workflow capabilities needed in many practical orchestration situations where pure reconciliation cycles of the operator pattern might be too slow.

The CSPs need to deploy Kubernetes at *large scale* with hundreds of thousands of instances at the edge. However, this distributed cloud architecture imposes challenges in terms of resource management and application orchestration. In this perspective, k3s a lightweight K8s is put forward by Rancher[52] to address the increasing demand for small, easy to manage Kubernetes clusters running in resource-constrained environments such as edge. It is straightforward to see that k3s will enable the rolling out of new 5G services relying on multi-access Edge Computing deployments.

As detailed in **Figure 13**, k3s relies on the following Kubernetes components:

- *kube-apiserver*: It acts as the gatekeeper through which all operations are passed to perform on the cluster. It is responsible for exposing different APIs. To do so, it maintains RESTful services to perform operations, hence allows the configuration and validation of data related to k3s objects including pods, services, etc. Note that the aforementioned objects will be detailed later.
- *kube-manager*: It is responsible for the overall coordination and health checking of the entire cluster. It acts as the conductor and the coordinator which ensures that the nodes are up and running and the pods are behaving the right way and the desired state of the configuration is continually maintained
- *kube-scheduler*: It is responsible for physically scheduling artifacts which could be containers or pods across multiple nodes. Depending on the specified constraints in terms of CPU, memory, disk, affinity/anti affinity, etc., the scheduler selects the appropriate nodes that meet the criteria and schedules then the pod appropriately.
- *kubelet*: It is an agent which runs on the node to ensure the monitoring of the pods which are composed of containers running on the node, restarting them if required to keep the replication level. To do so, it watches for pod specs via the Kubernetes API server.
- *kube-proxy*: This is a network proxy which runs on the node to ensure TCP, UDP forwarding. It is used to reach services. Specifically, it reflects the services as defined in the Kubernetes API. It refers to the API server to build a bunch of iptables rules and reference the portal IP.

All these components are bundles into combined processes that are presented as a simple server and agent model which will facilitate their deployment in the edge environment.

[51] https://argoproj.github.io/

[52] https://rancher.com/

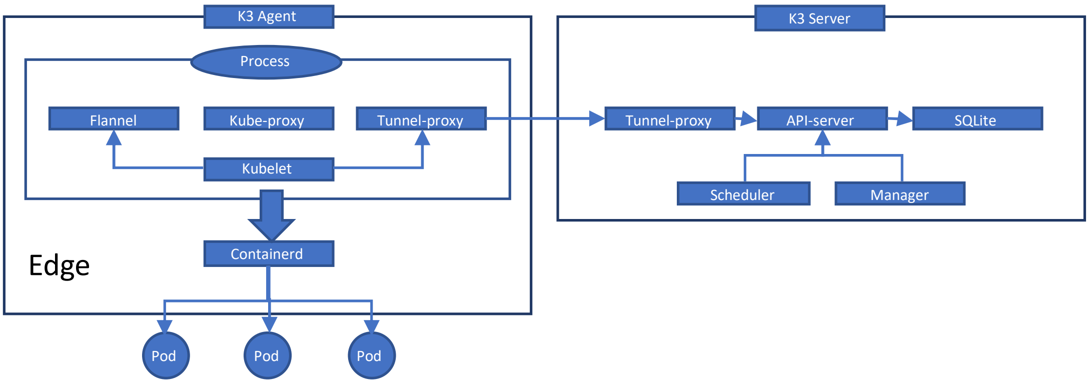


**Figure 13: k3s architecture**

## 2.2.2 OSM

Open Source MANO (OSM) is an ETSI-hosted open source community delivering a production-quality MANO stack for NFV, capable of consuming openly published information models, available to everyone, suitable for all VNFs, operationally significant and VIM-independent. OSM is aligned to NFV ISG information models while providing first-hand feedback based on its implementation experience[53] [54].

OSM Release EIGHT brings several improvements over previous releases. It allows you to combine within the same Network Service the flexibility of cloud-native applications with the predictability of traditional virtual and physical network functions (VNFs and PNFs) and all the required advanced networking required to build complex E2E telecom services. OSM Release EIGHT is at the forefront of Edge and 5G operations technology, deploying and operating containerized network functions on Kubernetes with a complete lifecycle management, and automated integration.

In addition, OSM extends the SDN framework to support the next generation of SDN solutions providing higher level primitives and increasing the number of available options for supporting I/O-intensive applications. Furthermore, the plugin models for intra and inter-datacenter SDN have been consolidated, and the management, addition, and maintenance of SDN plugins significantly simplified.

OSM Release EIGHT also brings major enhancements designed to improve the overall user experience and interoperability choices. This includes an improved workflow for VNF configuration which allows much faster and complex operations, and the support of additional types of infrastructures, such as Azure and VMware's vCD 10, complementing the previously available choices (OpenStack-based VIMs, VMware VIO, VMware vCD, AWS, Fog05 and OpenVIM). It improves the orchestration of diverse virtualization environments, including PNFs, a number of different VIMs for VNFs, and Kubernetes for CNFs.

[53] https://osm.etsi.org/docs/user-guide/02-osm-architecture-and-functions.html

[54] https://www.etsi.org/technologies/open-source-mano

### 2.2.3 ONAP

Open Network Automation Platform (ONAP) is an open-source project hosted by Linux Foundation[55], officially launched in 2017, enabling telco networks to become increasingly more autonomous. ONAP is capable of providing a real time, policy-driven service orchestration and automation, enabling telco operators and application developers to instantiate and configure network functions. ONAP, through the different releases, supports features like a) multi-site and multi-vendor automation capabilities, b) service and resources deployment, providing c) cloud network elements and services instantiation in a dynamic, real time and closed-loop manner for several major telco activities, (e.g. design, deployment and operation of services at design-time and run-time.

Various edge cloud architectures have already emerged from different communities and potentially can be plugged into the ONAP architecture for service orchestration. The ONAP community analyses the orchestration requirements of services over various edge clouds and how these requirements impact ONAP components in terms of data collection, processing, policy management, resource management, control loop models, security, as well as application & network function deployment and control. We invite the reader to read more detail in this link[56].

## 2.3 Networking programmability framework

Software Defined Networking (SDN) has been proposed as an alternative approach to manage, operate and design a computer network[57]. Differently from the Internet, based on a decentralized control plane, SDN concentrates the whole control plane within (at least logically) a single SDN controller. The controller is a software running on a server and this allows to identify a single "point-of-programmability" in the whole network. On the other side, the switches are stateless, natively unable to operate any forwarding operation and requiring the controllers to populate the internal flow tables, mapping the packet headers to the forwarding instructions. The network applications are developed as standard programming applications providing an unprecedented level of flexibility. Notably, before the advent of SDN, the network was only partially programmable, because the distributed nature of the data plane did not allow a coherent network view of the network state, which is instead available in an SDN controller.

The SDN controller is also defined as the "Network Operating System" (NOS) because of his similar role of a computer operating system: it acts as a middle layer between the network applications and the network resources (i.e., switches).

The northbound interface of the controller is responsible for the interaction with the network applications and provides all the programming APIs that can be exploited by the network developer. The level of abstraction provided by such APIs can be very different, from very low-level details (i.e., describing each single processing and forwarding operation of the switch) to a very high level (i.e., describing only what the application should do, and not how, as in the Intent-Based approach). The southbound interface is

[55] Open Network Automation Platform , https://docs.onap.org/en/elalto/index.html#

[56] https://wiki.onap.org/display/DW/Edge+Automation+through+ONAP+Arch.+Task+Force+-+Distributed+Management+%28ONAP+etc.%29+components

[57] Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE, 2014

instead responsible for the interaction with the network switches and supports all the protocols necessary to program the forwarding behaviour of the switch. OpenFlow has become one reference protocol for the southbound interface, but many other protocols have been defined as alternative or complementary to OpenFlow (e.g., *netconf*, *ovsdb*).

The reference architecture for SDN is typically *network based*, in the sense that the controller interacts with the switches along the path of a data flow in order to process and route the traffic correctly. An alternative SDN architecture is *source based*, in the sense that the controller interacts only with the source switch (i.e., the first SDN switch along the path of a flow), which adds, in piggybacking, the route information on the packets. This information describes how packets should be processed and switched at each traversed switch, extending the classical concept of source routing. This source-based architecture offers a wide flexibility in programming the network and has been implemented through the Segment Routing (SR) protocol[58]. Notably, SR is compatible with hybrid architectures in which a standard IP network coexists with SDN networks, since the route information for SDN is encapsulated within standard IP packets, switched as usual between legacy non-SDN switches.

### 2.3.1 SDN for Edge Computing

SDN is a technology that can help bridge the gap when combining Edge Computing and traditional clouds. For example, SDN can be used to act as a decision-maker on whether tasks should be uploaded and processed in the cloud or at the Edge.

SDN controllers can implement advanced traffic engineering schemes, able to cope autonomously with network impairments (e.g., link congestion, node/link failure). The adoption of AI enables the operation of "self-managing" networks.

Another dimension of the usage of SDN is related to users' mobility. This implies that the services should migrate from one EDGE to another in a seamless fashion for the final user. Migrating services is very challenging, since it requires to migrate the corresponding VM to a remote server, after having synchronized the internal state of the corresponding VMs and rerouted the corresponding traffic to the new server. The complexity of such migration requires a strict control on the traffic routing, as enabled by SDN[59].

### 2.3.2 Data plane programmability

Data plane programmability is a key technology towards network softwarization, enabling increased flexibility in networking. It extends the SDN paradigm beyond OpenFlow, offering full programmability on the packet processing pipeline of network devices. Furthermore, switches are stateful and can take local decisions, without the interaction with the SDN controller, with a beneficial effect on the latencies. Consequently, the design of network protocols/architectures evolves in a top-down fashion, in which NFs are defined in an abstract manner and then enforced to the network infrastructure. This enables the definition of specific packet processing pipelines tailored

[58] RFC 8402

[59] Baktir, A. C., Ozgovde, A., & Ersoy, C. , "How can EDGE computing benefit from software-defined networking: A survey, use cases, and future directions", IEEE Communications Surveys & Tutorials, 2017

to network applications (e.g., load balancing, in-band network telemetry, etc.) while providing high-performance and efficiency. Such applications may be implemented in software-based switches using commodity CPUs or hardware-accelerated devices such as programmable switches, smartNICs, etc. Programming these network elements to support complex network functions is achieved by defining finite state machines directly within the processing pipeline[60] or by defining primitives through a domain-specific language e.g., P4[61].

P4[62] is a declarative programming language for programming protocol-independent packet processors. It is a domain specific language with constructs (e.g., headers, parser, actions, tables, control flows, etc.) optimized for writing packet forwarding functions. Using P4, developers can program data plane packet pipelines based on a match/action architecture. They can create custom parsers for new protocol headers, define custom flow tables, the control flow between the tables, and custom actions. P4 programs allow developers to uniformly specify packet processing behaviour for a variety of targets (ASICs, FPGAs, CPUs, NPUs, and GPUs). The execution of a P4 program follows a simple abstract forwarding model with five distinct phases: parsing, ingress processing, replication and queuing, egress processing, and deparsing. The behaviour for each of these phases is defined by the declarations in the P4 program. A state during execution includes information from packet headers, metadata provided by the device or computed by the program, and the state kept in counters and registers. While the P4 language is target-independent, i.e., it abstracts from the specific hardware characteristics of the target device, a P4 compiler translates P4 programs into the instruction set of the hardware of the packet processor.

The current specification of the language P4 introduces the concept of the P4-programmable blocks; it essentially provides the interface to program the target via its set of P4-programmable components, externs and fixed components. Along with the corresponding P4 compiler, it enables programming the P4 target.

P4Runtime is the control plane interface for controlling forwarding behaviour at runtime. It is used for populating forwarding tables and manipulating forwarding state based on a P4 program and in a hardware agnostic way; the interface stays the same when your forwarding behaviour or hardware changes.

### Programmable traffic management

The centralization of the network's intelligence in SDN is an advantage for applications that do not have strict real-time requirements and depend on global network state. However, when the service uses local state information, the same level of flexibility must be supported at the data plane. Enabling advanced, highly portable, programmable L3 QoS behaviours at the Edge of the network, in order to support QoS requirements for MEC-enabled 5G networks, assumes fine-grained QoS control and standardized access to additional hardware capabilities.

[60] Pontarelli, Salvatore, et al. "FlowBlaze: Stateful packet processing in hardware." , NSDI 2019

[61] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese, and D. Walker, "P4: Programming protocol-independent packet processors," SIGCOMM Comput. Commun. Rev., vol. 44, no. 3, pp. 87–95, Jul. 2014.

[62] https://p4.org/

Towards that end, making stateful data plane algorithms programmable, complementing the programmable forwarding plane solutions, can be beneficial in terms of meeting QoS requirements (e.g., low latency communications) and enhance network flexibility. Programmable data plane solutions such as P4 and supported architectures, provide an excellent way to define the packet forwarding behaviour of network devices. However, most programmable devices still typically have non-programmable traffic managers. Towards that end, 5GROWTH[63,64] investigates fully programmable and customized data planes, through the introduction of simple data-plane abstractions and primitives beyond forwarding, enabling optimized traffic management per slice, depending on the application profile and corresponding Service Level Agreement (SLA).

### P4-assisted coordination of VNFs

Advanced network applications are based on stateful VNFs, i.e., an internal state is kept within the VNF during the traffic operations. Typical examples are traffic classifiers, traffic shapers, and firewalls. Scaling such network applications for large networks and/or for high data rate requires to replicate the same VNF into different servers and to distribute the traffic across all the instances of the VNF. This coordination between VNFs requires that the internal state should be shared across the replicas. As a toy example consider a distributed Denial-of-Service Detection (DoSD) application in which many replicas of the same VNF are distributed at different ingress routers of a network. The detection is based on evaluating the overall traffic entering the network from all edge routers. This application requires to share the metrics of the local traffic among the VNF replicas in order to compute the network-wide traffic. A solution to the problem of state replication would be to implement a standard replication protocol directly in the VNF (like Paxos, RAFT, etc), but this requires loading the VNF with also this replication process, which can be quite complex and computation intensive.

An alternative solution is to leverage a stateful data plane, e.g., based on P4. This implies that the state replication is offloaded from the VNFs to the P4 switches, which take the responsibility of coordinating the exchange of replication messages between VNFs, with a beneficial effect on the VNF load and thus on the overall scalability.

In particular, the 5G EVE[65] project is investigating how to implement a publish-subscribe scheme directly on P4 switches, according to which the VNFs can publish the updates on their internal states and can subscribe on the updates from the other VNFs. This allows to achieve a state replication which is lightweight for the VNFs and that exploits the high processing speed of P4 switches.

[63] D2.1: Initial Design of 5G End-to-End Service Platform, [online] http://5growth.eu/wp-content/uploads/2019/11/D2.1-Initial_Design_of_5G_End-to-End_Service_Platform.pdf

[64] D2.2: Initial implementation of 5G End-to-End Service Platform, [online] http://5growth.eu/wp-content/uploads/2020/05/D2.2-Initial_implementation_of_5G_End-to-End_Service_Platform.pdf

[65] https://www.5g-eve.eu/

## 2.4 Acceleration at the Edge: The Need for High Performance at the Edge

The challenge of achieving deterministic high bandwidth and low latency to support today’s demanding Edge Computing use cases is not trivial. The ability to increase bandwidth and reduce latency while providing required levels of data processing and security is extremely valuable at the Edge of the network.

The easiest way to achieve such high performance at the network Edge is to move the data processing and forwarding closer to the end users. There is no room for monolithic, single-function ASIC-based appliances to provide the necessary performance, so a solution is needed that can take advantage of existing networking equipment while accelerating the data path.

NFV has proven to be a breakthrough technology; by performing necessary networking and security functions in software installed on standard x86 servers in small Edge locations, it is possible to gain necessary flexibility and agility at the Edge. However, there is a limit to the performance attainable when standard CPUs are running networking software, and often it is required even more space because the functions are so CPU-intensive that too many cores are burned in the process, necessitating multiple expensive servers to handle the job. When it comes to very high bandwidth and low latency, CPU-based software networking alone is not a resource-effective solution.

The solution must provide the performance of an ASIC with the agility of software. The answer is to offload the virtual functions to hardware, providing the necessary acceleration while maintaining flexibility.

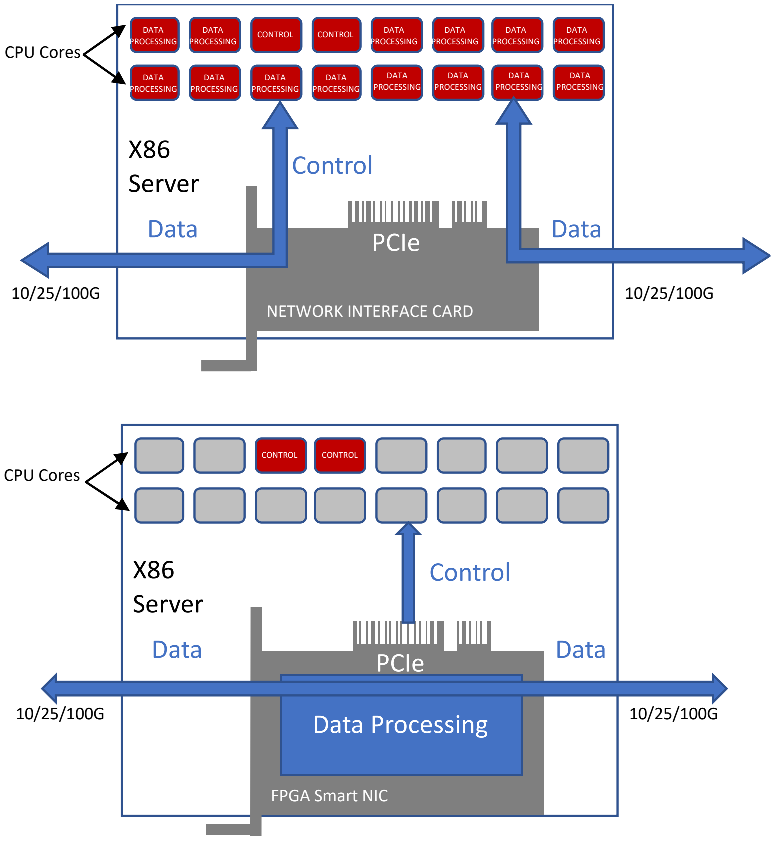


**Figure 14: Up: a standard NIC on an x86 server, Down: a FPGA-based SmartNIC**

In order to enable the Edge Computing infrastructure with high bandwidth, low latency, and a highly secure data path, we need hardware-based acceleration at the network Edge. There are various hardware technologies capable of offloading certain network functions, however none accomplish full offload of the data plane as well as field-programmable gate arrays (FPGAs).

### 2.4.1 FPGA as a Platform to Accelerate Edge Computing

FPGAs are programmable hardware, which offer the performance of an ASIC with the flexibility of software. Because of their parallel processing capabilities and their highly pipelined architecture, FPGAs are optimized to handle CPU-intensive networking and security functions efficiently. This enables very high bandwidth and better ability to scale for high throughput applications. That is why FPGA SmartNICs[66] are a key enabling technology in next-generation networks.

FPGA SmartNICs are also very effective in reducing latency and jitter. By using an FPGA to handle data processing, it is feasible to achieve a latency of a few microsecond (µs), because the data path avoids the CPU entirely. Instead, the data is fully offloaded from the CPU to the FPGA on the NIC. By comparison, when software on a CPU is used for networking, latency lower than 50-100 ms is considered a very good achievement.

Another important advantage of offloading the data path from CPUs is in the area of cybersecurity. If the data never needs to reach the CPU, the networking is entirely separated from the computation. Should the CPU, which is much more vulnerable to breaches than an FPGA, be hacked, the data path (handled by the FPGA) is still protected. The FPGA also can efficiently handle security functions such as encryption and decryption, Access Control List (ACL), and firewall, thereby reducing the load on the CPU.

Beyond meeting the bandwidth, latency, and security requirements of challenging Edge Computing implementations, FPGAs also have the benefit of being open, programmable and configurable hardware, and a perfect complement to commercial off-the-shelf servers in that they are general purpose and agile. Their full reprogrammability means that they are futureproof, i.e., hardware does not need to be replaced or upgraded when new functionalities and features emerge. The FPGA SmartNIC can be reprogrammed as needed instead of replacing the whole card if the applications or use cases change.

FPGA-based SmartNICs provide unmatched scalability to enable communication service providers to easily handle large numbers of subscribers and devices at cost without significantly adding latency and power. This is crucial for Edge Computing, which is expected to expand to ever-more network endpoints as the technology evolves and use cases become more prominent.

### 2.4.2 Direct Memory Access on FPGA

As the recognition that disaggregation solutions provide a respectable alternative for service providers and enterprises networks, the number of appliances that are based on general-purpose computing equipment increases. With it, the use of NFV implementations also increases and is rapidly growing as an enabling engine to all these

[66] https://ethernitynet.com/cornerstones/fpga-smartnics-for-network-acceleration/

appliances. Ideally, a preferred solution should be able to provide all the required features without compromising on performance and without having cost tradeoffs to achieve these performing solutions. However, such high-end server solutions are very expensive, as they contain a high number of CPU cores and vast amounts of memory in order to achieve such performance.

An FPGA that also includes an embedded PCIe Direct Memory Access (DMA) engine allows NFV performance to be boosted by accelerating several virtual software appliances in the hardware. Two main technologies can best use the DMA capabilities: SR-IOV and PCI Passthrough. By using these two technologies on a single FPGA board, the traffic can bypass different server bottlenecks around the hypervisors and gain direct access through the PCIe to many networking tasks in hardware. If the ability to use DPDK[67] is added to the DMA functionality, it is possible to receive even greater acceleration and further improvement to the NIC performance. The combined result is a boost to the performance of multiple virtual networking functions to the level of that of dedicated hardware appliances.

A server that incorporates FPGA-based SmartNICs, that are capable of combining DMA functionality with hardware forwarding offload engines, provides a highly performing, cost-optimized alternative to a costly server and/or to dedicated hardware appliances. The FPGA can perform different networking functions in hardware as if they were in a virtual software environment. This capability can replace multiple VMs, which then reduces the number of CPU cores and provides the required performance without any cost tradeoff.

### 2.4.3 Seamless Virtualized Acceleration Layer

DPDK APIs are now the de-facto standard for hardware offload. In DPDK, when a NIC is brought up, it lists its capabilities. The DPDK application can then decide whether to activate the hardware offload or not. If not, the DPDK application continues to work with no hardware offload, performing all the features through software.

DPDK uses several libraries for hardware offload. The main ones are *rte_eth* and *rte_flow*[68] . The *rte_eth* library includes APIs for configuration and reading statistics for the device itself and for the physical ports. The *rte_flow* library includes APIs for flow configuration and statistics. The *rte_flow* APIs provide a rich solution that is a good match for offloading a wide variety of Virtual Network Function/Container Network Functions.

The suggested approach for hardware offload is transparent control flow mode, in which the FPGA configuration is transparent to the DPDK application. In this mode, the FPGA is not a separate controlled element. The DPDK application sees a single SmartNIC entity that combines the Ethernet controller and FPGA. The benefit of this control flow mode is that the application does not need to write any specific code to use the FPGA acceleration and is therefore agnostic to the underlying hardware.

[67] Data Plane Development Kit: https://www.dpdk.org/

[68] https://doc.dpdk.org/guides-19.08/prog_guide/rte_flow.html

## 2.5 Operations at the Edge

### 2.5.1 From DevOps to Dev-for-Operations

Edge Computing is about placing workloads close to the Edge where the data and the actions have been taken. This special domain requires a generalized DevOps methodology to code, test, deploy, and run the apps.

The objective of DevOps is to break barriers between development and operations teams in the software engineering and usage stages[69]. This is usually done by assigning certain operation tasks to developers and vice versa. However, the whole concept goes much further and is best summarised as implementing a continuous cross-functional mode of working with focus on automation and alignment with the business objectives; this is commonly represented by a kind of "infinite" loop such as the one in  Figure 15:

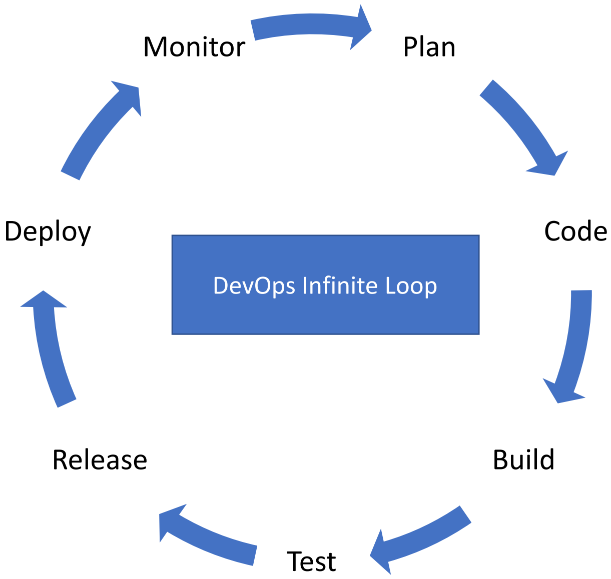


**Figure 15: DevOps Infinite Loop[70]**

This representation suggests the general concept of "continuity", with main focus on automation, which is usually applied to four main stages: integration, delivery, deployment, and monitoring. This has led to the introduction of the following fundamental concepts:

a) **Continuous Integration** (CI): It is a process where developers can integrate their changes continuously in the code repositories. While they do so, pre-defined test batteries are automatically executed to find and fix errors in a continuous way.

69 Erich F., Amrit C., Daneva M., "A Mapping Study on Cooperation between Information System Development and Operations", In: Jedlitschka A., Kuvaja P., Kuhrmann M., Männistö T., Münch J., Raatikainen M. (eds) Product-Focused Software Process Improvement. PROFES 2014. Lecture Notes in Computer Science, vol. 8892. Springer, Cham, 2014.

[70] 70 Ignacio Labrador, Aurora Ramos and Aljosa Pasic, "Next Generation Platform-as-a-Service (NGPaaS) From DevOps to Dev-for-Operations", White Paper, available online: https://atos.net/wp-content/uploads/2019/07/white-paper_ari_NGPaaS.pdf

This accelerates the software development process by reducing the time to validate and publish new software updates.

b) **Continuous Delivery** (CD): It refers to automating the writing process on the code repository, CD refers to the automation on extracting the code from it to generate ready-to-use software packages. CD is commonly used to automatically produce software releases in a regular way (e.g., daily, weekly, etc.) by just hitting a button on the CD tool.

c) **Continuous Deployment** (Cd) It refers to the automation of even the deployment phase without human intervention. So, changes from developers could be automatically propagated to the production environment without human intervention if no errors were detected.

d) **Continuous Monitoring** (CM) refers to monitoring (see section 6.4.2) performed along the whole cycle from development to production and operation environments. The goal is to use real production data for the development and operations teams. Automation also is applied here: instead of relying only on human responses to alerts or relevant events, autonomous responses to certain alarm conditions can be implemented.

It is not surprising that telco-grade operators are very interested in the DevOps methodology. After all, typical production and operational environments of telecommunication organisations can be very different from the usual testing environment, with many adjustments to be done. According to a recent article[71] telecoms industry is already the biggest adopter of DevOps and seems to be most willing to further enhance the usage of this methodology. But as digital transformation of the telecommunication sector is pushing towards software-defined communication services, the “traditional” DevOps approach is not sufficient anymore, since in this scenario, development and operation tasks are not just performed by different teams or departments of a single organization; instead they are spanning multiple vendors which independently develop the software (and hardware) resources which are combined together in an operational environment on the telecom operators' infrastructure. In addition, if we think about 5G and beyond networks, it is also necessary to consider that the network can be split in such a way that different network slices could be isolated and assigned to other different industries (verticals). The resulting picture is a complex ecosystem with large network operators working together with a plethora of vendors and verticals to implement and operate their network services under strict Service Level Agreements (SLA). The Dev-for-Operations model introduced in the NGPaaS project[72] considers these and other challenges to help in adapting a DevOps-like philosophy in the context of the forthcoming next generation telecommunications industry and is well fitted to the Edge domain, where many actors should interact.

The Dev-for-Operations model developed in the NGPaaS project differs and enhances in several aspects DevOps, for instance:

[71] Kahle, J. (2018). Why Are Some Industries So Far Behind on DevOps? - Highlight. [online] Highlight: The world of enterprise IT is changing, fast. Keep up. Available at: https://www.ca.com/en/blog-highlight/why-are-some-industries-so-far-behind-ondevops.html [Accessed 26 Apr. 2018]

[72] http://ngpaas.eu/

a) It should be possible to execute a vendor specific CI/CD loop at the vendor's site in order to make it possible to iteratively develop and debug the service before delivering it towards the operator's side.

b) The Dev-for-Operations model should make possible the communication of the operator insights towards the vendor's environment in some way. This should enable vendors to have a deep understanding of the operational environment, so they can perform a kind of "operation-aware" testing function on their own. This requirement has a lot of impact in the Edge domain. Unlike the cloud, the Edge can be unstable and even disconnected by design. There can be many points of failure in an Edge solution. Building an Edge-native application requires the ability to be ready to scale back to the cloud at any point. This means they should perform CI/CD processes using test batteries already integrating the relevant features of the operational environment.

c) DevOps delivers the application, but Dev-for-Operations should make it possible to deliver a fully realized service including the core application, monitoring and analytic, as well as deployment and adaptation capabilities.

d) Like in the regular DevOps approach, there should be also a specific feedback loop to propagate the information from the Operator's side towards the vendor environment, but in this case, the feedback should integrate information not only from the software application itself, but also regarding the associated monitoring and analytics, as well as the deployment and adaptation indicators.

e) The feedback mechanism takes on a different character in Dev-for-Operations: it should consider the separation between vendor and operator but keeping the automatic or semi-automatic mechanisms needed to provide the feedback in a timely manner.

The Dev-for-Operations model is well suited to develop applications and services in the Edge which is characterized by a few nuances like scaling, types of devices, application footprint, operating speed, and disconnection.

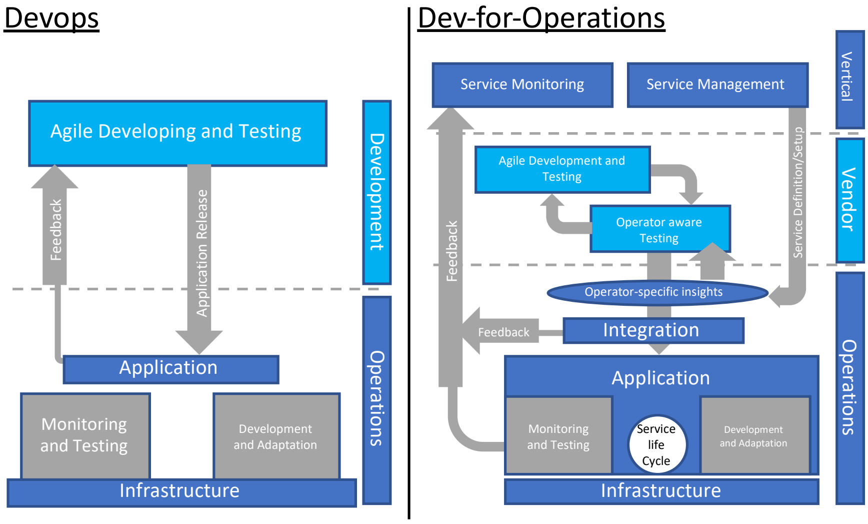


**Figure 16: DevOps vs Dev-for-Operations Workflow**

### 2.5.2 DevSecOps and Edge Computing

DevSecOps is a set of methods, practices, and tools for accelerating software cycles, from development to deployment, from business to operational systems in a continuous cycle. It is integrating security practices within the DevOps process by creating a security as code with ongoing, flexible collaboration between release engineers and security teams. The DevSecOps cycle comprises:

1. Design - modelling an application in high-level abstraction.
2. Development - translating application models to deployment read software components.
3. Testing - validating software components based on required behaviours.
4. Deployment - distributing software components to computational resources.
5. Maintenance and analysis - continuously monitoring application behaviour and adapting it to changing environmental conditions.

Security (both at software and system levels) and privacy need to be of utmost importance all along (1)-(5) activities in order to deliver a trustworthy software.

DevSecOps in the Edge context is challenging because: (a) design requires tools for describing the dynamic behaviour of application components in Edge environments; (b) development needs to implement quality assured software based on the designed model; (c) testing requires real-time simulation of the application behaviour in the runtime environment of a heterogeneous Fog environment; (d) deployment requires mechanisms to seamlessly redeploy the software in the Fog at runtime; (e) maintenance and analysis requires extraction of process logs to provide recommendations for redesigning the model. The transient nature of the environment and the massively distributed geographic resources make DevSecOps challenging. Additionally, a DevSecOps framework that can undertake activities, such as automated management, including software adaptivity to respond to the changing environment, would alleviate the burden of managing serverless functions. Currently, there are no DevSecOps platforms that can manage the activities from modelling to (re)deployment of a Fog application that is designed via serverless computing. A few example platforms are available for the Cloud[73] [74] [75]. However, they do not address adaptivity (provide tools for modelling and enacting self-properties) and are not designed for serverless environments.

Fog computing envisions a highly dynamic operational environment. In this context, to achieve ideal DevSecOps it is essential to first capture domain specific concepts, such as target platform features and non-functional software requirements in models to instruct self-adapting mechanisms. Then, operational data obtained from the runtime using lightweight monitoring solutions can be used for providing timely recommendations to users. DevSecOps however needs to be more sophisticated than simply providing the mechanisms for adapting to changes but must also ensure trustworthiness by design. This is extremely challenging given the massive disaggregation of resources along the Cloud-

[73] J. Wettinger et al. "Middleware-oriented Deployment Automation for Cloud Applications." IEEE Trans. on Cloud Computing, Vol. 6, Issue 4, pp. 1054-1066, 2016.

[74] G. Pallis et al. "DevOps as a Service: Pushing the Boundaries of Microservice Adoption." IEEE Internet Computing, Vol. 22, Issue 3, 2018, pp. 65-71.

[75] N. Ferry et al. “CloudMF: Model-Driven Management of Multi-Cloud Applications.” ACM Transactions on Internet Technology, Vol. 18, No. 2, pp. 16:1-16:24, 2018.

Edge continuum and the consequent distribution of data that needs to be managed from privacy breaches arising from unwanted and unforeseen data affinities.

### 2.5.3 Monitoring

Observability and analysis, consisting of monitoring, logging and tracing, are crucial requirements of any service deployment, and particularly for VNFs[76].

In this section we elaborate on how these requirements apply to the network functions that reside at the Edge of the network. But before we embark onto this, let’s define what each of these capabilities is and why they are critical for DevOps.

In general, observability involves gathering data about the operation of services, typically referred to as “telemetry”. Modern service platforms, infrastructures and frameworks have observability systems in place that gather three types of telemetry:

- Metrics: Time-series data that typically measure the four “golden signals” of monitoring: latency, traffic, errors, and saturation. Analysis is done in monitoring dashboards that summarize these metrics, providing aggregations, slicing & dicing, statistical analysis, outlier detection and alerting capabilities. DevOps depends on these metrics to understand the performance, throughput, reliability and scale of the services. They also monitor Service Level Indicators (SLIs) to detect any deviations from Service Level Objectives (SLOs), ideally before they lead to SLA violations.
- Logs: As traffic flows into a service, this is the capability to generate a full record of each request, including source and destination metadata. This information enables DevOps to audit service behaviour down to the individual service instance level. Analysis is typically done via search UIs that filter logs based on queries and patterns, indispensable for troubleshooting and root cause analysis of operational issues.
- Traces: Timestamped records about the handling of requests, or “calls”, by service instances. As a result of the decomposition of network services into many VNFs and of monoliths into numerous micro-services, and the creation of service chains/meshes that route calls between them, modern service infrastructures offer distributed tracing capabilities. They generate trace spans for each service, providing DevOps with detailed visibility of call flows and service dependencies within a chain/mesh.

On the surface, the approaches towards delivering the observability capabilities have been quite different between the NFV and Cloud Native Computing Foundation (CNCF) “ecosystems”. Before the softwarization of network functions, each PNF had to offer its own monitoring, logging and tracing functions, ideally through (de facto) standard protocols (SNMP, syslog, IPFIX/NetFlow, etc.). Moreover, specialized network appliances, such as Probes, DPIs and Application Delivery Controllers (ADCs) offered more advanced network visibility capabilities, in terms of gathering deep network telemetry, both in-band (inline) or out-of-band (via port-mirroring).

When PNFs transformed into VNFs, deployed as VMs, they have started to leverage the telemetry capabilities of initially the VIM and subsequently of the NFVO/MANO stack of choice. This resulted into a proliferation of relevant projects:

[76] https://5g-ppp.eu/wp-content/uploads/2019/09/5GPPP-Software-Network-WG-White-Paper-2019_FINAL.pdf

- OpenStack: The set of projects under OpenStack Telemetry, with Ceilometer being the one most widely adopted[77].
- OPNFV: The Barometer[78] and VES[79] projects.
- OSM: The OSM MON module and respective Performance Management capabilities[80].
- ONAP: The Data Collection Analytics and Events (DCAE) project[81].

On the deep network visibility front, there have been efforts to enable network monitoring in a programmable fashion[82] (see 2.3.2) and ongoing standardization activities under IETF[83].

On the CNCF side, there is a separate set of projects under the Observability & Analysis section of the landscape[84], with Prometheus[85], fluentd[86] and Jaeger[87] as the graduated monitoring, logging and tracing projects correspondingly, with OpenMetrics/OpenTelemetry aiming to establish open standards and protocols. The open APM ecosystem is even broader [88].

However, as mentioned earlier in this white paper, 5G service implementations are adopting cloud-native approaches. We expect that service infrastructures/frameworks will thus be enhanced with capabilities that offer observability as shared basic functions.

In addition, the specialized appliances we mentioned e.g., ADCs, which have since embraced or reinforced their softwarization, virtualization & cloudification, will be enhanced with capabilities that better position them in a hybrid multi-cloud world of cloud-native applications and services.

The enhancements towards cloud native and PaaS are discussed in ETSI IFA029[89], where the concept of VNF common and dedicated services has been introduced. These VNFs are instantiated inside the PaaS and expose capabilities that are consumed by the network services (composed by consumer VNFs) that run over the PaaS:

- VNF Common Service: common services or functions for multiple consumers. Instantiated independently of any consumer.

[77] https://wiki.openstack.org/wiki/Telemetry

[78] https://wiki.opnfv.org/display/fastpath/Barometer+Home

[79] https://wiki.opnfv.org/display/ves/VES+Home

[80] https://osm.etsi.org/wikipub/index.php/OSM_Performance_Management

[81] https://wiki.onap.org/display/DW/Data+Collection+Analytics+and+Events+Project

[82] https://p4.org/p4/inband-network-telemetry/

[83] https://datatracker.ietf.org/doc/draft-ietf-opsawg-ntf/

[84] https://landscape.cncf.io/category=observability-and-analysis

[85] https://prometheus.io

[86] https://www.fluentd.org

[87] https://www.jaegertracing.io

[88] https://openapm.io/landscape

[89] https://www.etsi.org/deliver/etsi_gr/NFV-IFA/001_099/029/03.03.01_60/gr_NFV-IFA029v030301p.pdf

- VNF Dedicated Service: required by a limited set of consumers with a specific scope. Instantiated dependently of their consumers (when required by a consumer) and destroyed when no relation exists with any consumer[90].

Worth highlighting is the fact that a "generic monitoring service" is mentioned as a specific example of a VNF Common Service. We anticipate that this trend will expand to cover all observability & analysis capabilities we covered. And due to the adoption of Kubernetes as the service orchestration framework, the implementation will be most probably based on the technologies/projects in the relevant area of the CNCF landscape.

For example, ONF EDGE Cloud[91] platforms, i.e. Aether, CORD & XOS, have already adopted the pattern of offering logging and monitoring as platform micro-services, leveraging projects from the CNCF observability and open APM ecosystems (Kafka, Prometheus/Grafana and ELK/Kibana).

This trend is strengthened further by the approach pursued by the Hyperscalers to expand their cloud services into the Edge of the network. AWS Outposts, Azure Stack, Google Anthos, IBM Cloud Satellite (will) all offer Kubernetes on the Edge. There is some fragmentation in how observability is implemented by each cloud provider, because of the different cloud services that support the monitoring aspects (AWS CloudWatch, Azure Monitor and Google Stackdriver). But Istio[92] is acting as a unifying service mesh technology, since it implements the observability functions in a common way, without additional burden on the service developers. We will have to see if/how the service mesh expands to the Edge offerings of the Hyperscalers.

In terms of how these capabilities will be implemented on Edge infrastructure of smaller footprint: In scenarios where Edge resources are too limited to justify a full-blown K8s installation, K3s[93] and KubeEDGE[94] are emerging as alternative options.

Similarly, early stage & fragmented are the monitoring features of serverless frameworks. Most of them provide or support eventing frameworks as standard, that can be used for building metrics and telemetry capabilities. But the approaches and tools are not common.

As cloud-native and Edge-enabled service deployments and implementations become a reality, the next challenge to be addressed is analysing the huge volumes of telemetry generated and the need for human-in-the-loop operations that increases toil (and costs). The evolution of monitoring and APM to the direction of introducing more automation and intelligence through ML/AI techniques is commonly referred to as "AIOps". The integration of ONAP DCAE with Linux Foundation Acumos AI[95] is exactly a development in that direction. MonB5G[96] introduces Monitoring System, Analytics Engine and Decision Engine elements as common functions, combined with ML/AI

[90] https://5g-ppp.eu/wp-content/uploads/2020/02/5G-PPP-SN-WG-5G-and-Cloud-Native.pdf

[91] https://www.opennetworking.org/onf-EDGE-cloud-platforms/

[92] https://istio.io

[93] https://k3s.io

[94] https://kubeEDGE.io/

[95] https://www.acumos.org

[96] www.monb5g.eu

techniques for data-driven decision making, to automate the management, orchestration and optimization of massive numbers of services divided across massive numbers of slices and deployed on RAN, Edge and Cloud POPs in beyond-5G networks.

# 3. Edge Computing and Security

## 3.1 Key security threats induced by virtualization

Edge computing ranges from single vertical 5G cabinet-run application to small multi-tenant cloud processing sheltered units. Edge computing are cost-optimized to fulfil the tailored local needs (computing, storage, throughput and latency). The cost driver impacts the software deployment solutions. Full-fledged VM (i.e., bearing integral OS) deployments offering the flexibility needed at core network processing could be viewed as too costly for edge computing.

Edge computing inherits its paradigm and key technical building blocks from virtualization and cloud-native processing. When deployed for 5G networking, edge computers will be one more computing resource over the network, able to receive certified payloads (VNF or CNF) from the orchestrator, check their validity running the security procedure and execute the code. It implicitly also inherits the security threats brought by virtualisation and containerization with a special emphasis however where it differs from core network computing. Edge computing are typically processed in isolated cabinets closed to users. Small processing units cannot compete with stringent security policy rules and standards of a single site massive processing delivered by core networks infra operators. Nevertheless, when verticals such as autonomous cars rely on cabinet-hosed edges, security is a major concern at the Edge too. It is important to reassert on which flank Edge Computing is or could be more vulnerable on possible attacks which are more likely to occur. Looking at a high level, the main security needs can be defined as:

i) Protecting a payload (container or VM) from the application inside it
ii) Inter payload (container or VM) protection
iii) Protecting the host from a payload (container or VM)
iv) Protecting the payload (container or VM) against the host (aka, introspection)

Simply said, the attack path may originate from the container or the VM and is directed to the host (with an intent to brake isolation barrier of a targeted VM or container) or reversely be initiated at the host with full introspection mean to access to one VM or container memory space. The former threat is remediated by VM or container **isolation techniques** which act at several levels (i.e., limiting the types of interactions-system calls with the host, memory segregation into payload isolated partitions, payload resource consumption control). For the latter (e.g., introspection), the remediation comes with the concept of **trusted execution** and the associated technologies (e.g, Intel SGX enclave) that makes certain that even a malicious host OS or operator cannot tamper or inspect any managed payload memory space.

## 3.2 Security of the MEC infrastructure

As defined in the ETSI MEC 003 standard[97], the MEC reference architecture consists of different functional elements, the infrastructure of which should be secured at every level according to best practices for similar non-MEC-specific technologies, as described here below.

The MEC platform manager has privileged access to all the *managed* MEC hosts where MEC applications are running, therefore should be protected against unauthorized access using best practices of access control, e.g. least privilege principle, separation of duties, RBAC/ABAC policy enforcement, to name a few. In particular, the MEC platform manager should strongly authenticate requests (e.g. with X.509 certificate) on its management interfaces (Mm2/Mm3), to verify they originate from an authorized MEC orchestrator or OSS. Similarly, the underlying VIM, which manages the virtualization infrastructure of the MEC hosts (where the data plane runs), should strongly authenticate requests on its management interfaces (Mm4/Mm6) as coming from an authorized MEC platform manager if not in the same trust domain (e.g. co-located), or an authorized MEC orchestrator.

The MEC hosts must be secured according to best practices of server security and virtualization infrastructure security.

- NFV recommendations: for MEC systems based on the NFV architecture and running sensitive workloads, the ETSI NFV-SEC 003 specification[98] defines specific security requirements for isolation of such workloads (e.g. security functions) from non-sensitive ones and describes different technologies to enhance the security of the host system (e.g. MEC host) in this regard: system hardening techniques, system-level authentication and access control, physical controls, communications security, software integrity protection, Trusted Execution Environments, Hardware Security Modules, etc.
- MEC-specific recommendations MEC platform should strongly authenticate requests on its Mm5 interface as coming from an authorized MEC platform manager. Similarly, the Virtualisation infrastructure should strongly authenticate requests on its Mm7 interface to make sure each one is a valid request from an authorized VIM. Furthermore, inside the MEC host, both isolations of resources and data must be guaranteed between the MEC apps, since they may belong to different tenants, users, or network slices in 5G context. In particular, the MEC platform is shared by the various MEC apps and therefore must use fine-grained access control mechanisms to guarantee such isolations, i.e. let a given MEC app access only the services and information they have been authorized to.

At the MEC system level, the MEC orchestrator is not only critical because it has privileged access to the MEC platform manager and VIM, but also because it is particularly exposed to end-user devices via the User app Life Cycle Management proxy. Indeed, this proxy allows device applications to create and terminate (and possibly more) user applications in the MEC system, via the MEC orchestrator.

[97] ETSI GS MEC 003 v2.1.1 (Framework and reference architecture)

[98] ETSI GS NFV-SEC 012 v3.1.1 (System architecture specification for execution of sensitive NFV components)

### 3.2.1 MEC specific threats

In section 5.4 of a recent document[99], ENISA has identified specific threats to the MEC that should be addressed:

1. *False or rogue MEC gateway*: this concerns MEC systems deployed fully or partially on the end-user side, e.g., inside residential gateways or smart connected devices, that become more and more open, therefore more exposed to malicious users deploying their own MEC software or device and acting as a Man in the Middle (MitM).
2. *Edge node overload*: certain user applications (typically mobile ones) and/or IoT devices may flood one or more MEC nodes with traffic, resulting in a Denial-of-Service (DoS) for other connected users or devices.
3. *Abuse of edge open APIs*: MEC uses open APIs *mainly to provide support for federated services and interactions with different providers and content creators*. Such API openness can be easily abused without proper security controls in place, resulting in DoS, MitM, unauthorized access, privilege escalation, etc…

Besides the threats identified by ENISA, the ETSI MEC 002 specification[100] has stated a few security requirements in section 8.1:

- [Security-01] The MEC system shall provide a secure environment for running services for the following actors: the user, the network operator, the third-party application provider, the application developer, the content provider, and the platform vendor.
- [Security-02] The MEC platform shall only provide a MEC application with the information for which the application is authorized.

### 3.2.2 E2E slice security in the context of MEC

As part of 5G networks, MEC systems should support "5G slices", a concept introduced originally in the NGMN 5G White Paper[101] and expanded ever since by the various 5G standardisation organisations. Indeed, especially regarding security, it is critical to include MEC in the network slicing in order to meet E2E security requirements from verticals. On the one hand, MEC support for network slicing has been addressed in NFV domain by ETSI MEC 024 specification[102]. On the other hand, ETSI NFV-SEC 013 specification[103] also defines a high-level policy-driven security management architecture for NVF infrastructures, that could apply to NFV-based MEC, and therefore bring E2E slice security to the MEC.

## 3.3 New trends in virtualization techniques

For the reason exposed above, we will consider container based and *unikernel*-based virtualization schemes only, viewed as two possible paths for Edge Computing future.

[99] https://www.enisa.europa.eu/publications/enisa-threat-landscape-for-5g-networks

[100] ETSI GS MEC 002 v2.1.1 (Phase 2 : Use cases and requirements)

[101] NGMN Alliance: "5G White Paper", February 2015

[102] ETSI GR MEC 024 v2.1.1 (Support for network slicing)

[103] ETSI GS NFV-SEC 013 V3.1.1 (Security management and monitoring specification)

Both meet the cost effectiveness needed at the Edge. There are two emerging competing techniques dealing with both security, limited storage requirement and instant payload start-up. They are lightweight hardware-level virtualization (aka, lightweight virtual machine), embarking one bare minimal guest kernel on the one hand, and on the other hand, operating system level virtualization (aka, containers). Both technologies are backed by intense research and industrial deployment by IT leaders (Intel, IBM, Amazon, Google) resulting from internal developments and first running deployments. Amazon and Google are already exploiting these technologies on their running operations for improving the security, running costs and quality of service.

The **relative strengths on the two techniques** are accepted as follows: VMs bring higher process isolation and deployment flexibility but at higher memory costs (i.e., replication of different feature-rich guest operating systems in each VM) and are much slower to start. Designing a lightweight virtualization (as Amazon's *Firecracker*) is aimed at maintaining the security advantage while significantly thinning-out the above-mentioned known drawbacks and somehow losing the flexibility advantage too as the guest OS is reduced, optimized and unique.

Valuated as less secure, containers last improvements were aimed at enhancing security and process isolation to bridge the security gap from what virtualization offers. Linux container isolation has been significantly improved in the recent past with new frameworks (see below), instantiating same core Linux OS container security enablers (cgroups, namespaces, seccomp, …).

For an interested reader on this subject, there are four initiatives that are likely to pave the way for the future of (Edge Computing) virtualization: IBM Nabla containers, Google gVisor containers, Amazon's Firecracker lightweight VMs and OpenStack Kata lightweight VMs.

IBM's researcher James Bottomley had reached an atypical conclusion (versus the commonly accepted opinion) by discerning from his research that containers are more secured than VMs. Simply said, he estimates the number of lines of kernel code (with a linear relationship with the number of possible vulnerabilities resident there) that interacts with the payload. The container engine (a kernel module that interacts with all containers) exposes less code than a VM hypervisor added with the full OS code resident in each VM. An extra benefit is viewed that if the container engine has been found vulnerable, its replacement directly benefits to all supported containers without requiring any changes on containers. This opposes to a failed VM hypervisor which entails the replacement of all guest OS in the majority of the cases. This quantitative approach has its merits to shed light on the kernel code potential vulnerabilities and the much higher size of virtual machine kernel code. However, a complementary qualitative approach would be beneficial to evaluate the security gains brought by hardware-based Intel Virtual Technology (or equivalent at AMD) as well as the gains brought by the barrier erected by the guest OS (of VMs), creating a walled-garden for the attacker.

#### *Containers isolation techniques*

Containerization is also known as OS-virtualization technology. It involved software layer creates several isolated spaces over one single OS, the host OS. No guest OS are therefore deployed. Each container (application and its dependencies) interacts with a unique container engine. Each start of a container is prompt with no time to load and launch such (inexistent) guest OS as for a VM environment.

Container isolation is based on the virtualization as defined above and alternatively or in conjunction with kernel security functions.

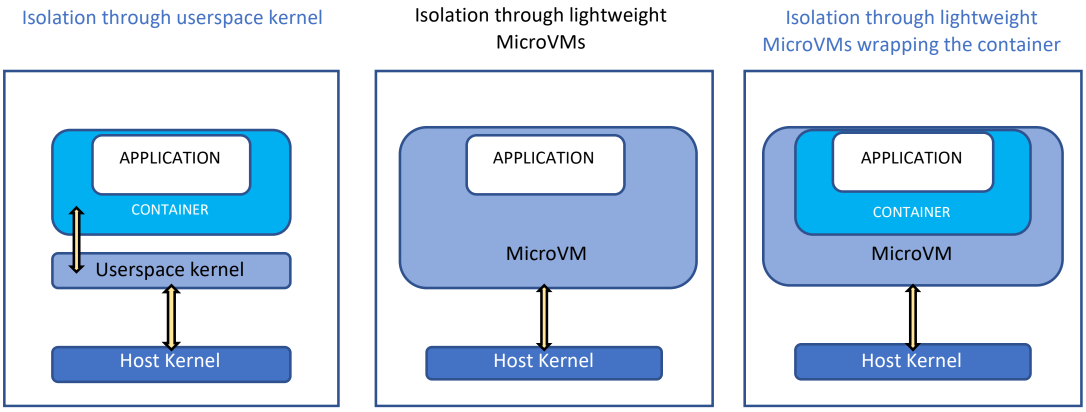

Isolation through userspace kernel
APPLICATION
CONTAINER
Userspace kernel
Host Kernel
Isolation through lightweight MicroVMs
APPLICATION
MicroVM
Host Kernel
Isolation through lightweight MicroVMs wrapping the container
APPLICATION
CONTAINER
MicroVM
Host Kernel

**Figure 17: Comparison of Container isolation approaches**

- Isolation by virtualization and optimization: To reduce the performance and memory resource heavy burden of virtualization, containers vendors design lightweight virtual machine OS, as originally designed by the unikernel concept and its bare minimal OS. The OS functional reduction entails possible conflicts with unavailable kernel functions required for the application or its dependencies.
- Isolation by Linux Kernel modules: In the Linux world, the latter are based on LXC (Linux Containers) kernel modules to allow the creation and running of multiple isolated Linux virtual environments on a single host. Container isolation is defined by leveraging *cgroups* and *namespaces* LXC security features which respectively allocate resource consumption ceiling and secluded user space. *Cgroups* controls and monitors system resource as CPU, memory and network according to a user defined policy while *namespace* attributes specific user id, process id, filesystem and network stack to a container. *Namespace* feature and more generally container do not rely on processor-based virtualization technology (e.g. Intel VT). In complement to LXC feature, another security step is taken by leveraging seccomp-bpf Linux feature which sandboxes a process in a system call restricted zone. Altogether, these features isolate both the container into its own exclusive memory space, limit its resource consumption and controls each container interaction with the host.

***Short survey of lightweight VM and secure container solutions.***

***IBM Nabla and Google's gVisor*** are two similar container technologies, offered for enhanced container security. Both adds a userspace kernel code to sandbox the container system calls (seccomp functionality). This code is capable to handle most of the system calls inside the container so that the pending system calls to the OS are limited in type and quantity. Both technologies need their specific runtime module (runnc and runsc respectively) to be installed on the machine.

***Amazon Firecracker and OpenStack foundation 's Kata*** are two similar **lightweight VM** technologies, delivering feature-restricted agile guest OS for instant start-up and low footprint. They are both developed in different language for security reasons and can also be considered for direct applications or containerized applications. Both are derived or directly using KVM hypervisor and leverage Intel VT hardware virtualization technology. With the emergence of lightweight VM, containerization is possible. The microVM is delivered with the interface with the container.

## 3.4 Integrity and remote attestation

Remote attestation is a technique that has gain momentum in Telco NFV environment because it generates trust and liability for the NFVI and VNFs. Indeed, this technology has been standardized by ETSI NFV-SEC group as a clear statement of intentions to be adopted. Remote attestation involves the use of the above mentioned TPM, and it extends the chain of trust outside of the execution platform to involve a trusted third party, who verifies that the conditions are still valid. Figure 18 shows the general concept, where the "Trust assessor" is in possession of a set of good known values or "golden values", that are nothing else than PCR registers stored in a database of the "Target platform". "Remote verifier" triggers the remote attestation to check the integrity and trust of the platform and upper layers (hypervisor and VNFs). This is as simple as request an integrity measurement report to the "target platform" and compare the values obtained with the golden values. This application remote attestation is possible thanks to the extensions defined by Integrity Measurement Architecture. If there is no match, the "remote verifier" will lose the trust in the platform and software. Who has the role of "Remote verifier" in the NFV ecosystem is multiple, from the NFVI provider to the tenant of the VNFs, to the Network Service provider, supporting multiples attestation?

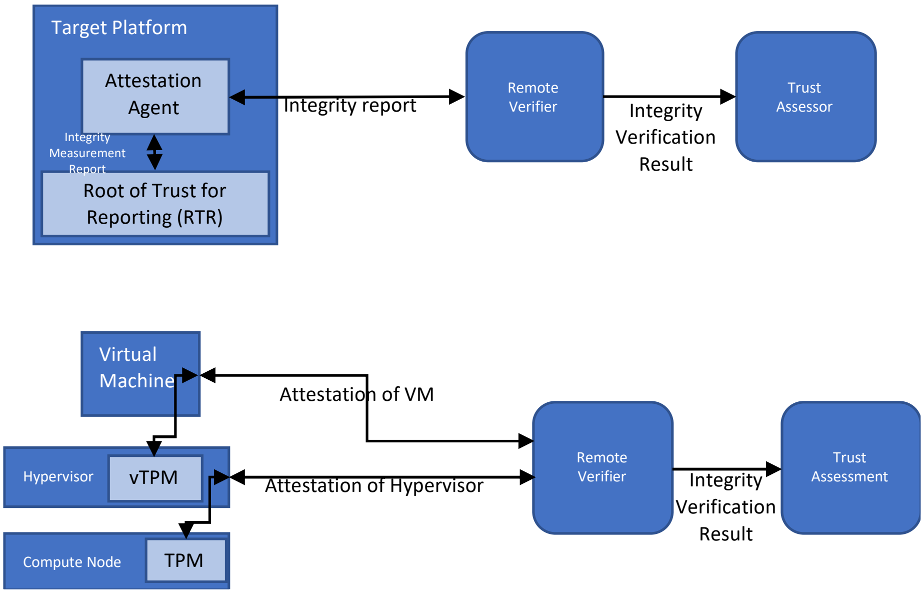


**Figure 18: Remote attestation for NFVI and VNFs.**

3GPPP adoption of the Service Base Architecture (SBA) and the microservices approach for 5G networks, has generated a lot of attraction in the Containers technology, i.e. dockers, mainly by its efficiency in resources demand and instantiation deployment. Precisely, the security exposure for this light virtualization technology, that share kernel functions, demands technologies to provide trust. There are already initiatives in progress to extend the remote attestation to the containers technology to address this lack of trust problems. One of the most attractive aspects for Remote attestation technology is that being based on TPM standard (currently in version 2) lead by Trust Computing Group, and not dependent on proprietary implementations, such as intel SGX Enclave or AMD trust Zones.

**Software and Hardware based attestation. The difference**

- Software solution can bring authentication service. Before starting a process, a call is made to a verification routine which produces the hash and decrypts the signature (associated with the code package) and compares them. A tampered code will not launch or at the cost of strapping the authentication routine. It is a first layer of security.
- TPM based authentication prevents such tampering and in addition creates a secure communication channel to deliver safely at a remote place (at the security management location) the unalterable evidence (using Diffie-Hellman asymmetric encryption based protocol) that the code is original. TPM based attestation delivers more security locally and a remote evidence of code correctness.

## 3.5 Remediation to introspection attacks by trusted execution environments

The Trusted Execution Environment (TEE) concept is a vast and highly documented subject. It rooted in kernel process isolation back from the 70's and emerged with hardware-based processor enabled techniques in the last decade. It actually defines a safe execution environment, bringing both confidentiality and integrity to code and data, in any opened and exposed standard IT execution environment and especially distant cloud operation. With hardware-based TEE, a malicious operator with root access on the Edge processing machine cannot reach the memory map of what is processed there. Code and data are fully secured.

If TEE are strong security enablers to consider, there are strong operational obstacles to put them in practice. These relate to the performance overhead, effort to setup, compilation requirement and access to source level code changes. Most importantly, TEE technologies are not compatible one with each other's. TEE-enabled software deployment must be carefully done (on targeted processors only). Intel TEE-enabled VNF will not run on one AMD board (TEE enabled or not).

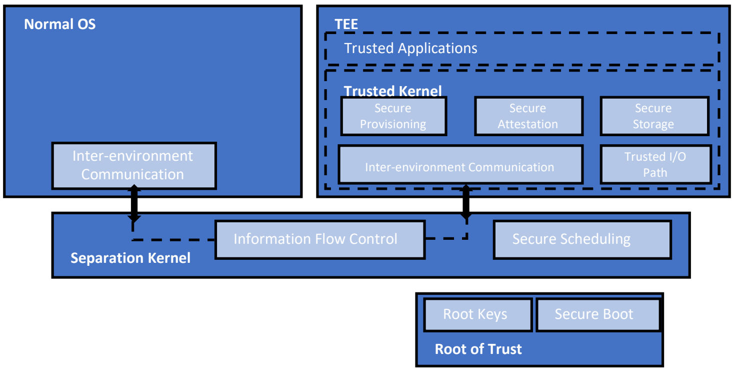


**Figure 19: Trusted Execution Environment**

Each processor vendor has its own definition with some overlaps on a restricted functional area from one solution to another.

When considering the SDN-NFV (and 5G core network and Edge Computing), Intel SGX[104] and SME-SEV[105] are the two first TEEs to consider, as brought to standard X86 architecture processors, capturing the entirety or a very large share (at the time of writing) of the cloud blade market. In its view of comparison of SGX and SEV, The Wayne State University[106] and their presentation at HASP, June 2018[107] reflected the two diverging approaches which rely **on two opposing architectural designs**. Intel SGX is depicted a means to secure small payload which must be preferably be an extraction of a reduced part of a larger code, whereas SEV is a basic VM encryption with no code extraction-selection to be made. Moreover, Intel‘s SGX interacts with user code (ring-3) while SEV operates on ring-0. When SGX imposes code changes (typically to remove all system calls) and a new compilation worked out through Intel’s SGX user SDK, SEV is totally transparent to the payload. When reading these elements, it is difficult to get more diverging techniques. In all respects (required code changes, size of the Trusted Computing Basis from a security-sensitive function or a complete VM with its operating system, offered security guaranties), SGX and SEV differ.

**Current trend related to TEE: Vendor-agnostic and easier workflow frameworks**
As one cannot foresee any technical convergence of SGX and SEV technologies, only a software abstraction layer (exposing common APIs to exploit both technologies) can bridge them. Software vendors and academics, as well as industry working group (Trusted Computing Group) had developed frameworks. As such *Asylo* and *OpenEnclave* abstract the TEE to remove dependency from the hardware. These frameworks are certainly to be considered as they break the two SGX-SEV separation, making it possible for a developer to reach a TEE execution in situation where she does not control which soldered processor is on the execution machine as it is the case for off-premises execution (cloud). As at the end of the day, the framework activates diverging technologies (offering different guaranties), a question remains if this valuable workflow facility is not adulterated with either a security loss or a performance loss, as one can foresee with any abstraction extra layer looking for the best of several underlying (diverging) techniques.

Because Intel SGX enclave implementation is relatively complex (and relatively scaring for a wildcard developer with no special expertise on security), several frameworks emerged as *Panoply*, *Scone* and *SGX-LKL*. These frameworks simplify the setup workflow, all sharing the same design idea of placing a micro kernel inside the SGX enclave to limit and control all interactions with the external world. This is motivated to shrink all developer work related to system calls as they are not permitted inside the TEE. They also remove the burden of selecting the correct section of code as the complete application is placed. However, the overhead impact is of at least 30%. On a pure security point of view, these frameworks deviate with Intel’s recommendations for the smallest TCB (i.e., the code inside the TEE), as they not only insert a complete un-touched application but associated with an external micro-kernel. They expose a large flank to vulnerability exploitations.

[104] https://www.intel.com/content/www/us/en/architecture-and-technology/software-guard-extensions.html

[105] https://developer.amd.com/sev/

[106] https://caslab.csl.yale.edu/workshops/hasp2018/HASP18_a9-mofrad_slides.pdf

[107] http://webpages.eng.wayne.edu/~fy8421/paper/sgxsev-hasp18.pdf

## 3.6 Conclusions on security

As a conclusion, we would like to stress the following points:

- Edge Computing security covers the same threats as core computing. However, the platforms are generally not offering the same security rich features and security policy and procedures. Edge Computing is more vulnerable typically on local introspection attacks.
- Virtualization technique domain is buoyant with a many competing emerging technologies for hardening containers and VMs, solving the equation of isolation versus overhead. Edge Computing will adopt one or several of these new technologies.
- Introspection attacks can be remediated by use of trusted execution, another active research domain, seeking for the best association of easy workflow (before compilation and at deployment), low overhead and security.
- Last, none of the previously stated security measures stop vulnerabilities (if present) to be possibly exploited. In particular, a vulnerable software inside a trusted execution environment is still as vulnerable (and its malicious execution hidden by effect of the trusted execution environment). All classical software verification and bug correction procedures apply for edge computing as for any other domain.

# 4. The Battle for the Edge

## 4.1 Edge Computing Ecosystem

The Edge Computing ecosystem involves a considerable set of stakeholders that either directly participate in or indirectly affect the provisioning of Edge Computing-enabled services towards the vertical customers. Leveraging on the general 5G actor role model introduced by the 5G-VINNI project[108] in line with 3GPP actor role model[109], we investigate the relationships and interactions among the different actors when it comes to service offerings that either involve Edge Computing services as part of a broader E2E service or focus only on the Edge. In our analysis, we consider that the service offering towards the vertical customer is an E2E network slice instance (if strict resource offering and isolation is required, aka NSaaS), or, when this is not the case, the service offering is a Logical Network as a Service (LNaaS) that includes MEC features. For the following discussion we do not go into the details of this difference and we use the notion of LNaaS in what follows, when not otherwise indicated.

Before describing the actor role model, it is important to highlight some fundamental business modelling concepts:

- A **stakeholder** is a party that holds an interest in the Edge Computing and in the 5G and beyond ecosystem.
- An **actor** is a party that consumes services or contributes to the service provisioning.
- An **actor role** is a specific well-defined function performed by an actor. An actor may perform multiple actor roles, while and actor role can be adopted by several actors.
- A **business relationship** is an association or interaction between two actor roles.

Figure 20 presents the main actor roles involved in Edge Computing-enabled services provisioning. The actor roles (blue rectangles) are grouped into “actor role clusters” (dotted rectangles) of several colors, while the potential business relationships are identified with blue arrows. Solid arrows reflect the money flow, while open arrows the service flow.

[108] 5G-VINNI report D5.1 “Ecosystem analysis and specification of B&E KPIs”, June 2019.

[109] 3GPP TR 28.801. Telecommunications management; Study on management and orchestration of network slicing for next generation networks.

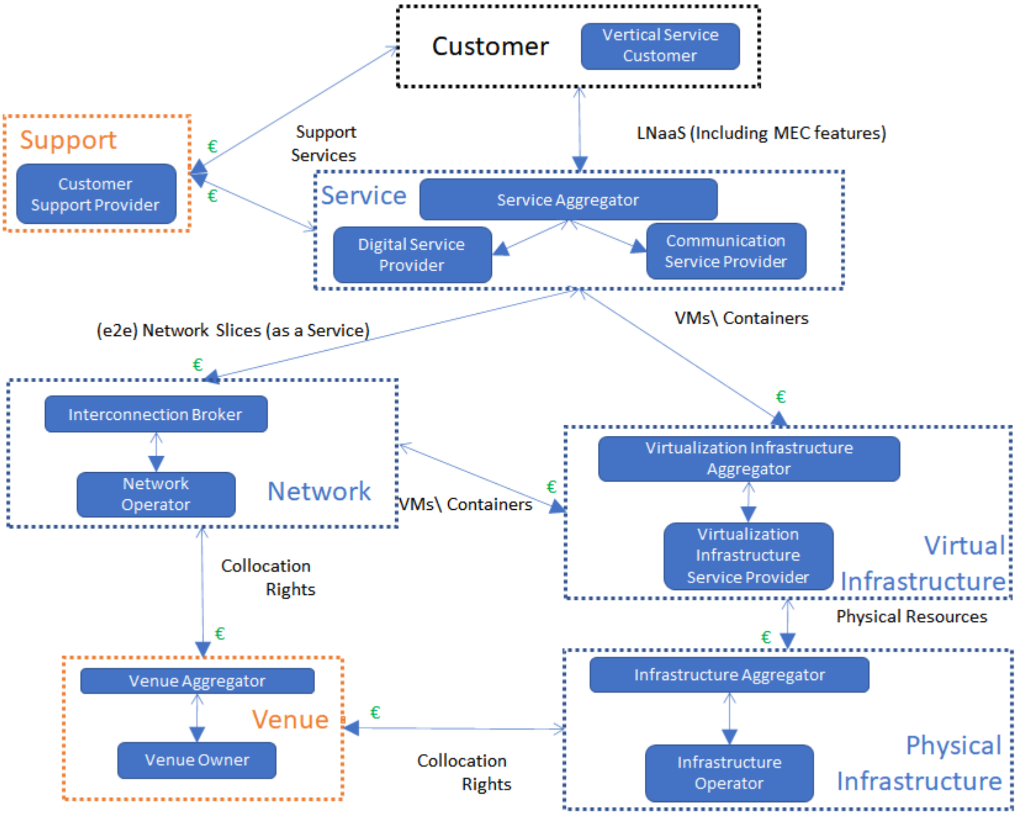


**Figure 20: Actor Role Model for the Edge Computing Ecosystem**

The actor roles are:

- **Vertical Service Customer (VSC)**, who represents a vertical company, organization or user that acquires the required communication and application services in order to support a qualified set of UE. In the most common scenario this role is adopted by an SME (Small to Medium Enterprise) doing business on a specific vertical sector, a large service provider that offers online application services, or even an end-user.

- **Communication Service Provider (CSP),** who offers communications services to VSCs through own/leased/brokered network. This role is often taken by a Telecommunications Network Operator or an MNO. A CSP takes advantage of network slicing and NSaaS or LNaaS concepts to offer communication services that can be either: (i) B2C, e.g., mobile data, voice and messaging, (ii) B2B, e.g. an uRLLC network slice instance connecting a factory with a remote operations centre, or (iii) B2B2X, e.g., roaming, RAN sharing, etc.

- **Digital Service Provider (DSP),** who offers online application/services to VSCs that usually require to be deployed on the Edge and consume computational resources. These application/services are specific to vertical industries, such as transportation, entertainment, eHealth, public safety, etc. For example, a company offering a real-time video analysis service that utilises AI techniques for identifying public safety incidents would fall into this category. In the example above the VSC would be a Public Safety organization, e.g., a police department. Similarly, to communication services, application services can be B2C, B2B or B2B2X.

- **Service Aggregator (SA)** or System Integrator, who bundles several communication and application services and sells these to VSCs. For instance, an OTT Service Provider that integrates communication services (e.g., mMTC network slices in order to connect a large set of IoT sensors) and application services for analysing data collected and *resells* the whole as an integrated, value-added service. This actor role materializes the concepts of (Edge cloud) **platform ecosystem** and **one-stop-shop** since the VSC has a single contact point to acquire a value-added service that may (transparently) involve the contribution of multiple actors of lower levels.
- **Customer Support Provider,** who offers technical, behavioural, economic, and legal consultancy services to VSCs or DSPs, as a facilitator for the faster adoption of 5G technologies and services.
- **Network Operator**, who designs, builds and operates a network for offering Layer 2 or Layer 3 network services, and can be further classified into Access, Transport, Core or Backbone Network Operator. A Network Operator provide services to CSPs in the form of network slice instances, and may expose functionalities related to monitoring, control and management, etc., towards external entities through APIs. A Network Operator may bundle its own services with virtualization infrastructure services offered by VISPs (below), in order to provision value-added services.
- **Interconnection Broker**, who has agreements with multiple Network Operators and combine network slice instances from different Network Operators to build and operate E2E network slices. Network Operators are well positioned to take this role, nevertheless, independent third parties running BSS/OSS functionality can adopt this role as well.
- **Virtualization Infrastructure Service Provider (VISP),** who provides virtualized infrastructure services, by utilizing the physical infrastructure offered by the Infrastructure Operators and Aggregators. A VISP designs, builds and operates its virtualization infrastructure(s), and offers its virtualized infrastructure services to other actor roles, such as Network Operators or CSPs. A VISP may offer virtualization infrastructure services ranging from multi-purpose VMs/Containers to complete virtualized infrastructure management solutions on compute, storage, network, IoT, etc.
- **Virtualization Infrastructure Aggregator (VIA),** who aggregates virtualized infrastructure services from multiple VISPs.
- **Infrastructure Operator,** who maintains physical infrastructure that includes Computing, Storage, Networking or IoT resources. This infrastructure can be at a local, regional or global level.
- **Infrastructure Aggregator,** who aggregates physical infrastructure and associated services from multiple Infrastructure Operators to achieve and extended coverage or presence.
- **Venue owner,** who manages a venue (e.g., lampposts, tall structures) where infrastructure (e.g., base station) may need to be established. This also applies to the deployment of physical data center infrastructure.
- **Venue Aggregator**, who has business relationships with several venue owners and simplify the process of finding the appropriate locations for deploying infrastructure.

## 4.2 Coopetitive Landscape

The proposed actor role model is setting the "big picture" of 5G and Edge ecosystem, focusing on actor roles that directly or indirectly provide offerings that involve Edge Computing capabilities. As already mentioned, each role may be adopted by different actors, e.g., a MNO may adopt the role of the Interconnection Broker, but he may also adopt the role of the SA, aiming towards federated Operator Platform for the Edge[110]. Similarly, a large VISP-Hyperscaler (e.g., Amazon) may also adopt the role of VIA, but it can strategically aim at adopting the role of SA who is responsible for operating the "platform" and is the contact point with the VSC. Considering the above, it is straightforward that multiple actors may have the incentive to take over key actor roles such as the SA or VIA that may give them a competitive advantage. However, there is also increased potential for collaboration among different actors that may lead to win-win situations. In this section, we study the coopetitive (cooperation & competitive) landscape that may arise when it comes to service offering that involve Edge Computing capabilities, focusing on the following three **key actors**: the MNOs, the Hyperscaler and the Local/Regional IT/Cloud Providers. Along with them, we consider also actors such as an Enterprise Customer, an Application Provider, a Consultation Service Provider, a Reseller and a Venue Owner. Note that in the illustration below, the actors will be represented by boxes of different colors and the adoption of a certain actor role will be identified by coloring the appropriate rectangles appearing in Figure 20.

We first introduce three value chain scenarios where one actor controls the customer relationship and supply chain. Next, we introduce two collaborative ecosystem scenarios where the actors are inter-dependent in their value creation and supply and the customer interface and operation is many faceted. The scenarios serve to illustrate how Edge Computing affects the possible evolution of market dynamics, ways of organizing services, roles, and how current actors may position in the roles; in future markets the different introduced scenarios may exist in parallel.

### 4.2.1 Competitive Scenarios

In the three scenarios below we investigate the case where one of the three key actors, driven by a competitive spirit, adopts multiple actor roles, in order to provide a complete LNaaS to the VSC. The actors end up having a more "prominent position" overall in the delivery system, and also serve as the main contact point for the VSC. These scenarios have characteristics of being value chains, where one actor has control of its supply chain. In the scenarios below we speak of specific actors as example, and hence, we use labels such as MNO *A*, etc.

#### Scenario 1: MNO maintains the prominent position

As illustrated in Figure 21, we assume that MNO A maintains a prominent position and customer relationship, having adopted multiple roles of high importance.

[110] Operator Platform Concept, Phase 1: Edge Cloud Computing, GSMA, January 2020.

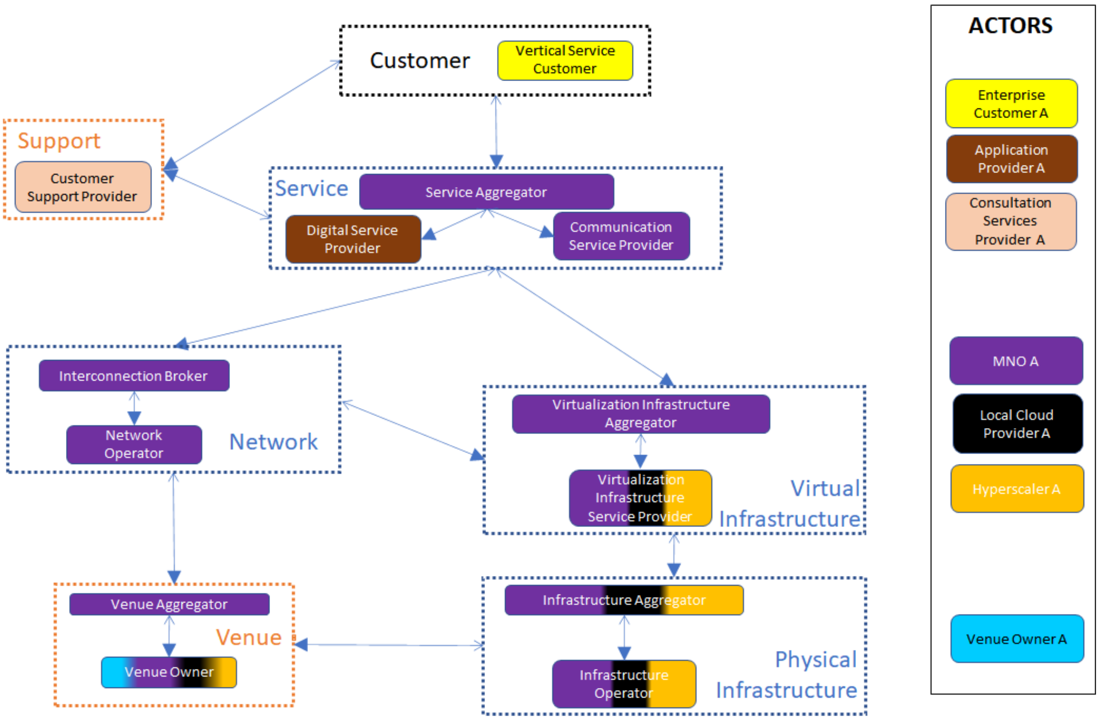


**Figure 21: MNO A maintains the prominent position.**

We also assume that all three key actors, i.e., MNO A, Hyperscaler A, and Local Cloud Provider A, maintain a physical infrastructure. In particular, each of them maintains its own physical infrastructure (i.e., datacenter resources) deployed to location(s) owned either by itself, another key actor or Venue Owner A. Given that MNOs traditionally interact with other venue owners in order to deploy their equipment in the appropriate geographic locations and structures, the MNO A, naturally, adopts the role of Venue Aggregator and thus has access and control over multiple venues. Taking advantage of this business opportunity, we assume that MNO A serves as facilitator for the deployment of other key actors' infrastructure in the venues it controls, of course, with the appropriate charging. All three key actors play the role of Infrastructure Aggregator, i.e., they aggregate physical resource that may belong to different Infrastructure Operators.

In the Virtual Infrastructure layer, we assume that all key actors adopt the VISP role, that is they build and operate virtualized infrastructure over the physical resources they control. When it comes to the aggregation of virtualized infrastructure that may be located to different geographic regions, the MNO A that also play the Network Operator role has again a competitive advantage, such as current presence in multiple locations along with transport network infrastructure already in place. In this scenario, we assume that MNO A exploits this competitive advantage to take over the VIA role and serves as the intermediate between the Service and Virtual Infrastructure layers, in fact "displacing" Hyperscaler A and Local Cloud Provider A from the Service layer.

MNO A controls the Service layer by adopting the SA role. That is, we assume that a strength of the MNO A is that it *operates a global platform*, where Edge Computing-enabled services are offered to the Enterprise Customer A (having the VSC role). The global reach and coverage are achieved by anticipated future federation and interconnection among partner MNOs. Note that the services offered by the SA may include Edge-provided applications developed by Application Provider A and communication services (i.e., network slices) provisioned by MNO A. Hence, the service is anticipated as a, potentially E2E, LNaaS/network slice that incorporates Edge Computing resource. Finally, the Consultation Service Provider A has a direct business relationship with the Enterprise Customer A, supporting Enterprise Customer A when interaction with the platform.

### Scenario 2: Hyperscaler maintains the prominent position

As shown in Figure 22, we assume that Hyperscaler A develops a prominent position, by taking over important roles at the Service and Virtualized Infrastructure layer and customer relationship. We focus our discussion on the differences between this scenario and the previously described Scenario 1.

We now assume that Hyperscaler A is more aggressive at the Venue layer, by also adopting the Venue Aggregator role. This means that Hyperscaler A can also now aggregate venues from different Venue Owners and then provide collocation rights to other actors over multiple locations. However, as discussed in Scenario 1, MNOs have a competitive advantage when it comes to interaction with venue owners, thus we expect that both MNO A and Hyperscaler A will remain active at this role. Thus, competition among these two actors may arise when it comes to venue aggregation.

Taking advantage of his experience on operating distributed and global cloud infrastructures, and leveraging upon concepts such as datacenter federation and hyperscale computing, we assume that the VIA role is taken by Hyperscaler A. Hyperscaler A aggregates virtualised resources coming also from MNO A (the virtualized RAN and 5G core infrastructures can be leveraged for this purpose) and Local Cloud Provider A and operates a wide coverage cloud infrastructure. In this way the Hyperscaler A can offer a seamless approach across the global cloud and the Edge cloud infrastructures.

At the Service layer, the SA role is now performed by Hyperscaler A, leveraging upon Hyperscalers experience with end-customers on offering self-service cloud services. In general, we foresee that Hyperscalers will push towards the adoption of a platform/service model that is similar to the traditional cloud computing services. MNO A still contributes to/complements the Service layer/platform through the CSP role and by offering network slices (potentially across domains) that enable Edge-provisioned service/applications in UEs with his network. However, MNO A does not directly interact with the Enterprise Customer A.

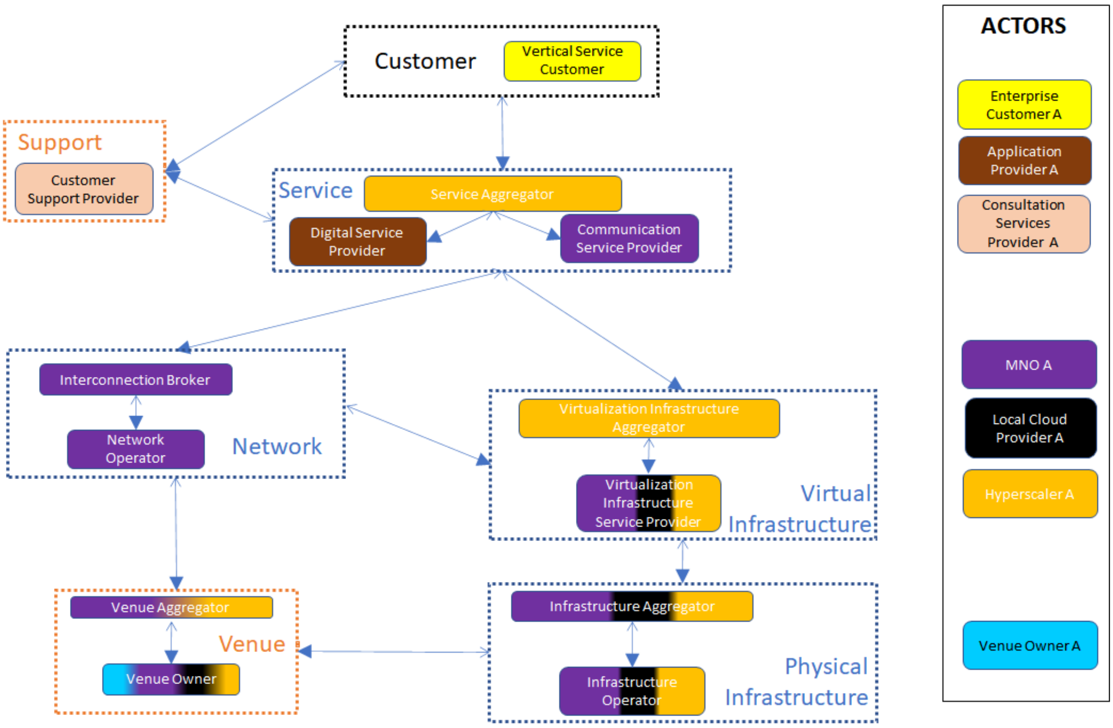


**Figure 22: Hyperscaler A maintains the prominent position.**

### Scenario 3: Local/Regional IT/Cloud Provider maintains the Prominent position

Figure 23 illustrates a scenario where the Local Cloud Provider A maintains the prominent position within its national region(s) and customer relationship. This scenario describes well the current market structures, however it is hard to foresee that it can scale at the same level as the previous two scenarios. Local Cloud Providers focus mostly on Edge cloud aggregation and vertical services provided on regional/local level, based on local and regional market specific strengths and knowledge.

We assume that Local Cloud Provider A adopts the Venue Aggregator role, aggregating multiple venues in a specific geographic region by having agreements with multiple local venue owners. Then, Local Cloud Provider A could also offer collocation rights to other actors. However, MNO A and Hyperscaler A still maintain the role of Venue Aggregators covering multiple regions. Thus, competition over venue aggregation may arise both at a global and local level. We also assume that Local Cloud Provider A takes over the VIA role aggregating and operating virtualised infrastructure in a local level. In such a scenario, Local Cloud Provider A could take control over MNO A’s and Hyperscale A’s virtualised infrastructure in a certain region.

We foresee that there is a potential for local platforms operated by local/regional Cloud Operators to be emerged. Hence, we assume that Local Cloud Operator A is actively involved at the Service layer, by operating a local platform utilizing Edge-provisioned applications provided by Application Provider A and communications services provider by MNO A.  Nevertheless, it may be difficult for a local platform to attract DSPs, and when it comes to services that involve multiple locations, collaboration with other actors should be established.

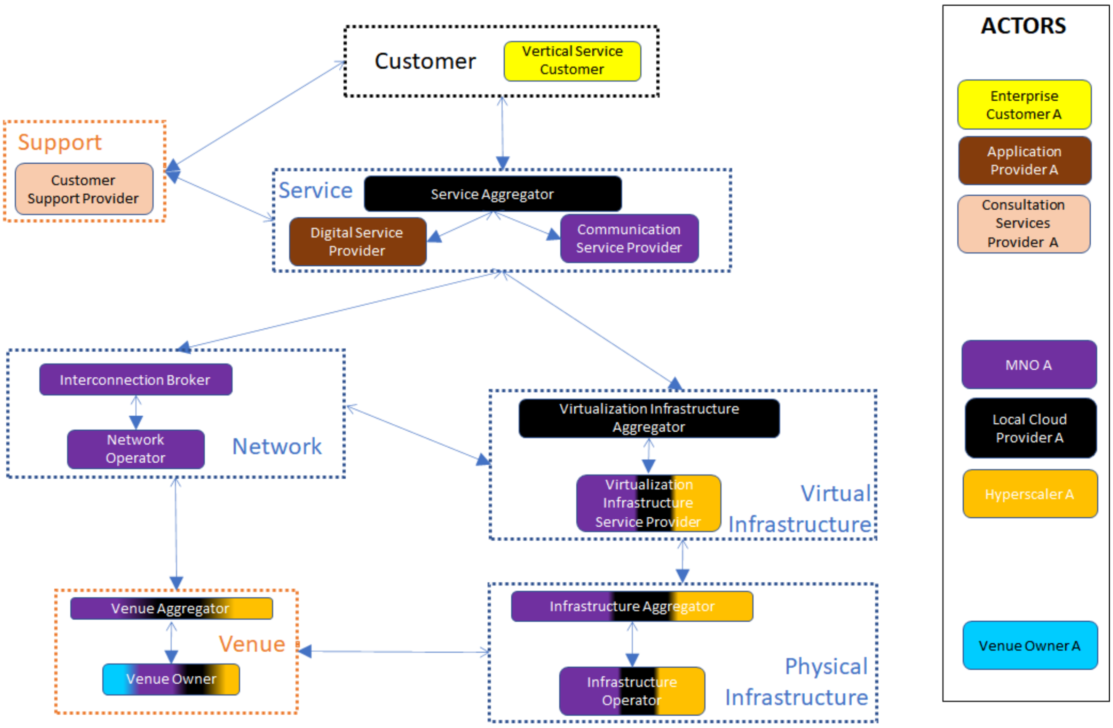


**Figure 23 Local Cloud Provider A maintains the prominent position, only possible at local level.**

### 4.2.2 Partially collaborative scenarios

In this section we investigate partially collaborative scenarios, i.e., scenarios between pairs of key actors. A meaningful scenario of partial collaboration would be between MNO A and Hyperscaler A, which acts as fully inter-dependent in providing the LNaaS to the VSC. In such a case, as shown in Figure 24, both actors collaborate at the venue aggregation, infrastructure aggregation, virtualised infrastructure aggregation and service aggregation level, building and operating a service platform that ensures global coverage. This scenario acknowledges the experience of MNOs in providing communications services and of Hyperscalers in delivering cloud services through platforms.

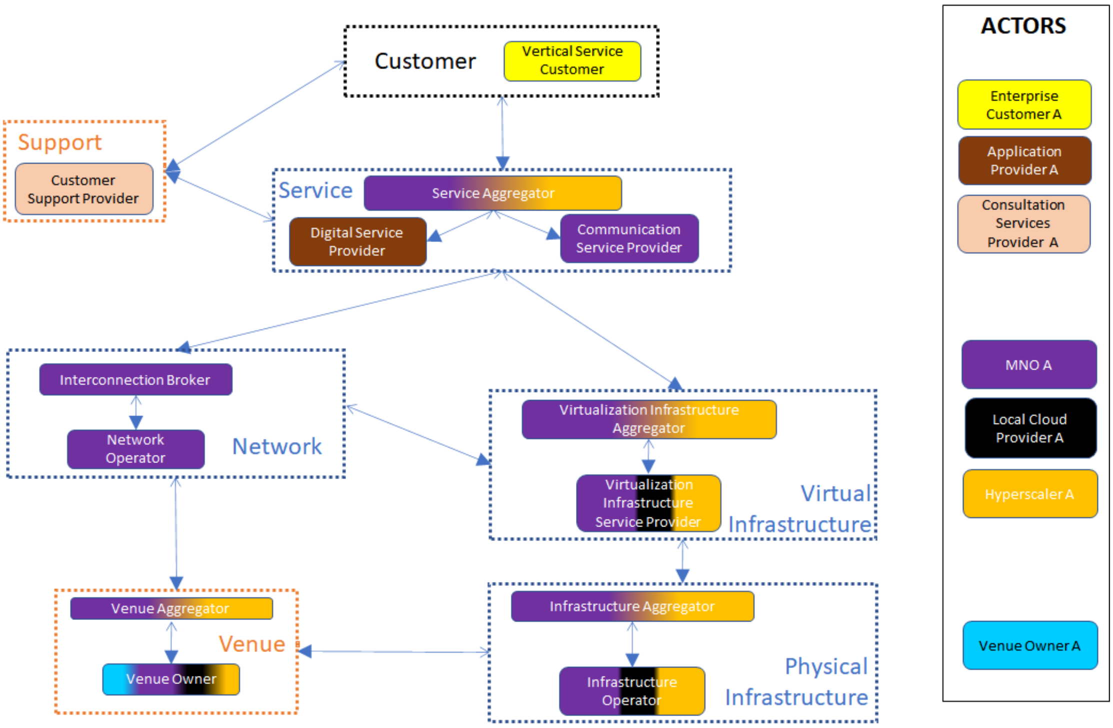


**Figure 24: Collaboration between MNO A and Hyperscaler A.**

A scenario where MNO A collaborates with Local Cloud Provider A could also make sense, if local MNOs succeed in joining forces with local Cloud Operators and take advantage of local presence in customer relationships. Also, Hyperscaler A and Local Cloud Provider A can be complementary and join forces to serve local customers; however, they would always be dependent on contribution of MNO A for applications/services that require communication services to be established.

### 4.2.3 Fully Collaborative Scenario

A market with full ecosystem characteristics will be achieved in a collaborative scenario where all actors are inter-dependent, cooperate and create a platform with multiple POPs, where VSCs can access global services. In this scenario, all three key actors play the role of SA, by operating different segments of the platform and being the customer contact point in different geographic locations and customer segments. We assume that the collaboration among the different actors is transparent to VSC who only use the platform to potentially establish a service of global coverage.

Figure 25 illustrates an example where MNO A, Hyperscaler A and Local Cloud Provider A play the roles of Venue Aggregator, Infrastructure Aggregator and Virtualized Infrastructure Aggregator. This means that none of the key actors follows an aggressive strategy taking full control of the customer relationships or one central platform. This leaves space for all actors to enter the market and address customers with LNaaS, provide resources, and pursuit collaborations.

In the Service layer, apart from three key actors, we assume that Consultation Service Provider A may also adopt the SA role serving mostly as reseller towards the Enterprise Customer A, while there is also a potential for Application Provider A to play the SA role and be the contact point for the Enterprise Customer A.

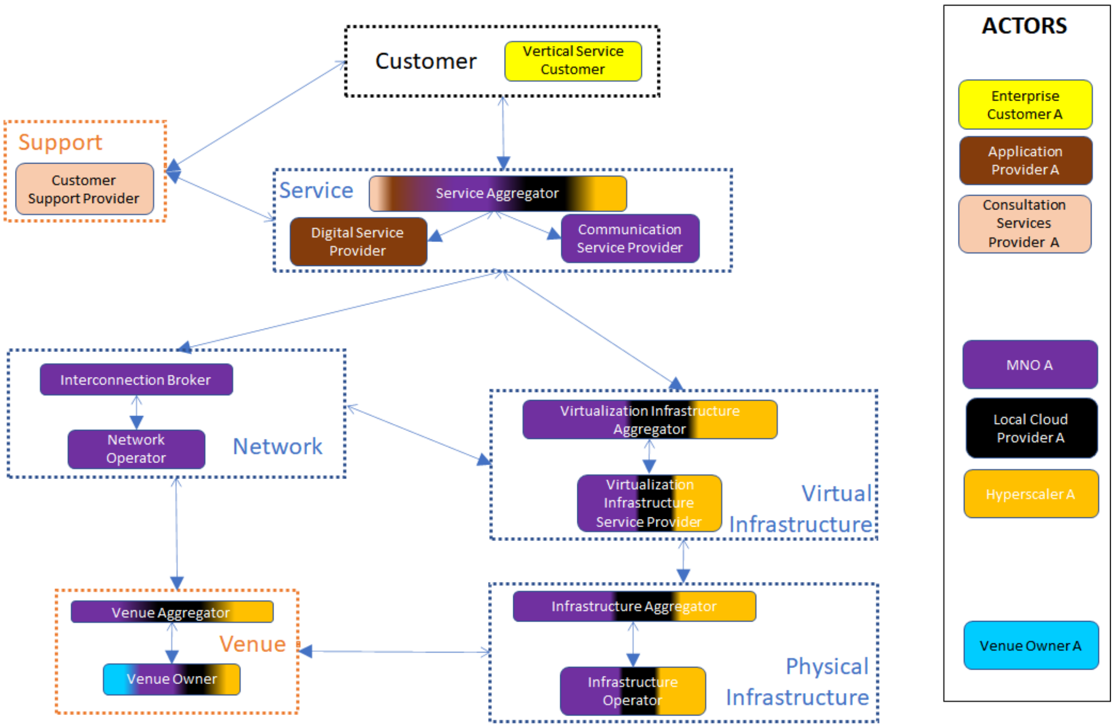


**Figure 25: Fully collaborative scenario, where multiple aggregator roles are shared among the different actors.**

### 4.2.4 Complementary players and mixed scenarios

While the above scenarios are considered as the main baseline for analysing and discussing the competitive landscape, collaboration and potential future partnerships, we also highlight complementary actor roles and partnership scenarios that complement and affect the above main scenarios. This is both to recognize that the above is coming from the point of view of the telecom sector and to acknowledge the strength and evolution of the hyperscalers / global OTTs. When entering a new area of 5G and beyond where the Edge Computing and Edge cloud services become an important and dedicated market, and an industry by itself, there are more players and aspects to consider in order to complement the above vision.

While the above scenarios are largely motivated by the ability and strengths by these actors to cater for and effectively enable customer management and service support, the below bullet points are key in introducing and discussing complementary factors and actors that can have large impacts on the Edge cloud ecosystem.

- Telco Network Equipment Providers (e.g., Ericsson, Nokia, Huawei, etc.) offering managed network and cloud services to the MNOs and Telcos. These actors have the opportunity to leverage their managed services for managing operator networks and evolve these capabilities into managed services for managing Edge cloud infrastructures and general services.
- IT solution providers (e.g. IBM, HPE, Dell, Oracle, etc.) offer cloud services, solutions and/or support services that have the potential to impact the ecosystem. These actors are strong in the enterprise office IT and Software market and we see solutions to address the emerging Edge cloud space.

Both of the above types of actors can help either MNOs / Telcos or Local Cloud (IT and hosting) providers in strengthening their position towards the vertical enterprise customers. However, the fact that currently Telco NEP collaborate with MNOs / Telcos, whereas the global IT providers have a strong relationship with the Local Cloud (IT and hosting) providers, can influence how these business relationships will evolve.

The venue location, condition and context of the Edge datacentre will also play a key role. Three categories of venues have distinct properties and contexts and will require separate analysis of strengths and conditions from a multi-actor PoV;

i) MNO RAN and base station venue
ii) Enterprise office indoor venue (enterprise office park), and
iii) Industry or factory venue

While i) is facilitating for and is part of the public network, iii) on the other hand is focused on non-public networks, while ii) in particular, is required to facilitate for a mix of public and private logical networks. In particular, iii) will facilitate for industrial, production operational technologies, solutions and networks, which typically includes and relies on strict time sensitive and deterministic networking which sets strict requirements towards the 5G and beyond services.

In the area of Factory of the Future (FoF) / Industry 4.0 there are large global players for industry equipment and solutions that will play an important role in the establishment of industrial indoor and non-public Edge computing (technology orientation) and Edge cloud (service orientation) solutions. The importance of time sensitive networking indicates that both these industry players, as well as Telcos putting emphasis on these capabilities, can become important players in this field. The new OT networks enabled by 5G and adjacent technologies and the new operator’s dashboards needed for such next generation OT industry appears to be a key area of technology and business development.

Looking further up the value stack and to the upper part of the general Actor Role Model for Edge Computing Ecosystem and addressing again the SA role, one may argue that this will not be simply a role played by a single player. One may speak of multiple roles needed in this area both addressing the aggregation of the general Edge Cloud Services and even more certainly there is a need of specific Vertical Service Aggregator Players address the dedicated needs of the specific verticals.

### 4.2.5 Collaborative evolution and alliances

From the above scenarios and discussion, one may conclude that there will not be just one winner to take it all. The need for cooperation and collaboration as well as development and evolution of numerous future proof APIs will be important. We will also see local, national and regional differences in how the different scenarios will be mixed, evolve, or dominate.

However, along with the need for collaboration, cooperation, and industry development, we expect alliances or multi-actor partnerships to appear. While the global hyperscalers have the strength and might drive such partnerships individually and shape how they collaborate with Telcos and other players, traditionally local Telcos will need a more collaborative and structured approach to the establishment of one strong alliance to enter the global scene. Again, in the field of Industry 4.0 the preferences and choices of the large and global players within the device, solutions and applications industry will have a large influence of the evolution of multi-actor partnerships and alliances. The traditional strengths of Telcos in enabling interconnection and the potential strength and ability of developing future oriented collaborative solutions (and offerings based on global reach and standardized global solutions) implies again uncertainty in how the various ecosystems and business models around Edge cloud and the verticals will evolve.

## 4.3 Emerging initiatives

We have so far provided a framework to start conducting a business analysis and an initial analysis of various scenarios where different service provider stakeholders have selected different strategic action plans. We can observe several industry driven initiatives taking place, and in particular the early partnerships between Telcos (Operators) and Hyperscaler platforms (Google Cloud, AWS, Microsoft Azure) have already been formed[111]. Moreover, we also see initiatives by large and global players in the industry equipment and solutions space, as they address industrial indoor and outdoor service and solution offerings and non-public Edge Computing.

Along with these per-stakeholder-driven initiatives it is also worth to notice the multilateral GSMA initiative along with their Future Network Programme. A central part of this programme is the Operator Platform concept and the Edge Cloud Computing that is in focus for their phase 1. GSMA envisages that operators will collaborate to offer a unified "operator platform" that will support federation among multiple Operators' Edge Computing infrastructure - "*… to give application providers access to a global edge cloud to run innovative, distributed and low latency services through a set of common APIs*."[112]. Recently, and as follow-up, GSMA released both the Operator Platform Telco Edge Proposal Version 1.0[113] and a Telco Edge Cloud Whitepaper on "Edge Service Description and Commercial Principles"[114].

Initiatives driven by the public side should also be noted. Recently, the European Commission sent out a press release welcoming the political intention expressed by all 27 Member States on the next generation cloud for Europe. It is pointed out that – "*Cloud computing enables data-driven innovation and emerging technologies, such as 5G/6G, artificial intelligence and Internet of Things. It allows European businesses and the public sector to run and store their data safely, according to European rules and standards*."[115] Alongside the expression of these goals, we also recognize the GAIA-X initiative driven first by France and Germany that want to create the next generation of data infrastructure for Europe, its states, its companies and its citizens[116]. Other initiatives, for instance the one driven by the BDVA association[117], are shaping the convergence of Data, AI and Robotics in the networks of the future, where Edge capabilities will play a pivotal role and will be instrumental to fulfil a smooth integration of different technologies.

The European Telecommunications Network Operators' Association (ETNO) also welcomes the public initiatives and emphasises the importance of how this can stimulate

[111] https://stlpartners.com/research/telco-edge-computing-how-to-partner-with-hyperscalers/

[112] Operator Platform Concept, Phase 1: Edge Cloud Computing, GSMA, January 2020.

[113] Operator Platform Telco Edge Proposal, Version 1.0, GSMA Whitepaper, 22 October 2020

[114] Telco Edge Cloud: Edge Service Description and Commercial Principles, GSMA Whitepaper, October 2020

[115] https://ec.europa.eu/digital-single-market/en/news/towards-next-generation-cloud-europe

[116] https://www.data-infrastructure.eu/GAIAX/Redaktion/EN/FAQ/faq-projekt-gaia-x.html?cms_artId=1825136

[117] https://www.bdva.eu/.

the industry and support the EU’s vision by building a pan-European cloud “federation” of interconnected cloud capabilities.[118] Furthermore, ETNO underlines – “*A resilient, efficient digital infrastructure is the necessary backbone of any trusted data sharing architecture. Cloud infrastructure will need widespread 5G and fibre networks that support data processing closer to the user, including edge computing. ... European telecom companies have a key role in investing and operating edge computing capabilities over their networks. This will offer a major alternative to the centralised cloud computing model operated by Big Tech.*”.

Apparently, the industry will see increased investments in the years ahead into cloud solutions in general and more specifically in enabling solutions for Edge Computing.

In summary and as a concluding remark, we do see the shaping of a quite complex landscape involving many stakeholders. While the hyperscalers will aim at collaboration while strengthening their position individually, the Telcos will find it important to drive collaboration and standardized solutions overall, where multilateral agreements can become the preferred and more effective approach. Potentially, this might evolve further only considering the technical level (enabled by industry association initiatives) as well as including business level collaboration agreements (industry alliance).

This whitepaper provides a holistic overview of all the technical topics to consider and insights into the maturity and evolution of the different technical areas. The assessment of technical maturity and judgements on what will be the smarter technical roadmap are crucial topics to be analysed in order to drive and settle the business level decisions, action plans and agreements needed in the years ahead.

[118] https://etno.eu/news/all-news/683:eu-telcos-welcome-cloud-declaration.html

# 5. Approaches to Edge Computing in 5G-PPP projects

This section reports on the various approaches to Edge Computing adopted by Phase 2 and Phase 3 research projects funded under the umbrella of the Horizon 2020 5G PPP Programme. The analysis is based on information collected through a questionnaire circulated among the participants of the 5G-IA's Trials Working Group in Q2/2020. In total, 17 projects participated in the survey.

In order to provide the right context, we start by summarizing in Section 5.1 the main use cases addressed by those research projects. For more details, one can refer to the deliverables and project websites listed in Annex 1. Besides, in this section we also provide a project taxonomy/clustering according to the key functionalities deployed at the Edge (e.g., AR/VR/Video processing/analytics, 4G/5G core functionalities) for the various use cases. In subsequent sections, we analyze the specific implementations carried out by the projects in terms of type of Edge Computing infrastructure deployed (e.g., ETSI-MEC, CORD-like; Section 5.2), location of such computing infrastructure (e.g., on-premise, street-cabinets; Section 5.3), technologies used (e.g., server type, acceleration technology; Section 5.4), and applications/VNFs hosted at the Edge (vEPC, vertical applications; Section 5.5). Each section reports on details at the project level and discusses the rationale behind technological decisions. For each section, we also provide a brief analysis of the survey results.

## 5.1 Use cases

This section summarizes the various use cases addressed by the research projects which replied to the questionnaire. It is worth noting that the phase projects belong to, defines notably different scopes and purposes: Phase 2 projects are primarily focused on research and innovation of key 5G concepts, whereas Phase 3 projects have emphasis on the validation of 5G technology for specific vertical applications. Complementarily, Phase 3 long-term evolution projects are more forward-looking, aimed at developing advanced concepts which are more difficult to demonstrate in specific applications/scenarios. Consequently, there are substantial differences in terms of the number of use cases (as far as this White Paper is concerned, we restrict ourselves to the three most relevant use cases, at most) considered by those projects, and their breadth and depth. Further, it is important to note that Phase 3 Infrastructure projects are aimed at building experimental research infrastructure to be used by other projects. Hence, their answers are more general, since their objective is to be as open and flexible as possible.

Figure 26 provides a taxonomy/clustering of research projects according to the key functionality placed at the Edge. This follows from the use case descriptions provided by each project in the Questionnaire.

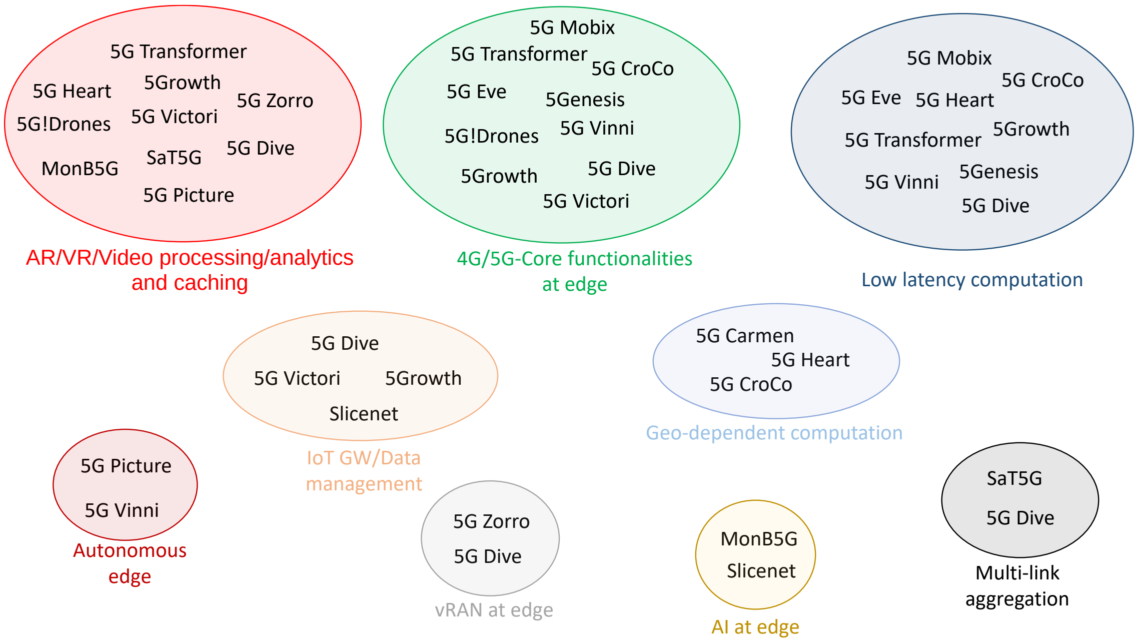


**Figure 26: Clustering of projects according to the specific key components in their respective use cases.**

The use cases clustering revolves around the following 9 key functionalities:

- AR/VR/Video processing/analytics and caching: Any kind of video processing or caching performed at the Edge with the aim of a faster computation of AR/VR, reduction of load at backhaul or other kind of video related processing requiring low latency.
- Low latency computation: Non-video applications located at the Edge in order to reduce the latency between the user and the application server.
- 4G/5G-Core functionalities at edge (e.g., PGW, UPF): Hosting at the Edge parts (typically, from the data plane) of the 4G or 5G core functions.
- IoT GW/Data management: Virtualized versions of IoT GWs hosted at the Edge as a mechanism to reduce load or pre-processing data.
- Geo-dependent computations: Championed by the automotive scenarios, this cluster includes the use cases which place functions at the Edge to serve a certain geographical region.
- Multi-link aggregation: The Edge as an aggregation point where multiple technologies can be used to connect to the core network.
- Autonomous Edge: The Edge as a mechanism to operate with low or non-existing backhaul, therefore typically hosting core functions to work in an autonomous way.
- AI functions at Edge: the Edge used to run AI functions leveraging contextual information available in the vicinity of the user.
- Virtual RAN (vRAN) at Edge: The Edge as a hosting platform for Virtual RAN functions.

Table 1 present the use cases that are being considered by each of the 17 projects. As it can been seen, the 5G PPP projects have used Edge Computing solution in multiple vertical sectors (e.g., Automotive and Transport, Industry 4.0, eHealth, smart cities, energy, etc.). This is to be expected as Edge Computing is identified as one of the most promising solutions to meet the vertical requirements (e.g., reduced delay).

Table 1: Use cases

| | | Use Case 1 | Use Case 2 | Use Case 3 |
|---|---|---|---|---|
| Phase 2 | 5G Transformer | Monitoring and emergency eHealth | Automotive extended virtual sensing | - |
| | Slicenet | Smart Grids | Smart lighting | eHealth 5G connected ambulance |
| | SaT5G | Edge delivery and offload for multimedia content and MEC VNF software | 5G fixed backhaul – broadband connectivity across a wide geographic | 5G to premises: connectivity complementing terrestrial networks |
| | 5G Picture | Smart city safety | Simple things virtual reality | Stadium use case |
| Phase 3 Infra | 5GEve | Industry 4.0: Remote AGV controller | Industry 4.0: Zero defect Manufacturing | - |
| | 5GVinni | Autonomous Edge | Mobile Gaming | URLLC use cases |
| | 5Genesis | Edge-based mission-critical services | - | - |
| Phase 3 Automotive | 5GCroCo | Tele-operated driving | HD mapping | Anticipated cooperative collision avoidance |
| | 5GCarmen | Green driving | Cooperative lane merging | Sensor and state sharing: back situation-awareness |
| | 5GMobix | Platooning with “See-what-I-see” functionality | Zero-touch border crossing | Remote driving |
| Phase 3 Adv. Trials | 5G Heart | Remote educational surgery | Remote surgery: load balancing via multicasting | Remote ultrasound robotics |
| | 5G Drones | Drones command & control with telemetry and video | 3D mapping and supporting visualization/analysis software | Connectivity extension & offloading during crowded events |
| | 5G Growth | Industry 4.0: Augmented zero-defect manufacturing | Transportation: safety-critical communications | Smart Grids: Advanced monitoring and maintenance support for secondary substation MV/LV distribution substation. |
| | 5G Victori | Immersive media services to travellers arriving at the train station | Digital future mobility | Smart energy metering for high-voltage |
| Phase 3 Long-term evol. | MonB5G | Zero-touch network and service management with end-to-end SLAs | AI-assisted policy-driven security monitoring & enforcement | - |
| | 5GZORRO | Smart contracts for ubiquitous computing/connectivity. | Dynamic spectrum allocation | Pervasive vCDN Services |
| EU-Taiwan | 5G-Dive | Industry 4.0: Digital twinning and zero-defect manufacturing | Autonomous drone scouting | IoT multi-RAT virtual GW |

## 5.2 Type of Edge Computing infrastructure deployed

In this section, the type of Edge infrastructure deployed by the various projects in Edge networks is analysed. Table 2 summarizes the answers given by the various projects, followed by further discussions and analysis in subsequent sections. Their answers are provided separately for projects belonging in Phase 2 and Phase 3 of the 5G PPP Programme.

**Table 2: Types of Edge Computing Infrastructure adopted by each project.**

| | | Fog Comp. | ETSI-MEC | CORD-like | Other |
|---|---|---|---|---|---|
| Phase 2 | 5G Transformer | | x | | x |
| | Slicenet | | x | | |
| | SaT5G | | x | x | |
| | 5G Picture | x | | | x |
| Phase 3 Infra | 5GEve | | | | x |
| | 5GVinni | | x | | x |
| | 5Genesis | | | x | |
| Phase 3 Automotive | 5GCroCo | | | | x |
| | 5GCarmen | | | | x |
| | 5GMobix | | | | x |
| Phase 3 Adv. Trials | 5G Heart | x | x | x | x |
| | 5G Drones | x | x | | x |
| | 5G Growth | | | | x |
| | 5G Victori | | x | | |
| Phase 3 Long-term evol. | MonB5G | | x | | |
| | 5GZORRO | | | | x |
| EU/Taiwan | 5G-Dive | x | x | | x |

## 5.2.1 Phase 2 projects

**5G-TRANSFORMER**[119] selected two approaches for MEC, namely ETSI MEC and generic Edge Computing deployments. On the one hand, the former allows complying with ETSI standards with high industrial support while exploiting the advantages of a full-fledged MEC architecture. As part of those benefits, 5G-TRANSFORMER exploited Radio Network Information Services functionalities for some vertical use cases to make decisions based on radio link quality. On the other hand, more generic Edge Computing deployments were carried out in the sense that the 5G-TRANSFORMER MANO stack also controls all the infrastructure at the Edge closer to the end users and considers it as part of the possible locations where VNF deployment can be done to fulfil latency-constrained service requirements. The goal of deploying one or the other was to comply with the requirements of the vertical services being deployed as far as low latency is concerned (e.g., automotive collision avoidance, AR-based eHealth emergency services).

The **SliceNet**[120] infrastructure is fully compliant with ETSI MEC specifications and has been used as an ETSI PoC. This framework manages E2E network slicing across all the different network segments of the infrastructure, namely, (i) enterprise network segment, where final users and vertical business are located; and (ii) RAN segment, providing coverage that final users via RAN front-haul interface. Edge Computing comprises physical devices located between the RAN and the datacenter. Edge Computing is connected to the RAN via back-haul interface and to the datacenter network segment via the transport network segment. Both Edge and datacenter locations support virtualization and containerization and they are controlled via a logically centralized management framework by making use of multi-zone support capabilities to decide where to deploy and migrate virtual resources. On top of this infrastructure, the project deploys softwarized 5G architectural components as services both at the Edge and datacenter locations. Usually, 5G Core VNFs are deployed at the datacenter and both 5G RAN VNFs

[119] http://5g-transformer.eu

[120] https://slicenet.eu

services and ETSI MEC VNFs services at the Edge. Even if both RAN and MEC VNFs are deployed at the Edge, they create a logical function chain where the traffic going from the RAN to the CORE goes through the MEC nodes, acting as a monitoring and control point for low-latency optimizations.

**SaT5G**[121] explored the integration of satellites in 5G network architectures, where they would become integral part of the 3GPP defined architecture. Edge computing in SAT-5G is used for media streaming applications and for multi-linking, where satellite is used in parallel with terrestrial paths to enhance broadband to premises.
For multi-linking, the project chose a CORD-like architecture mostly because of some requirements of the functions needed at the Edge. Indeed, unlike MEC, which is still in implementation phase, a CORD or CORD-like architecture is more suitable for hosting Network functions as well as resource hungry and transparent Edge services. In SAT-5G, most functions deployed at the Edge operate at layer three and four, hence our first choice was for a CORD-like architecture.

In **5G-PICTURE**[122], the emulated MEC solution had two main requirements: (i) low latency between devices for AR/VR application; and (ii) the creation of high throughput traffic between the nodes to demonstrate the FPGA based Time Shared Optical Network (TSON) used to aggregate fronthaul and backhaul at the Edge of the network and further distribute the links back at the central cloud network datacenter.

Based on these requirements and due to the lack of ETSI MEC availability, in the test network an emulated MEC solution was implemented. Different services and software components of the use cases were deployed at the Edge and central cloud datacenter similar to a Fog architecture yet not compliant to any existing standard.

This solution successfully provided the project with low latency communication between UEs and compute resources, while also prevented backhaul link capacity saturation for transferring raw video streams that were later used for analytic purposes.

### 5.2.2 Phase 3 projects: infrastructure

The distributed cloud is **5G EVE's**[123] general approach for meeting Edge Computing needs of the use cases supported by the project. 5G EVE postulates that the Edge Computing environments benefit from adhering to the same architecture, components and solutions used in the rest of the network, since that is the way for ensuring both contention in CAPEX and operational costs control to the Communication Service Providers and, as a consequence, to the whole ecosystem of players involved in crafting new 5G-enabled services leveraging the Edge. Therefore, 5G EVE is not in favour of ad-hoc Edge implementation solutions based on ad-hoc hardware, architecture or orchestration. Instead, the project encourages the extension of the central clouds to other locations (on-premises, Edge, regional clouds) using the same architecture and technology (i.e., same hardware, software and OSS systems).

[121] https://www.sat5g-project.eu

[122] https://www.5g-picture-project.eu

[123] https://www.5g-eve.eu

**5G-VINNI**[124] does not restrict to any specific Edge infrastructure type. As an ICT-17 project, each test facility has the freedom to include Edge infrastructure or not and implement in the way that suits their targeted experimentation and intent. 5G-VINNI Architecture v1 (D1.1) included a Research Item on Edge, which builds on ETSI MEC principles, but does not mandate that basis. In 5G-VINNI Architecture v2 (D1.4) a more prescriptive definition of Edge implementation will be provided but this will again be optional at a test facility and will not mandate any specific approach. 5G-VINNI takes 3GPP work as its basis, and notes that 3GPP TS 23.501 includes MEC natively with the 5G NR architecture, in particular allowing for the UPF to be distributed. In 5G-VINNI D1.4, work in SA2, SA6 EDGEAPP and ETSI MEC will be considered.

In addition, the 5G-VINNI Berlin and Luxembourg Experimentation Facility Sites implement an Edge-Central 5G Core Network functionality split, which is expressed as the split of the 5G system between Edge and central network, this being considered the most important item into establishing satellite as a reference technology within the 5G systems. In this context, in order to assure the connectivity to the 5G network though the different backhauls, the 5G Core Network is deployed with a functional split between the Edge and the core network. The Edge networks are considered as the best option to compensate specific limitation in the backhaul connectivity. This includes specifically delay and capacity limitations which are also considered the weakest points in having satellite in a convergent architecture. Additionally, having a wide distribution of Edge nodes, the strong characteristics of the satellite networks such as secure communication, global coverage, broadcasting/multicasting capabilities as well as the limited need of distributed terrestrial infrastructure. Because of the wide connectivity, the "communications on the move" (COTM) scenarios are easier to deploy with satellite then with terrestrial links.

**5GENESIS**[125]**:** Telefónica is the provider of the Edge Computing infrastructure in 5GENESIS Malaga Platform, and as Operator, is the owner of more than 1000s Central Offices in Spain. Mobile RAN infrastructure deployed by Telefónica is connected to Central Offices using Fiber as back haul to connect the RAN infrastructure to the transport network, where the Mobile core is connected at several PoP along the country.

From an operator's perspective, it makes sense from an economical point of view to concentrate compute resources in an aggregation point like Central Offices where several Base Stations are connected to, for sending the mobile traffic to the transport network.
In order to take advantage of compute resources at Central Offices, two possible technology options are available:

1) ETSI MEC Bump in the wire: ETSI MEC defines the solution for Local Break Out of traffic to Edge Applications terminating the GTP tunnels at the Edge Compute node, to be able to route traffic to local applications. Traffic that needs to go to PDN, is then encapsulated again in GTP tunnel and sent to the S-GW.
2) Deploy S/P-GW at Edge compute node: deploying partially EPC at the Edge Computing node, it is possible to terminate the GTP tunnels having the SGi interface in the Edge computing node. SGi interface is plain IP so it can be routed

[124] https://www.5g-vinni.eu

[125] https://5genesis.eu/.

easily to Local Applications (for Local Break Out) and to the PDN using simple IP routing.

CORD supports both approaches, as ETSI MEC SW stack and EPC can both be deployed at the Edge Computing infrastructure.

The 5GENESIS Malaga Platform has chosen the second option, due to the fact that a consortium partner is an EPC provider. Moreover, there exist a great variety of vendors and open source solutions, while there is a small number of entities that can provide ETSI MEC SW stacks, mainly in the commercial space.

Similarly, in the Athens 5GENESIS platform, COSMOTE is the provider of the Edge Computing infrastructure following the ETSI MEC approach. COSMOTE operates a hybrid 4G/ NSA 5G/ MEC testbed complemented with an Openstack-based SDN/NFV Cloud infrastructure with two flavors of MEC implementation i) via second SPGW and ii) via SGW-LBO.

### 5.2.3 Phase 3 projects: automotive

**5G-CROCO**[126] follows the “Automotive Edge Computing Consortium (AECC)” approach to Edge. The trial sites operated by Ericsson (France-German-Luxembourg Corridor, Motorway A9 (Germany), Montlhery (France), AstaZero[127] in Sweden) follow Ericsson commercial setups. At the moment it is OpenStack based for VNFs and Docker/Kubernetes for application servers. Ericsson plans to also use Cloud Native approaches for the future for VNFs, so there Docker/Kubernetes and many widely adapted tools around it (ref. Cloud Native Computing Consortium) will also be used for VNFs. So eventually VNFs and application servers will run and be managed by the same cloud software.

The trial site in Barcelona, operated by CTTC, I2CAT, Nextworks, and Worldsensing, also built upon well accepted open source solutions including OpenStack, ETSI OpenSource MANO combined with the 5GCity Neutral Hosting Platform, SONATA and the Service Orchestrator and Multi-domain Orchestrator for managing E2E Network Slice deployments across the target core and Edge domains.

In **5G-MOBIX**[128], an Edge Computing solution is implemented at different cross border trial sites. All sites implement a proprietary solution based on the vendor of choice (Ericsson or Nokia). In most cases this is based on an implementation with OpenStack for hosting VM-based or containerized applications. This implementation is carrier grade, facilitating both runtime applications and core components. Core components are based on 3GPP Rel 15. A local breakout is based on either PGW-U or an UPF (depending on a NSA or SA based 5G Core).

The Edge position selected are close to the gNBs in order to satisfy the strict latency criteria of the CCAM use cases. The main project focus is to analyse cross-border from a cellular network mobility viewpoint. Further, given the status of the work in standardization to define related 3GPP/MEC mobility concepts, the deployment of a

[126] https://5gcroco.eu

[127] https://www.astazero.com/the-test-site/about. Active Safety Test Area and Zero (AstaZero).

[128] https://www.5g-mobix.com

unified Edge Computing platform was not considered to be feasible and of priority. Those aspects where discussed in the project set-up phase and the design of the project has been decided accordingly.

### 5.2.4 Phase 3 projects: advanced trials across multiple vertical industries

**5G-HEART**[129] uses a combination of ETSI MEC and M-CORD-like approaches. 5G-HEART is a project spanning through several test-sites, including Finish 5GTN, Norwegian 5G-VINNI facility, Dutch 5Groningen and British 5GENESIS platforms. 5GTN selected ETSI MEC as the Edge Computing infrastructure allowing the project to develop evolutional dynamic MEC applications aiming for ultra-low latency and high bandwidth, all in real-time manner running within 5G. In addition, the Edge infrastructure also based on Fog Computing because the test network contains Edge servers that based on COTS technology (servers and devices) where we can run our own services.

At the Dutch 5Groningen platform, the M-CORD-like type of Edge Computing was chosen mainly due to the usage of commodity hardware, open source software and the communities behind open source projects. The openness allows for modularity and choice between different components depending on use case needs as well as easier switch between choices made.

The **5G!Drones**[130] project is an ICT19 (trial project) that aims at conducting trials implicating Drones on two of the ICT-17 trial facilities, namely 5G-EVE and Athens Platform of 5GENESIS. The consortium also plans to experiment drone’s usage and measure relevant KPIs on other 5G testbeds 5GTN and X-Network based in Finland. Implementations of Edge at 5G-EVE follow ETSI MEC specifications compliant with the 3GPP architecture. The 5GENESIS Athens Platform integrates Edge Computing infrastructure in various locations within its topology, for the deployment of Edge applications and Network Service components. More specifically, for the 5G!Drones trials two Edge Computing deployments of 5GENESIS have been exploited: The first one is based on the NCSR Demokritos 5G Amarisoft solution enhanced with lightweight Edge Computing capabilities that deployed at the Egaleo Stadium, while the second MEC deployment that supported 5G!Drones trials is operated at COSMOTE Academy campus and is based on production grade equipment. The 5GTN infrastructure uses Nokia vMEC, based on ETSI MEC. Finally, X-Network (ETSI MEC and FOG computing) provided by Aalto university, is composed of ETSI compliant MEC platform developed by Nokia and a set of Fog servers. Nokia vMEC was adopted due to its rich functionalities and its compatibility with other Nokia products available in the same facility. Meanwhile, Fog servers allow the deployment and the trial of new functionalities not available in the closed source Nokia vMEC (e.g., Edge services migration, container-based service orchestration).

**5GROWTH**[131] considers applying generic Edge Computing approach for the vertical pilots. The goal is to deliver traffic that requires low latencies to the vertical applications

[129] https://5gheart.org

[130] https://5gdrones.eu

[131] https://5growth.eu

running at the Edge, to comply with the low latency requirements of the vertical services (e.g., industry 4.0, railway transportation safety).

**5G-VICTORI**[132] architecture follows the ETSI NFV standards in order to provide the required services and functionality such as network slicing. This is extended to the Edge following the ETSI MEC principles. The Extended MEC (xMEC) hosting infrastructure includes Edge Computing functionalities involving virtualized MEC computing, networking and storage resources with the MEC NFVI being its overlay. xMEC provides a set of VNFs as well as access to communication, computing and storage resources to service functions of multiple domains in an integrated fashion and can accommodate all complex time critical functions, due to its physical proximity from the relevant network element. Therefore, the main drivers for choosing the ETSI MEC type of Edge architecture are: (a) compliance with the ETSI standards, (b) provision of compute as well as networking VNFs.

### 5.2.5 Phase 3 projects: 5G Long Term Evolution

One of the main objectives of the **MonB5G**[133] project is to design a scalable and secure architecture for the distributed management and orchestration of massive numbers of heterogeneous network slices. To this end, the project aims to provide distributed implementation of monitoring, analytics, and decision-making components with varying degrees of centralization. In this context, the Edge domain is regarded as a promising domain to deploy those kinds of services characterized by stringent delay constraints and/or high bandwidth requirements. Thus, the MonB5G architecture definition strictly follows the ETSI MEC standard guidelines to ensure compatibility and provide an E2E slice management solution suitable with the current telecommunication business scenario.

**5GZORRO**[134] envisions a multi-party distributed model for 5G through which a large group of parties can establish cross-operator/cross-domain service chains with security and trust. Regarding the Edge Computing scope, this architecture model is also applied, aiming to enable the integration and interoperation among different Edge resource owners. At the Edge, typically in street cabinets or in lampposts, one typically has constrained computing and networking platforms. These are automatically discovered from multiple owners, selected and configured to implement E2E service chains and cope with peak loads. By enabling Edge resource trading between different parties in an automated, trusted and secure manner, network slices can be extended on demand across the borders of administrative domains.

### 5.2.6 EU-Taiwan Cooperation

Edge and Fog Computing resources are considered within the **5G-DIVE**[135] project to support applications and services requiring very low latency and/or local processing and intelligence. The solutions developed within this project will build on top of the 5G-CORAL framework. This framework already envisages a hierarchical and integrated computing infrastructure spanning across multiple tiers, comprising clouds and central

[132] https://www.5g-victori-project.eu

[133] https://www.monb5g.eu

[134] https://www.5gzorro.eu

[135] https://5g-dive.eu

datacenters (DCs) on top, Edge datacenters (Edge DCs) in the middle, and Fog computing devices (Fog CDs) that are available locally in the access area. Finally, ITRI MEC prototype, called intelligent Mobile Edge Cloud (iMEC) will be integrated in the 5G-DIVE architecture and, later on, it will be used for the in-site trial of the Autonomous Drone Scouting vertical pilot. In summary, the Edge concept of 5G-DIVE is an integration of ETSI MEC concepts into OpenFog (now Industrial Internet Consortium) architecture.

### 5.2.7 Analysis of results

Figure 27 provides an analysis of the different typologies used by the various projects. Out of a total of 27 responses to the questionnaire, 9 used ETSI MEC, 4 were Fog-like, 2 CORD-like, and 11 used the 'Other' category. By 'Other' it is meant a vendor-specific platform provided by one of the project partners.

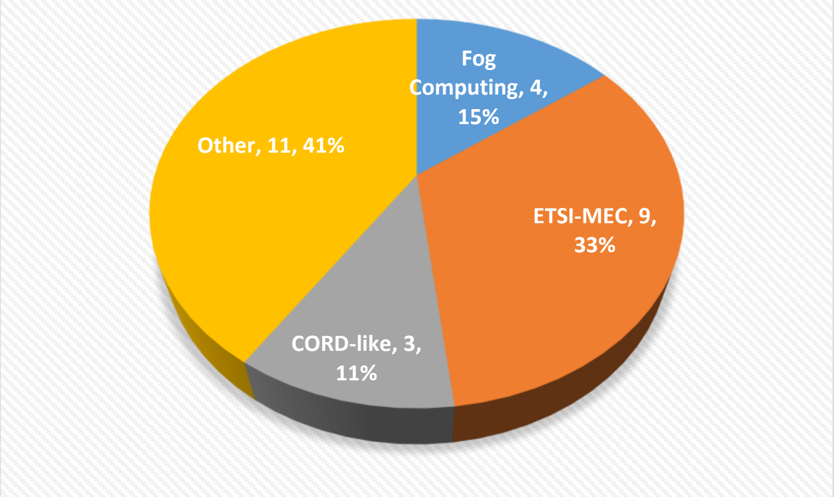


**Figure 27: Type of Edge Computing Platform.**

Fog Computing approaches account for a 15% of the answers and include projects adopting in the Edge concept computation capacity distributed in devices near the user or even in end-user devices.

It is worth noting that out of the Phase 3 Infrastructure projects, 5G-EVE reported Distributed Cloud as its Edge choice, 5G-VINNI reported that the project is Edge-type agnostic, so any kind of Edge can be used, while 5GENESIS declared the use of a CORD-like approach.

Finally, the prevalence of ETSI MEC over CORD and/or Fog approaches is clear in the European projects that replied to the questionnaire.

## 5.3 Location of 5G Edge Computing infrastructure

In this section we turn our attention on where the 5G Edge Computing infrastructure was deployed by the various projects. Again, we first summarize the findings in Table 3 and then provide a more detailed analysis in subsequent sections.

**Table 3: Location of edge infrastructure in each project.**

| | | On premise | B. Station / RAN | Fog Devices | Street Cabinet | Central Office | Micro Datac. | Private Datac. | Public Cloud | Other |
|---|---|---|---|---|---|---|---|---|---|---|
| Phase 2 | 5G Transformer | | x | | | | | x | | |
| | Slicenet | x | x | | x | | x | | | |
| | SaT5G | | x | | | | | | | x |
| | 5G Picture | x | | | | | | x | | |
| Phase 3 Infra | 5G Eve | x | | | | x | | x | x | |
| | 5G Vinni | | | | | | | | | x |
| | 5Genesis | | x | | | x | | | | |
| Phase 3 Automotive | 5G CroCo | x | | | x | x | | x | x | |
| | 5G Carmen | x | | | | | | | | |
| | 5G Mobix | | | | | | x | | | |
| Phase 3 Infra. | 5G Heart | x | x | | | | x | x | | |
| | 5G Drones | x | | | | | | x | | |
| | 5G Growth | | x | | | | | x | | |
| | 5G Victori | x | x | | x | | x | x | | |
| Phase 3 Long-term evol. | MonB5G | x | | | | | | x | | |
| | 5GZORRO | | | | x | | x | | | |
| EU-Taiwan | 5G Dive | x | | x | | | | x | | |

## 5.3.1 Phase 2 projects

**5G-TRANSFORMER**[136] features various use cases in which deployment at the Edge is required. In general, Edge VNFs are deployed next to the access network infrastructure offering coverage to the device being served (e.g., base station, access point). In this sense, Edge Computing is mostly deployed at the base station/RAN infrastructure, though it could also be deployed elsewhere in the operator infrastructure as long as latency requirements are fulfilled (e.g., micro-datacenter). More general scenarios including various computing platforms may, as well, be considered, e.g., private datacenters.

Three different testbeds are deployed in **SliceNet.** The Smart City use case has as Edge a micro-datacenter composed by only one or two nodes inside of a cabinet located on the top of the enterprise building. This cabinet has inside both RAN and Edge equipment and the antennas are directly installed close to the cabinet to provide coverage. The Smart Grid use case makes use of a Cloud-RAN deployment where the Edge and RAN are distributed across different locations with a 10-kilometer fiber cable. In this scenario, the Edge is composed by a microdatacenter where both 5G Centralized Unit (CU) and ETSI MEC are deployed and it is directly located in the telco premises. The Smart Heath use case is logically similar to the Smart Grid use case with the only difference that the Edge location is physically installed in a street cabinet rather than in the telco premises.

Differently, **5G-SAT**[137] Edge infrastructure is deployed at the RAN on the S1/N3 interface. The main reasons for this choice are: (i) backhaul traffic optimization and control, since the architecture is based on a hybrid backhaul network (Satellite and Terrestrial) ; (ii) transparency, because the project's Edge functions need to be as much transparent as possible to the network as well as the end users and, therefore, they need to understand at least S1/N3 protocols.

The 5GUK Test Network deployed in **5G-PICTURE**[138] is hosted at locations within the Bristol City Centre, while the cloud network was placed at the University of Bristol Smart

[136] http://5g-transformer.eu

[137] https://www.sat5g-project.eu

[138] https://www.5g-picture-project.eu

Internet Lab. The geographical spread of the nodes is within a couple of Km from each other and while using dark fiber for connectivity between sites, the location of the Edge servers made little difference to the latency observed in the service delivery. For this reason, the MEC architecture was emulated with VMs spread for the convenience of power and space at different locations similar to a Fog deployment. For the smart City Safety use case, the image processing node was performed at the "We The Curious" hosting site's IT room close to the end users in receiving the output for monitoring purposes. It should be noted that this service was deployed as Fog deployment and not all functions were at the Edge of the network.

### 5.3.2 Phase 3 projects: infrastructure

**5G-EVE's**[139] distributed cloud infrastructure can be deployed in any datacenter that provides the required infrastructure, generally at the Network Operator premises. However, the project envisages two exceptions: on-premise Edge Computing environments for large companies and the use of hybrid/public cloud for deploying Edge services.

**5G-VINNI**[140] is made up of multiple experimental sites, each of which is free to select its own architectural topology based on its own design and the requirements of the experimenters that wish to use each facility's network. As a consequence, MEC deployments vary from site to site. Many sites' infrastructure is housed in a single building, and, hence, it is a mixture of Central Office, Micro datacenter and Private datacenter. The 5G-VINNI Berlin and Luxembourg Experimentation Facility Sites implement an Edge-Central 5G Core Network functionality split, which is expressed as the split of the 5G system between Edge and central network, this being considered the most important item into establishing satellite as a reference technology within the 5G systems. Another Edge Cloud implementation is for Fish Farming, where analytics applications are deployed in an Edge Cloud that is connected to the 5G CPE in order to reduce the high uplink requirement. In this case there are no 3GPP functions deployed in the Edge Cloud.

The Málaga platform in **5GENESIS**[141] considers two different types of deployment, namely,

- Deployment of RAN in the campus of the Málaga University (UMA): This deployment consists in distributing 4 Remote Radio Head (RRH) units for 5G and 4 RRHs units for 4G connected to a Mobile core installed ad hoc for the project in the UMA campus.
- Deployment of RAN in the city center: This deployment consists in distributing 6 RRHs for 5G and 5 RRHs for 4G connected to Telefónica Commercial Network and to the Mobile core in UMA campus.

In order to fit both scenarios, the Edge Computing node deployed by Telefónica is located at the UMA campus connected to Telefónica Central Office. With this deployment, UMA RAN is connected directly to the Edge Computing node, as both of them are located in the same building, whereas the City Centre RAN is connected to Telefónica Central

[139] https://www.5g-eve.eu

[140] https://www.5g-vinni.eu

[141] https://5genesis.eu

Office using MOCN to connect to both Cores, i.e. the Telefónica Commercial core and the UMA Mobile core.

The 5GENESIS Athens Platform integrates Edge Computing infrastructure in three locations within its topology:

- **Site 1: The campus of NCSR "Demokritos"**. NCSRD is directly connected to Greek Educational, Academic and Research Network (GRNET)[142], which provides access to Internet and GÉANT (pan-European data network for the research and education community). This site will be responsible for hosting most of the infrastructure required for the management, orchestration and coordination of the Athens platform.
- **Site 2: The COSMOTE building (OTEAcademy)**, is also directly connected to GRNET which provides for access to GÉANT. This site will host infrastructure components, radio access components and NFV/Edge Computing infrastructure.
- **Site 3: The stadium of Egaleo (Stavros Mavrothalasitis)**, the location's connectivity is based on a wireless point-to-point link to NCSRD. This site will host infrastructure components that will allow the experimentation and support of use cases related with the Edge Computing, and Control Plane – User Plane separation in a realistic environment.

### 5.3.3 Phase 3 projects: automotive

**5G-CROCO[143]** firmly believes that upcoming network deployments will be very complex since MNOs and other stakeholders, e.g., Road Traffic Authorities (RTA) have different deployment options. The trial setups attempt to reflect such plurality with local packet cores and application servers directly at the trial site in street cabinets (Motorway A9, Barcelona, Montlhery, AstaZero), in private datacenters close to the trial sites (France-German-Luxembourg large-scale trial site, Barcelona, AstaZero), a central datacenter at Ericsson Germany (France-German-Luxembourg large-scale trial site, Motorway A9, Montlhery), and public clouds (or similar hosting on public Internet) available to all trial sites.

The Edge solution in **5G-MOBIX[144]** is deployed at a distributed site where the traffic from several radio sites is received in the commercial network. This site is used to aggregate radio traffic from several radio sites and redirect this traffic to the Core network. The system is deployed as a virtualized infrastructure comprising a full-fledged 5G EPC with and without LBO (PGW-U) at the Edge.

### 5.3.4 Phase 3 projects: advanced trials across multiple vertical industries

**5G-HEART[145]** uses different implementations depending on the specific use case. The different deployment options considered are (i) private datacenter; (ii) on premise data-

[142] GRNET http://grnet.gr

[143] https://5gcroco.eu

[144] https://www.5g-mobix.com

[145] https://5gheart.org

center/micro data-center; and (iii) micro-datacenter collocated or connected to RAN elements.

The **5G!Drones**[146] project leverages 5G trials facility for testing scenarios and evaluating KPIs involving Drones. In these facilities dedicated for testing and experiments the Edge infrastructures are deployed on premises for the following reasons:

- Availability of computing resources near to the deployed eNBs/gNBs.
- Availability of dedicated high-performance transport network within the facilities.
- Security concerns.
- Facilitating potential manual interventions.

In **5GROWTH**[147] Edge VNFs can be deployed either next to the access network infrastructure offering coverage to the devices being served (e.g., base station, access point) or within a private cloud/Edge infrastructure of the verticals at the vertical premises. In the former case, it is mostly deployed at the base station/RAN infrastructure provided by the operators shared among the private network of the vertical with the operator public network, though it could also be deployed elsewhere in the operator infrastructure as long as latency requirements are fulfilled (e.g., micro-datacenter). In the latter case, the Edge infrastructure is a private cloud infrastructure belonging to the verticals.

The **5G VICTORI**[148] project comprises four different testbed facilities, i.e., 5G-VINNI (Patras), 5GENESIS (Berlin), 5G-EVE (Alba Iulia) and 5G UK (Bristol). Each of these facilities provides different capabilities. However, in general, each facility is equipped with an on premise, private, micro datacenter, which is hosted at the premises of each testbed responsible organization. In addition, street cabinets and base stations are used in some of the facilities to host the Edge infrastructure even closer to the end users. Specifically, in Patras and in Bristol, “pop-up networks-in-a-box” will be deployed at certain locations, physically located inside street cabinets or on-prem IT rooms, which will provide 5G RAN connected with a local micro datacenter.

### 5.3.5 Phase 3 projects: 5G Long Term Evolution

The need to satisfy a multitude of heterogeneous slice-specific requirements, while guaranteeing slice isolation, demands for accurate vertical service deployments such that networking and computing resources will not be wasted. In the mobile network context, this often translates in deployment of services as closer to the end-users as possible. The setup of the Edge infrastructure at base-station (RAN) level will serve this purpose. At the same time, specific use-cases might require wider coverage areas or dedicated deployments on premises. Nevertheless, **MonB5G**[149] envisions vertical service migration towards private (operator-owned) datacenters as a means to overcome the limited resource availability of the Edge platforms, e.g., in case of traffic congestion or pro-active resource allocation during the phase of slice on-boarding. The MonB5G project will not

[146] https://5gdrones.eu

[147] https://5growth.eu

[148] https://www.5g-victori-project.eu

[149] https://www.monb5g.eu

strictly focus on a single deployment option, but rather consider several of them in order to support dynamic slice setup and reconfiguration in multiple scenarios.

Differently from MonB5G, in **5GZORRO**[150], two main types of locations are considered for the deployment of Edge infrastructure. The reasoning behind such selection is based on two main criteria: site availability and compatibility with use cases' requirements. In particular, street cabinets and micro datacenter, with the inherently reduced edge-compliant capacities, are available as part of the smart city IT infrastructures deployed in the 5GBarcelona facility. This deployment provides a minimal distributed Edge Computing ecosystem, where the presence of multiple stakeholders, controlling different Edge resources, are emulated in order to realize the considered use cases.

### 5.3.6 EU-Taiwan Cooperation

The computing substrate shall essentially include Edge and Fog Computing resources to support applications and services requiring low latency and/or local processing and intelligence. In **5G-DIVE**[151], this includes resources at the Edge of the network infrastructure (such as, private datacenters) as well as Fog Computing resources on the premises (such as, user equipment, customer premises equipment and other resources with limited computing capabilities).

### 5.3.7 Analysis of results

The responses from the different research projects show a high preference for the On-premise deployment of Edge solutions, mostly in Private Datacenters. A large number of projects also consider the deployment of Edge hardware in the Base Station or RAN, followed by its deployment in Street Cabinets. Basically, this is consistent with the nature of the use cases considered in the different projects. On the one hand, research considering Industry 4.0 scenarios, where it makes sense to constrain the deployments to company premises, are the typical example of a scenario deployed on a private datacenter on premise. On the other hand, several projects considering city-wide deployments are more focused on micro-datacenters deployed at Street Cabinets or Central Office. This is in stark contrast with the low number of projects considering Fog devices, since only one project is devoted to this kind of scenarios. Finally, it is also important to note the clear preference for micro-datacenters deployments versus the use of public cloud approaches.

[150] https://www.5gzorro.eu

[151] https://5g-dive.eu

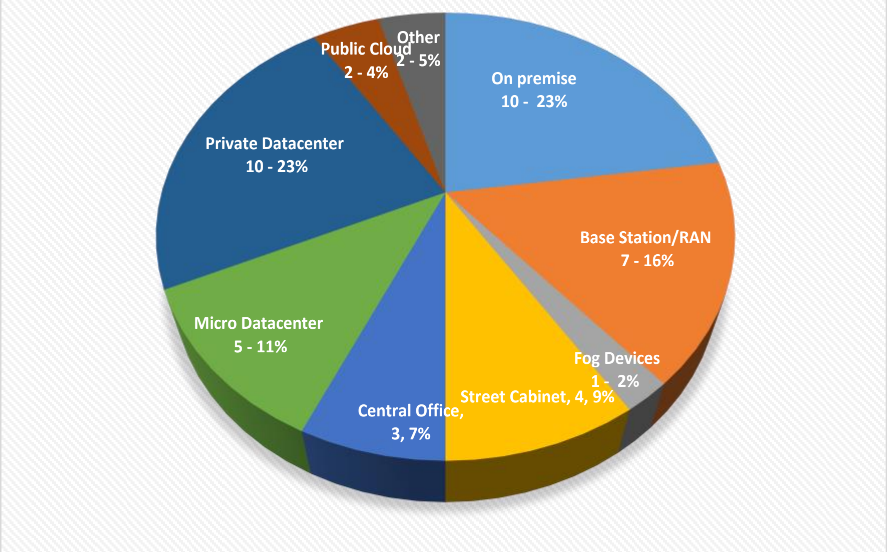


**Figure 28: Location of Edge computing infrastructure.**

## 5.4 Technologies used for Edge Computing

After discussing in previous sections, the type and location of 5G Edge Computing infrastructure, this section delves into the specific technologies underpinning such deployments, as Table 4 illustrates.

**Table 4: Main technologies adopted in Edge computing deployments by each project.**

| | | Server type | | | Acceleration | | | Device type | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | X86 | ARM | AMD | GPU | FPGA | Other | COTS | Constr. | Other |
| Phase 2 | 5G Transforme | x | | | x | | | x | | |
| | Slicenet | x | | | x | x | x | x | | |
| | SaT5G | x | | | | | | | | |
| | 5G Picture | x | | | | x | | | | |
| Phase 3 Infra | 5GEve | x | | | | | | x | | |
| | 5GVinni | | | | | | | | | |
| | 5Genesis | x | | | | | | | | x |
| Phase 3 Automotive | 5GCroCo | x | | | | | | | | |
| | 5GCarmen | x | | | | | | | | |
| | 5GMobix | x | | | | | | x | | |
| Phase 3 Adv. Trials | 5G Heart | x | | | x | | | x | | |
| | 5G Drones | x | | | | | | x | | |
| | 5G Growth | x | | | x | | | x | | |
| | 5G Victori | x | x | | x | | | x | | |
| Phase 3 Long-term evol. | MonB5G | x | | | | | | | | |
| | 5GZorro | x | x | | | | | | | |
| EC/Taiwan | 5G-Dive | x | | x | x | | | | x | |

| | | Orchestration plaform | | | | | SDN/NFV | | | Container orchestrator | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | OSM | ONAP | O. Baton | Cloudify | Other | ONOS | ODL | Other | OpenStack | Kubernetes | Other |
| Phase 2 | 5G Transforme | x | | | x | | x | x | x | x | x | |
| | Slicenet | x | | | | | x | x | | x | x | |
| | 5G - SAT | x | x | x | | | | | | x | x | |
| | 5G Picture | x | | | | | | x | | x | | |
| Phase 3 Infra | 5GEve | x | x | | | x | | | | x | x | |
| | 5GVinni | | | | | | | | | | | x |
| | 5Genesis | | | | | | x | | | | | x |
| Phase 3 Automotive | 5GCroCo | x | | | | x | | x | | x | x | |
| | 5GCarmen | | | | | | | | | x | | x |
| | 5GMobix | | | | | | | | | x | x | |
| Phase 3 Adv. Trials | 5G Heart | x | | | | x | | | | x | x | x |
| | 5G Drones | x | | | | x | | x | | x | x | x |
| | 5G Growth | x | | | x | | x | x | x | x | x | |
| | 5G Victori | x | | | | x | x | x | x | x | x | |
| Phase 3 Long-term evol. | MonB5G | x | x | | | | | | | x | x | |
| | 5GZORRO | x | | | | | | x | | x | x | |
| EU/Taiwan | 5G-Dive | | | | | x | | | | | x | x |

### 5.4.1 Phase 2 projects

The **5G-TRANSFORMER** stack allows to integrate any kind of technology out of those listed in Table 4 as long as the corresponding plug-in is developed. The marked technologies are those for which a plug-in/wrapper was developed during the lifetime of the project, though further support is ongoing in follow-up projects, e.g., **5GROWTH**. X86 servers were selected because of availability and familiarity. COTS devices were used as interfaces in the nodes, UEs in the tests, etc.

The **5G-TRANSFORMER** stack can also integrate any MANO project as long as a wrapper is defined to translate between the ETSI-compliant specifications that the **5G-TRANSFORMER** service orchestrator uses and the APIs of the corresponding MANO project. Due to functionality offered and community critical mass, two MANO platforms were integrated, namely OSM and Cloudify.

Likewise, the **5G-TRANSFORMER** Mobile Transport and Computing Platform (MTP) can integrate a plethora of heterogeneous transport and computing technologies through the corresponding plug-ins: ONOS, ODL, Ryu, and ABNO were the ones integrated due to functionality offered and their availability in the labs of partners that were familiar with them. This is also based on past developments of use from previous projects (e.g., 5G-Crosshaul). OpenStack and Kubernetes were also deployed within the project to explore different options to deploy VNFs.

As for the **SLICENET** project, it implements the same management plane for both Edge and datacenter in all its deployments. This is significantly different from other proposals where there is a complete management plane for the datacenter and another one for the Edge. SLICENET testbeds are based on OSM over OpenStack with either OpenDaylight or ONOS. The project also makes use of Kubernetes but mainly for the deployment of the management functions. The Edge is then seen as a geographical area in the management plane. Such area is composed by a set of X86 COTS servers that may have any type of acceleration. The Smart Grid use case requires GPU acceleration in the Edge to deal with Edge AI. Also, the project relies on FPGA and NPU acceleration in the network cards located in such servers. Then, the 5G VNFs are deployed as indicated in the previous question.

The focus of the **SaT5G** project is on satellite integration and not particularly on NFV / SDN development, although in the course of the project, partners virtualized multiple satellite specific functions. For this reason, partners preferred to use a platform which was not too demanding on computing facilities, straightforward to download and deploy, and with lots of support on fora, etc. Therefore, a decision to utilize OpenStack for the MEC part was made, although the project also used Kubernetes docker / container for virtualizing some core network components. The project also aimed at developing desktop demos to prove specific principles. The demos are typically based on 4 Intel NUCs, each with Intel i7 processor, 32GB of RAM, and 64 GB SSD. Devices are connected via gigabit switch. 4G base station and UE are connected via SDR (Software Defined Radio) boards, which are connected via cables instead of antennas. This was done so we could use the same frequencies as network operators would use, and to avoid actual transmission of radio signals, and their impact on commercial networks.

For the Smart City Demonstration, several x86 bare metal servers were used in **5G-PICTURE** to host the virtual infrastructure to ensure compatibility and compute resource availability. OpenStack was widely used as virtualization platform to host different VNFs such as fundamental network services like DNS, DHCP, VPN, etc., and the network services related to the use cases. OpenDayLight was used as the SDN controller in this network to offer compatibility with the switches and network resources in the Testbed. Also, OSM was implemented as a domain orchestrator for this project to deploy different VNFs within the network.

The Time-Shared Optical Network (TSON) was used as a dynamic optical transport network solution to provide high bandwidth and low latency connectivity between the network edges and the datacenter. For this solution an FPGA board from Xilinx (one of the consortium partners was used to demonstrate the programmability of the TSON solution).

For the Stadium Demonstration, the project used x86 servers because they were suitable for the compute requirements of the VNFs used during the demo and the different controllers.

Pishahang is an NFV MANO that allows management and orchestration of NFV services across multiple VIM domains. A single service in Pishahang can contain VNFs that can run on AWS, Open Stack and Kubernetes. This allows to use heterogeneous resources offered by different VIMs for the same services. Unlike other MANO frameworks that run Kubernetes on a VM, the Kubernetes VIM of Pishahang runs on bar metal. This removes one layer of virtualization and improves the performance of the containers. Kubernetes was used because of two main reason namely 1) it allows managing container-based VNFs, which have better performance compared to VM-based VNFs, and 2) it also allows faster management and orchestration of NFV services compared to other solutions such as OpenStack.

### 5.4.2 Phase 3 projects: infrastructure

Concerning Phase 3 projects, **5G-EVE** strongly advocates for COTS hardware for economic reasons. For the moment, the use cases supported at 5G-EVE do not require any kind of hardware acceleration, but that may change due to the ongoing introduction of gaming use cases or new use cases with an intensive use of ML (e.g., Computer Vision at the Edge use cases). Regarding the orchestration, 5G-EVE supports vendor-provided orchestrations, OSM and ONAP, depending on the site facility where the use case is to be hosted and executed and the partners involved. The use of OpenStack is extended in clouds dedicated to Vertical Applications and Kubernetes is used for the 5G Core deployments.

In **5G-VINNI**, on the contrary, no restriction is made on the technologies employed for implementation.

Finally, the Edge Computing infrastructure deployed in the **5GENESIS** Málaga Platform comprises:

1) <u>COTS OCP X86 Servers</u>: X86 servers from OCP (vendor agnostic) to run Edge datacenter management and provide compute resources to Edge VNFs. X86 is the most adopted compute resource and supported by most Open Source components used in CORD.
2) <u>Openflow Whiteboxes Switches</u>: Servers are connected to Openflow Whiteboxes Switches so that connectivity can be programmed and managed from an SDN

controller. All connectivity inside Edge Datacenter and from the Datacenter to the RAN and transport Network is managed by an SDN controller.

3) ONOS SDN Controller: This is the out of the box SDN controller developed by Open Networking Foundation, that is included in the CORD solution. It supports Openflow and now P4 to manage the Switching fabric of the Datacenter.
4) OpenNebula Datacenter VIM: Open Nebula is a lightweight VIM to manage hardware resources (compute, storage, networking and other PCI devices). It can use different hypervisors, but 5Genesis has selected KVM as it is one of the most adopted ones being part of Open source solutions, like OpenStack. Compared to this one, Open Nebula consumes much less resources, a critical feature for small Datacenter, designed for Edge capabilities.

Container solutions can be deployed inside the Edge Datacenter running on VMs, enabling the deployment of containerized VNFs and Applications if needed.

In the Athens 5GENESIS platform, COSMOTE is the provider of the Edge Computing infrastructure. COSMOTE operates a hybrid 4G/ NSA 5G/ MEC testbed complemented with an Openstack-based SDN/NFV Cloud infrastructure. More specifically, the COSMOTE 4G/5G testbed is composed of:

- A lightweight 4G/5G EPC/IMS core network (running on 2 VMs on a Dell R630 server)
- Two flavors of MEC implementation
  - Via second SPGW
  - Via SGW-LBO (Local BreakOut)
- Nokia Airscale 4G/ 5G BTSs for providing 5G radio connectivity
- Eight Nokia 4G/WiFi Flexi-Zone Multiband Indoor Pico BTS, supporting standard network interfaces (such as S1 and X2), 5/10/15/20 MHz LTE carriers with 2x2 MIMO, along with Wi-Fi connectivity @2.4 and 5GHz delivering thus a HetNet solution

The testbed includes additionally:

- An Openstack-based multi-cloud infrastructure. The testbed collectively consists of >720 CPU cores, >1700GB RAM and >120TB storage space, and is interconnected (mostly) via 10Gbps fiber/copper links.
- A MANO installation (ETSI OSM – based), offering NFV capabilities on the multi-cloud infrastructure.
- A flexible, scalable, E2E IoT platform – developed from scratch exclusively by COSMOTE- including:
  - A wide range of custom and commercial one end-devices/sensors such as, air-quality, temperature, humidity, pressure, activity, luminance, smoke/fire, activity as well as power/energy-related ones (relays, power meters, etc.), communicate with a backend (cloud) infrastructure over a wide range of short/long range technologies (Ethernet, WiFi, z-wave, BLE, LoRaWAN, 3G/4G and NB-IoT).
  - IoT hubs/gateways for facility automation and energy management/control (based on events/rules) supporting multiple HAN/BAN/LAN/WAN technologies/interfaces; over 150 technologies/protocols are currently supported.

  - A (common) backend infrastructure (incl., storage, monitoring/data visualization, command exchange, etc.).

10Gb/s broadband connection (over GRNET) serves as a backhaul link towards the internet and the NCSR Demokritos premises, where the ATHONET EPC and/or 5G Core is operated.

### 5.4.3 Phase 3 projects: automotive

The **5G-CARMEN**[152] implementation for the MEC considers two different approaches:

- DTAG uses a plain KVM based "Cloud" infrastructure as this is the actual Nokia implementation of a MEC.
- TIM uses OpenStack based Cloud infrastructure with 3 Nokia servers (the system is a cloud infrastructure only system without ETSI MEC specific Extensions)

In **5G-MOBIX**, all sites use COTS hardware based on X86 Servers (which is also the case in **5G-CROCO**) and NFVI using OpenStack. In some cases, OPNFV (a source carrier grade NFV reference platform hosted by Linux Foundation) is deployed. Also, Kubernetes is used on one site, providing a container orchestration solution.

### 5.4.4 Phase 3 projects: advanced trials across multiple vertical industries

In **5G-HEART**, there was no specific reason for relying on any specific technology from the processor architecture view. Most of the processors in the 5GTN site are based on Intel manufacture including the current Edge Computing architecture with the open-source CDN setup. Docker platform as a service was selected as a container for packing the necessary video compression software for easier displacement between different Edge nodes and allowing faster setup, upgrade, and dynamic migration of the Edge components into operational. The current setup is based purely on software acceleration (optimization), but GPU acceleration will be considered in future concerning especially the video transcoding part.

Overall, the choice of hardware is mainly due to the ease of procurement. The choice of orchestration solutions and virtual infrastructure management is due to their simplicity and large community support.

Some hardware acceleration is needed, specifically for the video analytics in the aquaculture use case.

As **5G!drones** relies on four different trial facilities, many of the technologies listed in Table 4 are used for the 5G!Drones Edge Computing deployment. Specifically,

- 5G-EVE relies on X86 Servers, a home-made MEC orchestrator as well as Kubernetes for managing edge resources;
- 5GENESIS provides in the Athens platform two types of Edge Computing infrastructures that are deployed on small form factor (SFF) x86 Servers: (i) OpenStack and (ii) Kubernetes. Katana Slice Manager is another open source

[152] https://5gcarmen.eu

software component that is closely aligned with the 3GPP and GSMA standards regarding the network slicing. It was designed to support the activities of the 5GENESIS platforms, supporting the management of E2E network slices on top of the platform facilities.

- 5GTN uses also X86 Servers in addition to COTS devices as MEC hardware while OpenStack and Kubernetes represent the VIMs. Open Source MANO is used as an orchestration tool. X-Network facility deploys X86 servers as Fog servers, while Nokia MEC comes as COTS solution. LXD is used for the management of Edge services using Linux Containers LXC, this technology was adopted because it allows live service migration between edge servers

As discussed earlier, **5GROWTH** leverages on the 5G-TRANSFORMER stack that allows integrating any kind of technology out of those listed in Table 4 as long as the corresponding plug-in is developed. Some of the marked technologies are those for which a plug-in/wrapper was already developed in 5G-TRANSFORMER, and some (e.g., GPU Acceleration) will be developed in 5GROWTH. In general, X86 servers were selected because of availability and familiarity. COTS devices were used as interfaces in the nodes, UEs in the tests, etc. The 5GROWTH stack has integrated two MANO platforms, namely OSM and Cloudify. Likewise, the 5GROWTH Resource Layer can integrate a plethora of heterogeneous transport and computing technologies through the corresponding plug-ins: ONOS, ODL, Ryu, and ABNO were the ones integrated due to functionality offered and their availability in the labs of partners that were familiar with them. This is also based on past developments of use from previous projects (e.g., 5G-Crosshaul). OpenStack and Kubernetes will also be further evaluated within the project to explore different options to deploy VNFs.

Each of the four facilities in the **5G-VICTORI** project features on premise, private, and micro datacenters. These are built using primarily COTS x86 servers, some of which have GPU acceleration and switches (some are SDN enabled). In addition, smaller form factor devices, such as Intel NUC, which also in some instances include GPUs, are deployed in the field (e.g. street cabinets). Most of the tools comprising the protocol stack are open source. All the facilities utilize OSM for network management. In addition to that, Orange Romania at the Alba Iulia site will also investigate the integration of ONAP. Last, in Patras, OpenSlice[153] is being exploited for service orchestration, as a tool that was developed by the University of Patras previously. In terms of SDN controller, both ONOS and ODL are deployed. In addition, the Bristol site is also equipped with the Zeetta NetOS[154], a network control and management software platform that simplifies and automates Network Operations (NetOps). This is used with their Rapide box. The primary VIM platform is OpenStack, providing support for VMs, which is the preferred solution for vertical application deployment. However, Kubernetes is also deployed in some Edge environments, because of its low resource footprint requirements. In addition, some of the underlying tools used, such as OSM and OpenSlice, are by themselves deployed in the form of containers.

### 5.4.5 Phase 3 projects: 5G Long Term Evolution

Since **MonB5G** recently started, the discussion around specific server architecture to be used in its architecture deployment has not reach the level of detail yet. In general,

[153] http://openslice.io.

[154] https://zeetta.com/netos-architecture.

however, x86 servers are likely to be adopted in future PoC deployments due to their popularity and relatively low costs. Both OSM and ONAP platforms are currently considered as MANO platforms. OSM follows the “de facto standard” of ETSI NFV MANO architecture, and ONAP is commonly considered as a future solution for automation of technical processes. Despite providing a valuable starting point from an architectural point of view however, none of them fully adheres with the scalable data-driven network slice management and orchestration architecture envisioned by the MonB5G project.

Again, the discussion around SDN did not reach a consensus on the specific platform to be exploited. At the time of writing, both ONOS and OpenDayLight controllers present some limitation in terms of scalability due to code size and documentation. Most likely the project will adopt lighter solutions.

The network slicing scenario considered in the MonB5G project implies the mobile network infrastructure to be highly flexible and dynamically reconfigurable. To exploit the full potential of NFV technologies and support the development of its distributed architecture, the MonB5G project will exploit both OpenStack and Kubernetes open-source platforms. On the one side, Kubernetes is the most widely used container orchestration tool and allows for fast automation and configuration of both networking and vertical services. When compared with VNF-based deployments, containers can usually provide faster setup and easier portability thanks to their lightweight. This might be especially useful in case of migration and/or service reconfiguration. On the other side, single VNFs instances hosted on VMs have been proved to be more secure thanks to the complete isolation from OS Kernel provided by the virtualization hypervisor. Thus, the project envisions a co-existence of these technologies to fulfil the flexibility and resilience requirements imposed by the network slicing scenario.

The Edge Computing deployment in the 5GBarcelona testbed of **5GZORRO** is implemented using x86 servers, which are virtualized to form OpenStack clouds that are managed and orchestrated by OSM. The aforementioned technologies were selected considering their widely extended use and strong community support. Another element supporting this choice is the availability of two of them (i.e., OpenStack and OSM) as open-source projects, where partners involved in the 5GZORRO consortium have an active participation. Moreover, specific contributions to the OSM open source framework might be put into place, if required to realize the project’s objectives related to the design of a security and trust framework, integrated with 5G service management platforms. The introduction of other computing technologies based on ARM processors in APU devices, SDN control based on OpenDaylight and support of container orchestration through Kubernetes is on the roadmap.

### 5.4.6 EU-Taiwan Cooperation

In **5G-DIVE**, the 5TONIC Lab (used for prior-trial validation) and OPTUNS Edge Datacenters (used for the in-site trial validation) rely on both AMD and Intel Servers. NVIDIA Jetson and CORAL TPU boards are also being considered as Fog CDs for local processing and acceleration of AI/ML tasks. Finally, the 5G-CORAL framework, which is leveraged in this project, adopted Eclipse fog05 implementation for the VIM and Orchestrator. Eclipse fog05 provides a decentralized infrastructure for provisioning and managing compute, storage, communication and I/O resources available anywhere across the network.

### 5.4.7 Analysis of results

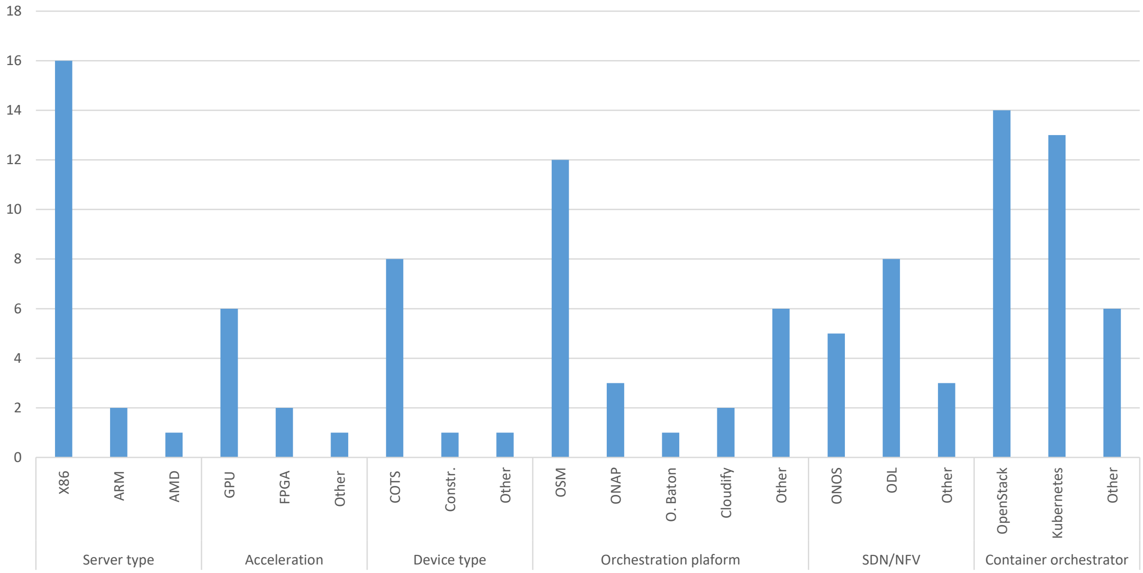


**Figure 29: Main technologies adopted in Edge computing deployments.**

Figure 29 reveals some clear trends. In terms of architecture, there is a clear preference for the x86 architecture, with just a couple of projects looking at other architectures, i.e., ARM. Regarding acceleration, it seems most of the projects are not using acceleration at all, and the few ones using it focus on GPU acceleration. This trend probably originates in projects studying AR/VR scenarios and requiring of GPU acceleration for the rendering of images. Regarding orchestration platforms, OSM clearly dominates. The high number of ‘Other’ answers in the Orchestration part evidences that multiple projects are developing their own solutions in contrast to using well-known platforms.

In terms of SDN controller platform, ODL is used as preferred platform, although ONOS is also widely used. In the category of ‘Other’ we can find mostly deployments using Ryu as SDN controller, or specific developments.

Finally, in terms of VIM (Container Orchestrator), OpenStack and Kubernetes are mostly used equally, followed by project-specific developments.

## 5.5 Applications / VNFs deployed at the Edge

Finally, we close our analysis by investigating the applications and/or VNF that have been considered by the various 5G PPP projects in their 5G Edge network deployments. Table 5 contains in a summarized form the information collected by the reporting projects.

**Table 5: Applications and VNFs deployed at Edge networks by each project.**

| | | Cloud RAN | vEPC | 5G Core | SDN Contr. | Vertical App. | End-user app. | Other |
|---|---|---|---|---|---|---|---|---|
| Phase 2 | 5G Transformer | | x | | | x | | |
| | Slicenet | x | | | | x | | |
| | SaT5G | | x | x | | | | x |
| | 5G Picture | | | | | x | | x |
| Phase 3 Infra | 5GEve | | x | x | | x | | |
| | 5GVinni | | | | | | | x |
| | 5Genesis | | x | | x | x | | |
| Phase 3 Automotive | 5GCroCo | | x | | | x | | |
| | 5GCarmen | | | | | x | x | |
| | 5GMobix | | x | x | x | x | | |
| Phase 3 Adv. Trials | 5G Heart | | x | x | | x | x | |
| | 5G Drones | | x | x | | x | x | |
| | 5G Growth | | x | x | | x | | |
| | 5G Victori | x | x | x | x | x | x | x |
| Phase 3 Long-term evol. | MonB5G | | x | x | | x | | x |
| | 5GZORRO | | x | | | | | x |
| EU-Taiwan | 5G-Dive | x | x | | | x | | |

## 5.5.1 Phase 2 projects

In **5G-TRANSFORMER**, two types of VNFs are mainly deployed, namely those related with the low-latency components of vertical applications (e.g., the algorithm for avoiding collisions, emergency server) and those components of the mobile network that are needed for the local breakout to have access to these applications directly at the Edge without having to reach the core network of the operator (e.g., virtual PGW in the eHealth use case).

For Cloud RAN, it is natural to deploy the CU/BBU at the edge. For vertical applications, firstly in the eHealth use case, **SliceNet** deploys an App called TeleStroke to benefit from the low latency in the MEC platform in order to support the timely diagnosis of onboard patients who may suffer from stroke. Secondly, in the Smart Lighting use case, SliceNet deploys an IoT Gateway MEC App to enhance the timely processing capabilities of the Gateway at the Edge of the network and also improve the scalability of the gateway in supporting mMTC.

**SaT5G,** considers a use case on DASH Live Streaming over Satellite Backhaul. Specifically, the project uses satellite communications as backhaul in a 5G network to support 4K video streaming applications with QoE assurance. SaT5G focuses on HTTP-based live streaming scenario, where video content is generated on-the-fly at a content origin server and delivered to geographically distributed end-users through a 5G network with satellite backhaul. Specifically, it presents a 5G SBA-based framework that provides QoE assurance in a context-aware manner. The project envisages which stakeholders are involved in this scenario, i.e., 5G MNO, video content provider (CP) and satellite network operator (SNO). In the proposed framework, the 5G MNO virtualizes its computing and storage resources and leases them to CPs, where the latter can deploy their own VNFs in MEC servers. Meanwhile, the SNO leases its satellite channel bandwidth resource to the 5G MNO, so that the latter uses it as a backhaul link in addition to the standard terrestrial backhaul. The key contributions are as follows:

- This is the first system developed in literature that utilizes both SBA-based 5G core network and satellite backhaul to support 4K HTTP-based live streaming applications with QoE assurance. Specifically, it leverages both the context awareness and flexibility that are enabled by 5G SBA architecture, as well as the multicast capability of satellite backhaul. It also utilizes virtualization technology to enable CPs to deploy their own VNFs in MEC servers at 5G mobile Edge, which not only performs content operations such as transient segment holding, but also realizes last-hop multicast at application layer. Overall, the proposed system assures live users' QoE while maintaining the video quality at or above 4K; it also ensures that video content is always delivered through the backhaul in the most efficient manner.

- This is the first time that a 5G core network and a real satellite communications link have been implemented and integrated as a holistic system, where the latter serves as the backhaul of the 5G network. The establishment of such a system means that it is possible to test the performance of MEC servers with content operations (such as transient segment holding) in terms of content delivery and QoE assurance through a real satellite backhaul.

In **5G-PICTURE's** stadium demo, a container-based VNF is deployed. The VNF identifies when a handover from a low-priority to a high-priority slice should take place. The VNF follows the microservice-based architecture and consists of four containers, namely massage broker, database, packet sniffer, and endpoints. This VNF identifies the need for hand over by sniffing the network traffic. It, then, stores some metadata in the database. Using the Endpoints, it allows the metadata to be accessed by other management entities. The project deployed the VNF at the Edge to react faster to changes in the network traffic which, consequently, improves the performance of the adoptive slicing provided in the demo. Having the VNF at the Edge also allows to save bandwidth as a large amount of traffic are sent to the VNF to be processed.

### 5.5.2 Phase 3 projects: infrastructure

**5G-VINNI** work on Edge deployment is intended to support 3rd party experimentation and as such, the VNFs and/or applications to be deployed at the Edge are set by the experiment design process. 5G-VINNI Edge implementations are therefore intended to be flexible enough to incorporate a range of different Edge/Core(Cloud) splits.

As already mentioned, the 5G-VINNI Berlin and Luxembourg Experimentation Facility Sites implement an Edge-Central 5G Core Network functionality split. Thus, the VNFs deployed at the Edge correspond to the relevant 5GC functions corresponding to the associated slice model. In particular, four slice models have already been implemented in the 5G-VINNI Berlin and Luxembourg Experimentation Facility Sites support:

- Centralized (Direct Connectivity): all the network functions are placed at the central location. This solution relies on the advantage of the optimized satellite direct connectivity (and on the foreseen integration of the satellite specific protocols with the 5G ones) to establish a large scale gNB at the hub side. In this extreme case, no 5GC VNF deployed at the Edge.
- Local Offload with centralized control plane: the Edge node is able to offload the data traffic to the Edge; however, the control is done from the central location. As the control remain centralized, in this situation a larger E2E procedure delay for establishing the offloading is required. In this case, the 5GC VNFs deployed at the Edge correspond to AF and UPF.
- Proxy Node: it offers a transparent connectivity service to the UEs and at the same time it "self-backhauls" across the same technology. In this case, the 5GC VNFs deployed at the Edge correspond to AF and UPF.

- Autonomous Edge Node: placing the comprehensive control plane elements at the Edge including an additional front-end for device management and for user data subscription, using information stored in the local cache and default subscription profiles will enable the system to act as an autonomous connectivity island which makes decisions on its own functioning. In this case the Edge side can function in a complete manner when the backhaul connectivity is lost. However, with passing the subscription profiles to the Edge node, an increase security of these nodes has to be established. This solution should be considered only when the trust in these edge nodes is large. In this extreme case, the 5GC VNFs deployed at the edge correspond to AF, UPF, AMF, SMF, PCF, and potentially also DM and UDM.

The main deployed service in **5GENESIS** is the 3GPP-compliant mission-critical services (MCS). The Nemergent MCS server-side provides the application-level components required to deploy MCPTT services: Mission Critical Video (MCVideo) and Mission Critical Data (MCData) services. The Nemergent MCS system is deployed as a series of server components, each of them fulfilling a different functional role. Among the required standardized components, the project offers MCS Application Server (both Participating and Controlling roles), MCS Management Servers (all Identity, Configuration, Key and Group Management Servers), IMS Core (with a SIP-based load-balancer), and networking-based management modules such as DNS, NAT transversal and so forth. All the above-mentioned components are VDUs that constitute an all-in-one MCSVNF. The main reason to select this VNF for the Edge infrastructure has been the ability and the great potential of this network paradigm. For instance, being able to handle crowded events and utilization of MCS communications that are sensitive to latencies, while at the same time being able to support a large number of resources like MCVideo communications. Additionally, in order to enable LBO of user´s traffic to MCS VNFs running at Edge Computing node, 5GENESIS has deployed vEPC VNFs. Since there is a need to steer traffic coming from RAN (S1interface) towards MCS VNFs, it is necessary to terminate GTP tunnels at the Edge node, and by doing so, the project has deployed S/P-GW function in a separate VNF and configured the Mobile Core accordingly. Lastly, the Edge Computing node runs additional VNFs for infrastructure management. Two mail VNFs are used for this management: ONOS is the SDN controller to manage connectivity inside the datacenter and derives from the CORD design. The project adopts a canonical ONOS SW release by the Open Networking Foundation. Open Nebula is the VIM selected to manage resources, and it runs also in VMs as VNFs. This VIM includes the interfaces to external management system like OSM that orchestrate the deployment of VNFs from VNF catalogue in the Edge Computing node.

In the Athens 5GENESIS platform, the deployed services support the various requirements of UAVs applications, namely the FCC virtualised units. The UAVs have been brought into the scope of the latest 3GPP releases, in order to study and address the related needs and requirements (e.g. TS22.125, TS22.261, TR 36.777, TS 22.125).

### 5.5.3 Phase 3 projects: automotive

In **5G-Carmen**, the main applications/VNFs running on the MEC are (i) geo-service and AMQP Broker; and (ii) cooperative lane merge application. Green Driving, on the contrary, is not deployed at the edge. In **5G-CroCo,** vEPC is deployed at Edge to enable local breakout towards MEC hosts. Finally, in **5G-Mobix,** both vertical applications and core functions are deployed at the Edge as well. For the EPC, a PGW-U is deployed and for the 5G Core a UPF is deployed. An SDN controller will make virtual networking functions possible which will be integrated with the SMF in case of the 5G core deployments. Depending on the site, different vertical applications are hosted. The most common application will be an application facilitating the exchange of V2X messages between vehicles and between infrastructure and vehicles. Also, several post processing applications are deployed, responsible for data fusion and vehicle controlling.

### 5.5.4 Phase 3 projects: advanced trials across multiple vertical industries

In **5G-HEART**, the referred vertical application to be run in the Edge is basically a video service that contains video transcoder, video storage and the video delivery service running on top of HTTP. A similar justification applies as previously stated; it is wiser to move the video processing and caching closer to users rather than to perform it in low-processing units, e.g., streaming cameras. One of the end user applications can be also run in the Edge, which is basically a HTTP-based video player (client) software in a browser. The project plans to use this software as an optional for easier measurement handling. NSA and SA 5GC will be deployed in Edge for eHealth: either UPF to support low latency applications or full 5GC to support NPN or Autonomous Edge.

As discussed earlier, the **5G!Drones** project focuses on trialling UAV scenarios on top of existing 5G facilities. UAV relies on flying drones which needs to be controlled and commanded via a remote application, where low latency communication is critical. Clearly, the remote control/command application needs to sit at the Edge, aiming at guaranteeing low-latency connection to the flying drones deployed on top of the 5G network. In addition to controlling the flying platforms, 5G!Drones also investigates use cases where the UAVs embed various services and applications such as video monitoring, 3D mapping, etc., which also require Edge Computing capabilities. The ETSI MEC deployment will bring many benefits to these use cases since they are latency-sensitive and require RNIS, Location API, video processing at the Edge. It will further improve the scalability and allows the sensor and components involved in this use cases to maintain a consistent and reliable connection.

There are mainly two types of VNFs deployed in **5GROWTH** , namely (i) those related with the low-latency components of vertical applications (e.g., virtual M3Box composed of several control applications for controlling AGV and the CMM); and (ii) those components of the mobile network that are needed to be able to do the LBO to have access to these applications directly at the Edge without having to reach the core network of the operator (e.g., vEPC, UPF).

There are three primary VNF group types deployed at the edge in **5G VICTORI**. First and foremost are the vertical/end-user applications as described with the three use cases in Table 1, as well as others being trialled in the project. 5G VICTORI will trial a number of use cases and most of them are deploying some component at the Edge in the form of VM and in some cases, as containers. The rationale of deploying these applications at the Edge is to meet the latency requirements and/or provide bandwidth efficiency. There are some secondary benefits of that, such as security, whereby data is not allowed to leave the premises of a facility, but these are not prevalent in the scenarios we evaluate. The second group is the SDN controller and the MANO system (OSM) that controls this Edge instance itself. It is possible to have a single MANO responsible for both the core and Edge cloud, but we have opted for a hierarchical architecture, where each edge is autonomous and a common platform (5G-VIOS) is providing the inter-domain (inter-edge) orchestration. Finally, in Bristol and in Patras where pop-up network-in-a-box solutions are utilized, a complete vEPC and/or 5GCore are deployed at the edge. In addition, there are scenarios where an edge also acts as a potential centralized unit (CU) of the disaggregated 5G-NR cell.

### 5.5.5 Phase 3 projects: 5G Long Term Evolution

The state-of-the-art solutions in the network slicing context often imply vertical service deployments at the Edge of the network in order to satisfy latency requirements and ease the virtual/physical function chaining necessary to build the E2E service. However, in the scenario considered by the **MonB5G** project, the management/control plane of the mobile network will need to handle the deployment of a massive number of network slices. This calls for the definition of a highly distributed management plane capable of providing self-management and self-configuration capabilities. To this aim, the MonB5G will evolve the traditional centralized cloud management system architecture, composed of three main entities: Monitoring System (MS), Analytics Engine (AE) and Decision Engine (DE), towards a distributed system, where the components will be decomposed in hierarchical levels and distributed over different technological domains (i.e., RAN, Core, Cloud, and MEC). Thus, the Edge platform will host not only the slice-specific application, but also a combination of MS/AE/DE functionalities to support and enhance the service provisioning from an E2E perspective. At the same time, specific use-cases might require some vEPC/5GCore functionalities to be deployed at the Edge, e.g., to reduce the gap between the data plane anchor point (for example, the end point of a PDN connection in LTE terminology, or the UPF in 5G terminology) and the vertical application itself, especially in case of low-latency scenarios.

Any services required by the RAN components of the infrastructure have to be placed as close as possible to the radio equipment. In **5GZORRO** these components will be the virtualized layer 3 component of the LTE stack (vL3) and vEPCs for the different operators that intend to deploy services in the network. By placing these elements at the Edge, the KPIs of small latency and round-trip times can be met, which would have not been the case if they were placed in the core/main datacenter. It will have to be evaluated during the project whether other VNFs or services have similar requirements and whether they will be deployed at the Edge.

### 5.5.6 EU-Taiwan Cooperation

Due to the very low latency and/or local processing requirements of the use cases target within the **5G-DIVE** project, vertical applications and services are required to be deployed at the Edge in order to contribute to the fulfilment of these requirements. For the Autonomous Drone Scouting use cases, it is being considered the virtualization of the SGW.

### 5.5.7 Analysis of results

Figure 30 presents the analysis of the survey regarding the kind of application deployed at the Edge. Currently, the Edge is mostly used to store vertical specific applications, such as VR/AR renderer, CDNs or robotic control. All these applications are not related to the network itself, but with the actual vertical industry in charge of the use case. Secondly, the Edge is used to store vEPCs, followed by 5G Core functions. Since 5G SA is yet not deployed widely, research projects are currently using 5G NSA, requiring therefore of a 4G core. This is the reason behind the increased hosting of vEPC instances in the Edge compared with 5G Core functions. Within the 5G Core functions, UPF is the most commonly deployed one.

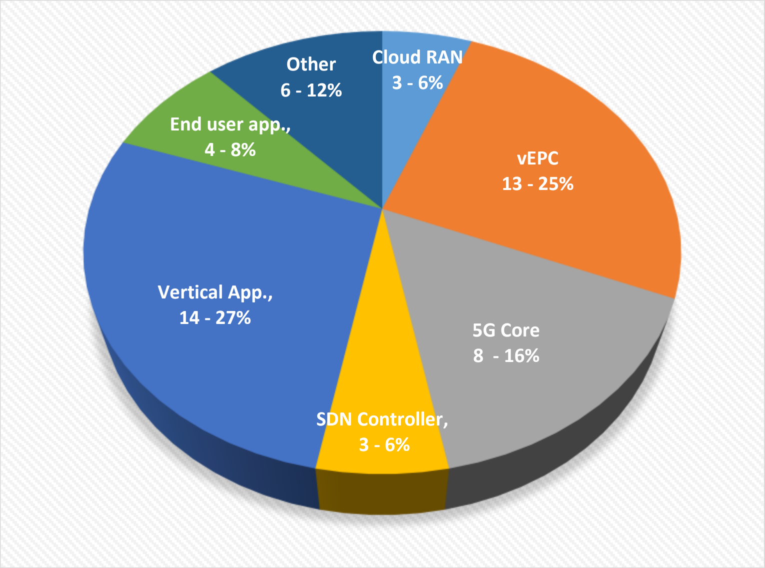


**Figure 30: Type of Applications/VNFs deployed at Edge networks.**

From this figure, we can see that Edge Computing infrastructure serves clearly two purposes:

1. Run infrastructure components of the 5G network such as Cloud RAN, SDN and Core elements (above 50%) to shorten the distance to …
2. Apps running at the edge (near to 40%), collocated with Core components for getting Local Break Out access to users traffic.

This combination of Core components + Apps enable delivering the 5G value proposition for URLLC, eMBB and mMTC.

# 6. Conclusions

Computers and Networks, Networks and Computers, are two technologies that have been evolving hand by hand in the last 50 years, since the introduction of Internet. [155]

We are now facing the next Mobile Network technology generation for 5G and, even though we heard about Edge Computing since the introduction of CDNs, in the 2000s, it is 5G the one that is driving the development of Edge Computing.

In this paper, we have seen that 5G PPP projects researching on very different 5G use cases and with a variety of companies and technologies, they all have embraced Edge Computing as part of their solution. Seventeen out of seventeen projects reported using some type of Edge Computing at the Edge of the network (Table 2).

We have also explored that the concept of Edge of the Network can be flexible, and depending on the context, 5G PPP projects have been using different locations for the Edge resources (Table 3), from pure On Premise infrastructure to the Public Cloud.

And we have confirmed that this infrastructure is used to implement the LBO function to shorten the path between Users and Applications. As shown in Table 5, vEPC and 5G Core functions are co-located with Vertical Apps at the Edge.

This approach is the way to go to deliver the 5G value proposition: Ultra reliable low latency communications, enhanced mobile broadband (higher bandwidth) and massive Machine Type Communications.

But the picture in not completely clear. The Edge Computing ecosystem needs to mature so that companies can get access to commercial solutions with a clear value chain. It is not clear if this ecosystem will be dominated by Telcos, Hyperscalers or there will be some kind of *coopetition* (collaborative competition). Telcos have the capillarity and dominate Locality, while Hyperscalers have the Cloud Technologies and are typically Global.

There are also uncertainties about security, privacy and regulation, to name a few, that need to be addressed before the market matures.

Security issues have been raised by Interpol regarding "Law enforcement and judicial aspects related to 5G" [156] on topics such as Lawful Interception and Authenticity of the evidence, in virtualized collaborative environments.

Privacy issues are also related to these types of environments, as guidance is needed on what data can be shared between network elements and applications running in the same infrastructure.

[155] https://en.wikipedia.org/wiki/History_of_the_Internet#Merging_the_networks_and_creating_the_Internet_(1973–95)

[156] https://www.statewatch.org/media/documents/news/2019/jun/eu-council-ctc-5g-law-enforcement-8983-19.pdf

Lastly, Telcos and Hyperscale are very different regulated. In offering services on 5G, new regulation is needed that harmonizes the roles for all the actors involved in the value chain and enable the development of a healthy Edge Ecosystem.

5G is ready to take off, now is time for Edge Computing to step up and mature to become the perfect partner.

# ANNEX 1: List of relevant project deliverables

This Annex presents a list of deliverables which may be interesting to the reader of Section 5, to deep dive in the technologies used by each research project.

**5G-VICTORI (https://www.5g-victori-project.eu)**

- D2.1 - 5G VICTORI Use case and requirements definition and reference architecture for vertical services.
- D2.2 - Preliminary individual site facility planning.

**5G-PICTURE (https://www.5g-picture-project.eu)**

- D4.1 State of the art and initial function design. [Section 4].
- D4.2 Complete design and initial evaluation of developed functions. [Section 3.2]
- D5.3 Support for multi-version services. [Section 3]
- D6.3 Final Demo and Testbed experimentation results. [Section 8]

**5G- VINNI (https://www.5g-vinni.eu)**

- D1.1 – Design of infrastructure architecture and subsystems. [Section 6.1]
- D2.1 – 5G-VINNI Solution Facility-sites High Level Design (HLD). [Section 4.5]

**5G-HEART (https://5gheart.org)**

- D2.1: Use Case Description and Scenario Analysis. [Chapter 4]
- D2.2: User Requirements Specification, Network KPIs Definition and Analysis. [Chapter 3]
- D3.2 Initial Solution and Verification of Healthcare Use Case Trials. [Chapters 2-6]
- D4.2 Initial Solution and Verification of Transport Use Case Trials.
- D5.2 Initial Solution and Verification of Aquaculture Use Case Trials

**5G-CROCO (https://5gcroco.eu)**

- D2.1 Test Case Definition and Test Site Description Part 1 [Section 3]
- D4.4 Detailed Roadmap of Test Sites- Project Year Two [Section 6; Sections 10-16]

**5G-MOBIX (https://www.5g-mobix.com)**

- D2.2 5G architecture and technologies for CCAM specifications. [Sections 2.2, 3.4.2, 4,5]
- D2.3 Specification of roadside and cloud infrastructure and applications to support CCAM. [Section 3]
- D3.1” Corridor and Trial Sites Rollout Plan” [Section 2.2]

**5GROWTH (http://5growth.eu)**

- D1.1 Business Model Design
- D2.1 Initial design of 5G End-to-End Service Platform
- D3.1 ICT-17 facilities Gap analysis

**5G-TRANSFORMER (http://5g-transformer.eu)**

- D4.3 Final design and implementation report on service orchestration, federation and monitoring platform [Section 6.7.3]
- D2.3 Final design and implementation report on the MTP (report) [Sections 5.3.1.4, 5.3.4.2., 5.5.3, 5.6.1.2]
- D1.4 5G-TRANSFORMER final system design and Techno-Economic Analysis [Section 3.3]

**SLICENET (https://slicenet.eu)**

- D2.2 Overall Architecture and Interfaces Definition [Section 6].
- D2.3 Control Plane System Definition, APIs and Interfaces [Section 5]
- D3.1 Design and Prototyping of SliceNet Virtualised Mobile Edge Computing Infrastructure

**SaT5G (https://www.sat5g-project.eu)**

- D2.3 Business Modelling and Techno-economic Analysis of Satellite eMBB
- D4.6 Caching and Multicast –Analysis, Design and Proof of Concepts
- D5.3 Demonstration of Fixed and Home Backhaul Scenarios Including Caching & Multicast
- D5.4 Demonstration of Mobile Backhaul Scenario Including Caching & Multicast

# Abbreviations and acronyms

| | |
|---|---|
| 3GPP | 3rd Generation Project Partnership |
| 5G NR | 5G New Radio |
| 5G NSA | 5G Non Stand Alone |
| 5G SA | 5G Stand Alone |
| 5G PPP | The 5G Infrastructure Public Private Partnership |
| ABAC | Attribute-Based Access Control |
| ACL | Access Control Lists |
| ADCs | Application Delivery Controllers |
| AECC | Automotive Edge Computing Consortium |
| AI | Artificial Intelligence |
| ASICs | Application-Specific Integrated Circuit |
| AWS | Amazon Web Services |
| BBF | Broadband Forum |
| CD | Continuous Deployment |
| CDNs | Content Deliver Networks |
| CI | Continuous Integration |
| CLI | Command Line Interface |
| CM | Continuous Monitoring |
| CNCF | Cloud Native Computing Foundation |
| CNFs | Container Network Functions |
| CORD | Central Office ReArchitected as Datacenter |
| COTS | Commercial of the self |
| CPUs | Central Processing Units |
| CSPs | Communication Service Providers |
| CUPS | Control – User Plane Separation |
| DevOps | Development and Operations |
| DMA | Direct Access Memory |
| DoSD | Denial of Service Detection |
| DPDK | Data Plane Development Kit |
| eMBB | Enhanced Mobile BroadBand |
| FaaS | Function as a Service |
| FCAPS | Fault, Configuration, Accounting, Performance, Security (ISO Telecommunications Management Network model and framework for network management) |
| FOG | Refers to Fog Computing, Fog Networking |
| FPGAs | Field Programmable Gate Array |
| GPUs | Graphic Processing Unit |
| Hyperscaler | Refers typically to Amazon Web Services, Microsoft Azure and Google Anthos |
| ITU-R | ITU Radiocommunication |
| KNI | Kubernetes Native Infrastructure |
| LBO | Local Break Out |

| | |
|---|---|
| LNaaS | Logical Network as a Service |
| MANO | Management and Orchestration |
| MCPTT | Mission Critical Push To Talk |
| MCS | Mission Critical Services |
| MEC | Multi-access Edge Computing |
| MitM | Man-in-the-Middle |
| mMTC | Massive Machine Type Communications |
| mmWave | Millimetre Wave (Spectrum from 30Ghz to 300GHz) |
| MNOs | Mobile Network Operators |
| NFV | Network Function Virtualization |
| NFVI | Network Function Virtualization Infrastructure |
| NOS | Network Operating System |
| NPUs | Network Processing Units |
| NSaaS | Network Security as a Service |
| NSI | Network Slice Instance |
| NSSAI | Network Slice Selection Assistance Information |
| NST | Network Slide Template |
| O-RAN | Open RAN |
| OCP | Open Compute Project |
| OFDM | Orthogonal Frequency-Division Multiplexing |
| ONAP | Open Network Automation Platform |
| OPNFV | Open Platform for NFV under Linux Foundation |
| OSM | Open Source Mano under ETSI NFV ISG |
| OTEC | Open Telco Edge Cloud |
| OTT | Over The Top |
| P4 | Named for Programming Protocol-independent Packet Processors |
| PaaS | Platform as a Service |
| PCIe | PCI Express |
| PLMN | Public Land Mobile Network |
| PNFs | Physical Network Functions |
| POPs | Points of Presence |
| RBAC | Role-Based Access Control |
| RIC | Radio Intelligent Controller in Open RAN |
| RRHs | Remote Radio Headers |
| RRLHs | Remote Radio Light Heads |
| RRM | Radio Resource Management |
| SDN | Software Defined Networks |
| SLA | Service Level Agreement |
| SLIs | Service Level Indicators |
| SLOs | Service Level Objectives |
| SmartNICs | Smart Network Interface Cards |
| SR-IOV | Single Root Input Output Virtualization |
| TEE | Trusted Execution Environment |

| | |
|---|---|
| UAV | Unarmed Aerial Vehicle |
| UPF | User Plane Function |
| URLLC | Ultra Reliable Low Latency Communication |
| vBNG | Virtual Border Network Gateway |
| vFW | Virtual Firewall |
| VIA | Virtualization Infrastructure Aggregator |
| VIM | Virtualisation Infrastructure Manager |
| VISP | Virtualization Infrastructure Service Provider |
| VMM | VM Monitor |
| VoLTE | Voice over LTE |
| vRAN | Virtual RAN |
| vRouter | Virtual Router |
| x86 | x86 is the generic name for instruction set architectures initiated by Intel processors |
| XOS | XaaS (Everything as a Service) Operating System |

# List of Contributors

| Name | Company / Institute / University | Country |
|---|---|---|
| **Editorial Team** | | |
| ***Overall Editors*** | | |
| David Artuñedo | Telefónica I+D | Spain |
| ***Section 2 Editors*** | | |
| Bessem Sayadi | NOKIA Bell Labs<br>5G-PPP Software Network WG Chairman | France |
| ***Section 3 Editors*** | | |
| Pascal Bisson | Thales Group | France |
| Jean Phillippe Wary | Orange | France |
| ***Section 4 Editors*** | | |
| Hakon Lonsethagen | Telenor | Norway |
| ***Section 5 Editors*** | | |
| Carles Anton-Haro | Centre Tecnològic Telecom. Catalunya (CTTC) | Spain |
| Antonio Oliva | Universidad Carlos III de Madrid (UC3M) | Spain |
| **Contributors** | | |
| Alexandros Kaloxylos | The 5GIA (Reviewer) | Belgium |
| John Cosmas | Brunel University | UK |
| Robert Muller | Fraunhofer Institute for Integrated Circuits IIS | Germany |
| Ben Meunier | Brunel University | UK |
| Yue Zhang | University of Leicester | UK |
| Xun Zhang | Institut supérieur d'électronique de Paris | France |
| Josep Mangues | CTTC (on behalf of 5G Transformer and 5Growth) | Spain |
| Carlos Bernardos | U. Carlos III Madrid (on behalf of 5G Transformer) | Spain |
| Xi Li | NEC Labs (on behalf of 5G Transformer) | Germany |
| Qi Wang | U. West Scotland (on behalf of Slicenet) | UK |
| Maria Barros | Eurescom (on behalf of Slicenet) | Germany |
| Amelie Werbrouck | SES (on behalf of Sat5G) | Luxembourg |
| Jesus Gutierrez Teran | IHP (on behalf of 5G Picture) | Germany |
| Marc Molla | Ericsson (on behalf of 5G EVE) | Spain |
| Manuel Lorenzo | Ericsson (on behalf of 5G EVE) | Spain |
| Dan Warren | SAMSUNG (on behalf of 5G VINNI) | UK |
| Valerio Frascolla | Intel (on behalf of 5Genesis, Reviewer) | Germany |
| David Artuñedo | Telefonica (on behalf of 5Genesis) | Spain |
| Fofy Setaki | COSMOTE (on behalf of 5Genesis) | Greece |
| Dimitris Tsolkas | FOGUS (on behalf of 5Genesis) | Greece |
| George Xiloutis | NCSR Democritos (on behalf of 5Genesis) | Greece |
| Harilaos Koumaras | NCSR Democritos (on behalf of 5Genesis) | Greece |
| Maciej Muehleisen | Ericsson (on behalf of 5GCroCo) | Belgium |
| Andreas Heider | T-systems (on behalf of 5G Carmen) | Germany |
| Kostas Trichias | WINGS (on behalf of 5GMobix) | Greece |
| Pascal Bisson | Thales (on behalf of 5G-Drones) | France |

| | | |
|---|---|---|
| Giannis Tzanettis | WINGS (on behalf of 5G Heart) | Greece |
| Kostas Katsaros | Digital Catapult (on behalf of 5G Victori) | UK |
| Lanfranco Zanzi | NEC lab (on behalf of MonB5G) | Germany |
| Gino Carrozzo | NEXTWORKS (on behalf of 5GZorro) | Italy |
| Antonio de la Oliva | U. Carlos III Madrid (on behalf of 5G-Dive) | Spain |
| Bilal Al-Jammal | NOKIA Bell Labs (5G-TOURS) | France |
| Aravinthan Gopalasingham | NOKIA Bell Labs (5G-TOURS) | France |
| Alejandro Fornés Leal | Universitat Politècnica de València (5GENESIS) | Spain |
| Marius Iordache | ORANGE Romania (5G-VICTORI) | Romania |
| Slawomir Kuklinski | ORANGE Poland (5G-VICTORI) | Poland |
| Lechoslaw Tomaszewski | ORANGE Poland (5G-VICTORI) | Poland |
| David Breitgand | IBM Research (5G-MEDIA/5G-ZORRO, Reviewer) | Israel |
| Paolo Giaccone | Politecnico di Torino (5G-EVE, Reviewer) | Italy |
| Pablo Serrano YAÑEZ-MINGOT | UC3M (5G-EVE) | Spain |
| Chrysa Papagianni | NOKIA Bell-Labs (5G-GROWTH) | Belgium |
| George Tsolis | CITRIX (MonB5G, Reviewer) | Greece |
| Christos Tselios | CITRIX (MonB5G) | Greece |
| John Cosmas | Brunel Univ | UK |
| David Levi | Ethernitynet | Israel |
| Kostas Trichias | Wings ICT Solutions (5G-EVE) | Greece |
| Hamzeh Khalili | i2CAT Foundation (SAT5G/5G-CLARITY) | Spain |
| Pouria Sayyad Khodashenas | I2CAT Foundation (SAT5G/5G-CLARITY) | Spain |