ETSI White Paper No. 11

Mobile Edge Computing
A key technology towards 5G

First edition – September 2015

ISBN No. 979-10-92620-08-5

Authors:

Yun Chao Hu, Milan Patel, Dario Sabella, Nurit Sprecher and Valerie Young

ETSI (European Telecommunications Standards Institute)
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Yun Chao Hu
Contributor, Huawei, Vice Chair ETSI MEC ISG, Chair MEC IEG Working Group

Milan Patel
Contributor, Huawei

Dario Sabella
Contributor, Telecom Italia; Vice-Chair MEC IEG Working Group

Nurit Sprecher
Contributor, Nokia; Chair ETSI MEC ISG

Valerie Young
Contributor, Intel

Contents

Introduction

Mobile Edge Computing (MEC) is a new technology which is currently being standardized in an ETSI Industry Specification Group (ISG) of the same name. Mobile Edge Computing provides an IT service environment and cloud-computing capabilities at the edge of the mobile network, within the Radio Access Network (RAN) and in close proximity to mobile subscribers. The aim is to reduce latency, ensure highly efficient network operation and service delivery, and offer an improved user experience.

Mobile Edge Computing is a natural development in the evolution of mobile base stations and the convergence of IT and telecommunications networking. Based on a virtualized platform, MEC is recognized by the European 5G PPP (5G Infrastructure Public Private Partnership) research body as one of the key emerging technologies for 5G networks (together with Network Functions Virtualization (NFV) and Software-Defined Networking (SDN)) [1]. In addition to defining more advanced air interface technologies, 5G networks will leverage more programmable approaches to software networking and use IT virtualization technology extensively within the telecommunications infrastructure, functions, and applications. MEC thus represents a key technology and architectural concept to enable the evolution to 5G, since it helps advance the transformation of the mobile broadband network into a programmable world and contributes to satisfying the demanding requirements of 5G in terms of expected throughout, latency, scalability and automation.

MEC is based on a virtualized platform, with an approach complementary to NFV: in fact, while NVF is focused on network functions, the MEC framework enables applications running at the edge of the network. The infrastructure that hosts MEC and NFV or network functions is quite similar; thus, in order to allow operators to benefit as much as possible from their investment, it will be beneficial to reuse the infrastructure and infrastructure management of NFV to the largest extent possible, by hosting both VNFs (Virtual Network Functions) and MEC applications on the same platform.

The environment of Mobile Edge Computing is characterized by low latency, proximity, high bandwidth, and real-time insight into radio network information and location awareness. All of this can be translated into value and can create opportunities for mobile operators, application and content providers enabling them to play complementary and profitable roles within their respective business models and allowing them to better monetize the mobile broadband experience.

Mobile Edge Computing opens up services to consumers and enterprise customers as well as to adjacent industries that can now deliver their mission-critical applications over the mobile network. It enables a new value chain, fresh business opportunities and a myriad of new use cases across multiple sectors. The intention is to develop favourable market conditions which will create sustainable business for all players in the value chain, and to facilitate global market growth. To this end, a standardized, open environment needs to be created to allow the efficient and seamless integration of such applications across multi-vendor Mobile Edge Computing platforms. This will also ensure that the vast majority of the customers of a mobile operator can be served.

The objectives of this white paper are to introduce the concept of Mobile Edge Computing and the related key market drivers, and to discuss the business and technical benefits of Mobile Edge Computing. A few examples of service scenarios that can benefit from the technology and possible deployment scenarios are presented. The white paper explains what is being standardized and how

innovation can be stimulated through using a standardized API. It outlines the specifications being produced by the Industry Specification Group on MEC.

ETSI ISG MEC encourages Proofs of Concept (PoC) to demonstrate the viability of MEC implementations. PoCs help to build awareness and confidence in the technology, and to develop a diverse and open MEC ecosystem. This white paper describes the MEC PoC framework and calls for active participation.

Market Drivers

The growth of mobile traffic and pressure on costs are driving a need to implement several changes in order to maintain quality of experience, to generate revenue, and optimize network operations and resource utilization. The Internet of Things is further congesting the network and network operators need to do local analysis to ease security and backhaul impacts. Enterprises want the ability to enable and engage with their customers with more efficient, secure and low latency connections. Application and content providers are challenged with the latency of the network when connecting to the cloud. These challenges need to be resolved.

Mobile operators need to shorten the time to launch new revenue generating applications for current customers but also for specific industries and sectors, such as but not limited to automotive, industry automation and welfare industries. Business transformation based on collaboration with the different players in the value chain can help in facing these challenges.

Technology improvements which provide low latency, better flexibility, agility, usage of virtualization, network and context awareness, etc. can provide the opportunity to increase the Quality of Experience of end users and make network operation more cost effective and competitive.

Smart phone applications and content are moving to the cloud. The access to the cloud needs to be optimised to guarantee a rich experience for consumers of an application or consumers of content and a tight collaboration is essential between network operators and application and content providers. This collaboration can lead to the deployment of applications/content at the edge of the mobile operator's network, providing awareness of the network and context information. Standardization will be essential to support this collaboration and the hosting of cloud or internet based applications within a multi-vendor environment.

The market drivers of MEC include business transformation, technology integration and industry collaboration (as illustrated in Figure 1). All of these can be enabled by MEC and a wide variety of use cases can be supported for new and innovative markets, such as e-Health, connected vehicles, industry automation, augmented reality, gaming and IoT services.

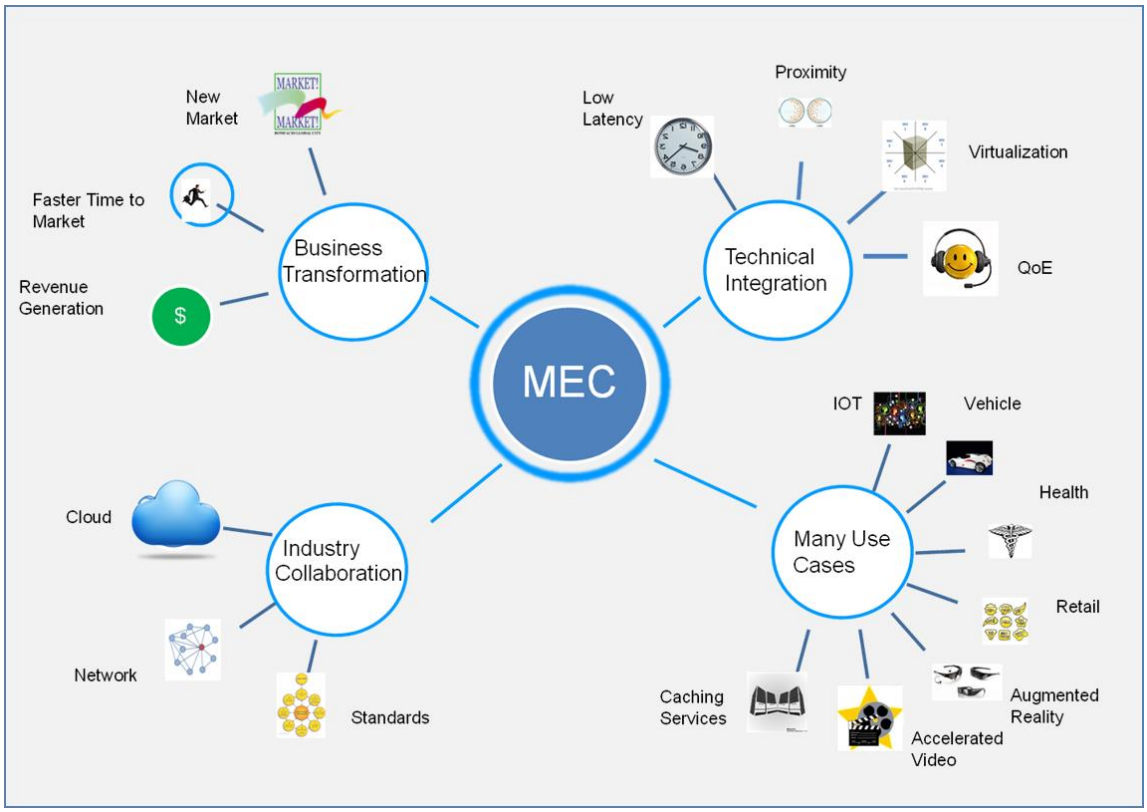


Figure 1: MEC market drivers

Business Value

Mobile Edge Computing offers an IT service environment at a location considered to be a lucrative point in the mobile network: the Radio Access Network (RAN) edge. Characterized by proximity, low latency and high bandwidth, this environment will offer localized cloud computing capabilities as well as exposure to real-time radio network and context information.

Opening up this IT service environment will allow applications and services from mobile operators, service and content providers to be efficiently and seamlessly integrated across multi-vendor, mobile edge computing platforms. The characteristics and capabilities offered by a MEC platform can be leveraged in a way that will allow proximity, context, agility and speed to be used for wider innovation that can be translated into unique value and revenue generation.

Access to content and applications can be accelerated; their responsiveness can be increased, maximizing speed and interactivity. Popular and locally-relevant content can be delivered directly where users connect, limiting ingress bandwidth to the core and cloud.

Knowledge of real-time radio network conditions and context information can be used to optimize the network and service operation (responding and adapting to changing network conditions). This would improve service experience and the utilization of network resources, enabling them to efficiently handle increased amounts of traffic. Real-time network and fine-granular context information (including

location) could be used to enrich the mobile broadband experience by creating highly personalized services which are tailored to individual needs and preferences.

Operators can reposition themselves in the value chain and redefine personalized services. They can capitalize their networks and open them up to authorized third-parties (in a secure way), exposing capabilities to Over the Top (OTT) players and application developers to flexibly, agilely and rapidly deploy innovative applications and services towards mobile subscribers, enterprises and vertical segments. Operators will be able to create new revenue streams, delight their customers by developing a new breed of applications that provide incremental value, and open up new market opportunities. In addition, applications supporting tighter integration of network and service parameters will improve both service experience and utilization of the network resources.

Application service providers, OTT players and independent software vendors will be able to translate proximity and context into value, and be able to generate new revenue. Their applications and services can be enhanced and accelerated to provide a unique and unparalleled experience. Innovative applications can be deployed rapidly in a new standards-based environment, taking advantage of new levels of flexibility and agility. Applications will be able to expand their cloud into the mobile network and create a whole new set of services. They will be able to feel and react to end-user experience in real time, based on the actual radio conditions. The new MEC specifications will allow applications and services to be deployed on top of multi-vendor Mobile Edge Computing platforms, enabling them to be used by the vast majority of the customers of a single mobile operator. The mobile end user will enjoy a unique, gratifying and personalized mobile-broadband experience.

The MEC initiative will help to develop favourable market conditions for all players in the value chain as well as facilitate economic growth with a myriad of new use cases across multiple sectors (see Figure 2).

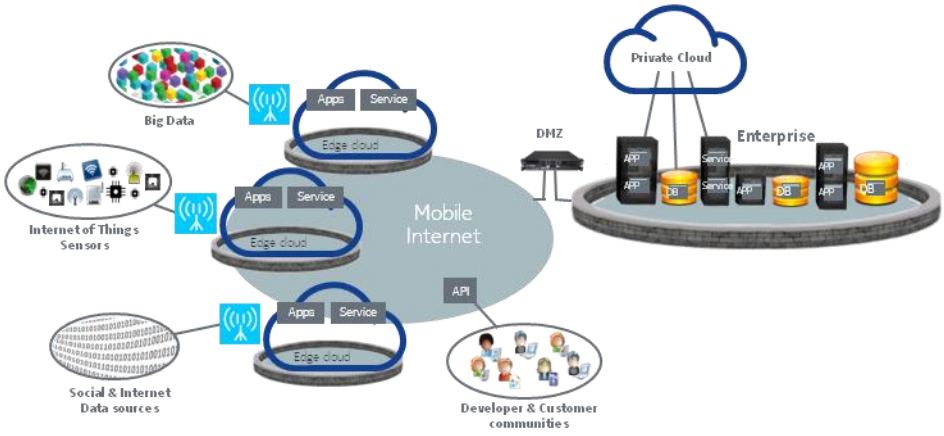


Figure 2: Improved QoE with Mobile Edge Computing in close proximity to end users

Mobile Edge Computing Service Scenarios

General

The following sections describe a number of service scenarios that have been considered within ETSI ISG MEC. These illustrate various scenarios which can take advantage of Mobile Edge Computing to either

increase performance compared to providing such services through the cloud or through core network servers, or to utilize the unique capabilities offered by MEC platforms such as proximity to the user and network edge, serving a highly localized area. It should be noted these examples are non-exhaustive and further service scenarios are available in the ETSI ISG MEC specification for Mobile Edge Computing Service Scenarios, GS MEC 004. Other scenarios which can make use of MEC are also possible.

Augmented Reality

New services become possible when mobile networks supporting high data rates and low latency computation are deployed. One example of such services is Augmented Reality. Augmented reality (AR) is the combination of a view of the real-world environment and supplementary computer-generated sensory input such as sound, video, graphics or GPS data. Augmented reality can enhance the experience of a visitor to a museum or another point of interest. Consider a visitor to a museum, art gallery, city monument, music or sports event, holding their mobile device towards a particular point of interest with the application related to their visit activated (i.e., the museum application). The camera captures the point of interest and the application displays additional information related to what the visitor is viewing.

Augmented reality services require an application to analyse the output from a device's camera and/or a precise location in order to supplement a user's experience when visiting a point of interest by providing additional information to the user about what they are currently experiencing. The application needs to be aware of a user's position and the direction they are facing, either through positioning techniques or through the camera view, or both. After analysing such information, the application can provide additional information in real-time to the user. If the user moves, the information needs to be refreshed. Hosting the Augmented Reality service on a MEC platform instead of in the cloud is advantageous since supplementary information pertaining to a point of interest is highly localised and is often irrelevant beyond the particular point of interest. Figure 3 shows how a MEC platform can be used to provide an Augmented Reality service.

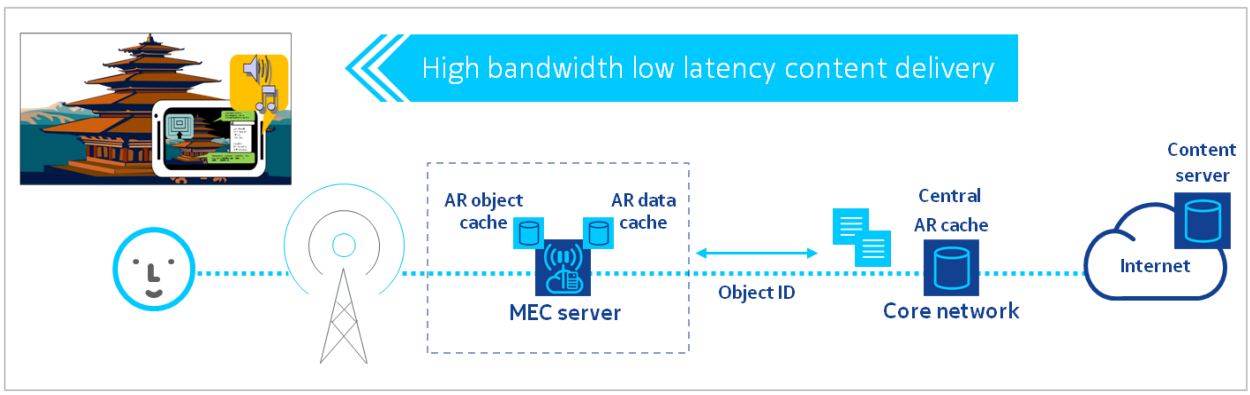


Figure 3: Augmented Reality Service Scenario

Additionally, the processing of user location or camera view can be performed on a MEC platform rather than on a more centralized server. There may be a need to update information at a fast rate, depending on how the user moves, and the context in which the augmented reality service is used (e.g. in an art gallery, exhibits are positioned only a few metres apart and each piece is supplemented with additional text on the artist, the interpretation of the artwork, etc.) In other words, augmented reality data requires low latency and a high rate of data processing in order to provide the correct information to

the user's device depending on the location and orientation of the user. Performing such data processing on the MEC platform also has the advantage of collecting metrics, anonymized meta-data, etc., in order to analyse the service usage and help to improve the service in order to provide a better user experience.

Intelligent Video Acceleration

End user Quality of Experience (QoE) and utilization of radio network resources can be improved through intelligent video acceleration. Internet media and file delivery are typically streamed or downloaded today using Hypertext Transmission Protocol (HTTP) over the TCP protocol. Available capacity can vary by an order of magnitude within seconds (as a result of changes in radio channel conditions, devices entering/leaving network). TCP may not be able to adapt fast enough to rapidly-varying conditions in the radio access network (RAN). This may lead to under-utilisation of precious radio resources and to a sub-optimal user experience.

Figure 4 shows an example of the intelligent video acceleration service scenario which attempts to overcome the challenges described above. In this scenario, a radio analytics application, which resides in a MEC server, provides the video server with an indication on the throughput estimated to be available at the radio downlink interface. This information can be used to assist the TCP congestion control decisions (for example in selecting the initial window size, setting the value of the congestion window during the congestion avoidance phase, and adjusting the size of the congestion window when the conditions on the "radio link" deteriorate). The information can also be used to ensure that the application-level coding matches the estimated capacity at the radio downlink.

The video server may use this information to assist TCP congestion control decisions (for example by ensuring that the application level coding matches the estimated capacity at the radio downlink). The content's time-to-start as well as video-stall occurrences can be reduced, enabling improved video quality and throughput.

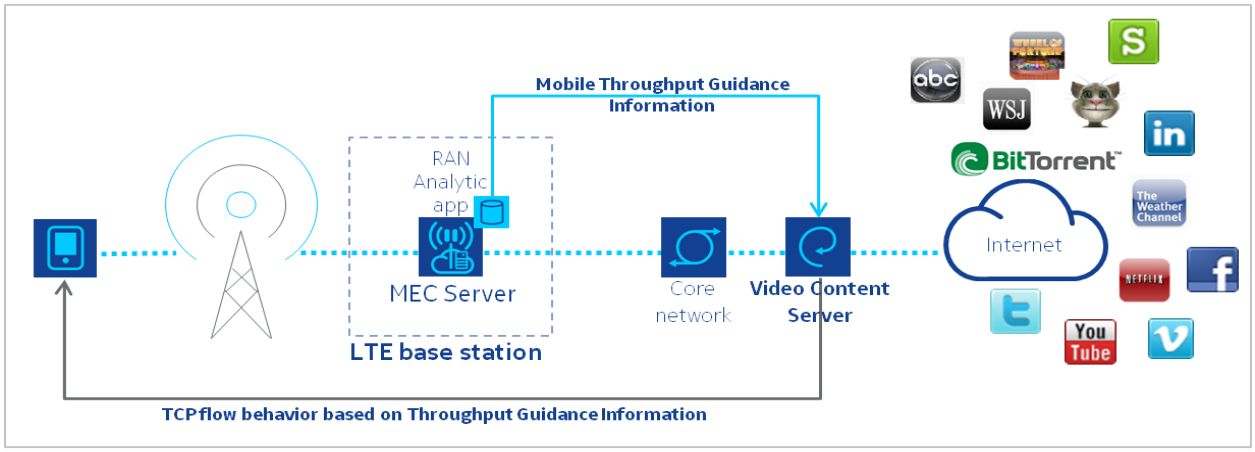


Figure 4: Intelligent Video Acceleration Service Scenario

Connected Cars

The number of connected vehicles is rapidly growing and will continue to do so over the coming years. Communication of vehicles and roadside sensors with a roadside unit is intended to increase the safety, efficiency, and convenience of the transportation system, by the exchange of critical safety and

operational data. The communication can also be used to provide value added services, such as car finder, parking location and to support entertainment services (e.g. video distribution).

As the number of connected vehicles increases and use cases evolve, the volume of data will continue to increase along with the need to minimize latency. Whilst data stored and processed centrally may be adequate for some use cases, it can be unreliable and slow for others.

LTE can significantly accelerate the deployment of connected vehicle communications. LTE cells can provide "beyond the line of sight" visibility i.e. beyond the range of direct communication between vehicles of 300 – 500m. It can also satisfy the tight latency requirement of connected vehicle communications, below 100ms in some use cases. Messages could be distributed in real time over LTE, eliminating the need to build a countrywide Digital Short-Range Communications (DSRC) network. Cars can leverage their increasingly inbuilt LTE connectivity; in deployments where DSRC exists, LTE would be able complement it.

Mobile Edge Computing can be used to extend the connected car cloud into the highly distributed mobile base station environment, and enable data and applications to be housed close to the vehicles. This can reduce the round trip time of data and enable a layer of abstraction from both the core network and applications provided over the internet. MEC applications can run on MEC servers which are deployed at the LTE base station site to provide the roadside functionality. The MEC applications can receive local messages directly from the applications in the vehicles and the roadside sensors, analyse them and then propagate (with extremely low latency) hazard warnings and other latency-sensitive messages to other cars in the area (as depicted in Figure 5). This enables a nearby car to receive data in a matter of milliseconds, allowing the driver to immediately react.

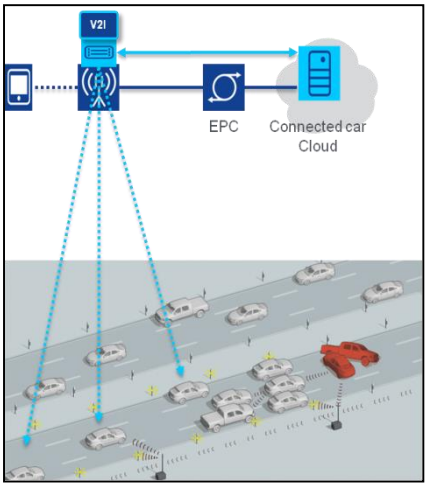


Figure 5: Connected Vehicles Service Scenario

The roadside MEC application will be able to inform adjacent Mobile Edge Computing servers about the event(s) and in so doing, enable these servers to propagate hazard warnings to cars that are close to the affected area.

The roadside application will be able to send local information to the applications at the connected car cloud for further centralized processing and reporting.

Internet of Things Gateway

The Internet of Things (IoT) generates additional messaging on telecoms networks, and requires gateways to aggregate the messages and ensure low latency and security. Because of the nature of some of the devices being connected, a real time capability is required and a grouping of sensors and devices is needed for efficient service.

IoT devices are often resource constrained in terms of processor and memory capacity. There is a need to aggregate various IoT device messages connected through the mobile network close to the devices. This also provides an analytics processing capability and a low latency response time.

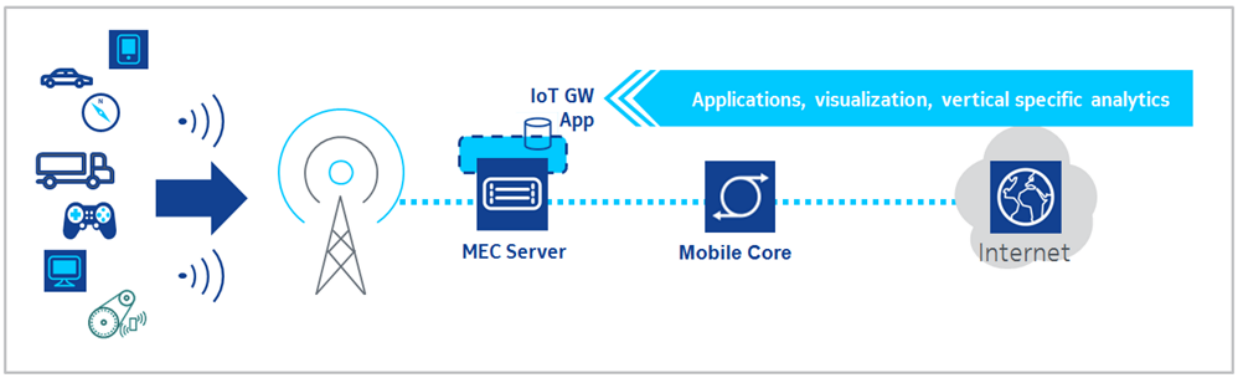


Figure 6: IoT Gateway Service Scenario

Various devices are connected over different forms of connectivity, such as 3G, LTE, Wi-Fi or other radio technologies. In general the messages are small, encrypted and come in different forms of protocols. There is a need for a low latency aggregation point to manage the various protocols, distribution of messages and for the processing of analytics. The MEC server provides the capability to resolve these challenges.

Mobile Edge Computing can be used to connect and control devices remotely, analyse and provide real time provisioning and analytics. MEC enables the aggregation and distribution of IoT services into the highly distributed mobile base station environment, and enable applications to respond in real-time. This can reduce the round trip time of data and enable a layer of abstraction from both the core network and applications in the cloud. IoT applications can run on MEC servers which are deployed at the LTE base station site to provide this functionality.

Deployment Scenarios

Mobile Edge Computing servers can be deployed at multiple locations, such as at the LTE macro base station (eNodeB) site, at the 3G Radio Network Controller (RNC) site, at a multi-Radio Access Technology (RAT) cell aggregation site, and at an aggregation point (which may also be at the edge of the core network). The multi-RAT cell aggregation site can be located indoors within an enterprise (e.g. hospital, large corporate HQ), or indoors/outdoors for a special public coverage scenario (e.g. stadium, shopping mall) to control a number of local multi-RAT access points providing radio coverage to the premises. This deployment option enables the direct delivery of locally-relevant, fast services from base station clusters. Where a MEC platform is deployed may depend on a number of factors, including scalability,

physical deployment constraints, performance criteria (e.g. latency) and which network information will be exposed. Note that some MEC services may not be available / applicable in certain deployment options.

MEC applications can be intelligently and flexibly deployed in a seamless manner on different MEC platforms based on technical and business parameters. The deployment of MEC applications on a particular MEC platform may depend on the availability of specific MEC services and on other parameters, such as latency requirements, required resources, availability of a particular VNF, scalability considerations, cost etc.

MEC will utilise the NFV infrastructure. The NFV platform may be dedicated to MEC or shared with other network functions or applications. MEC will also use (as much as possible) the NFV management and orchestration entities and interfaces.

ETSI Industry Specification Group on Mobile Edge Computing

The ETSI Industry Specification Group (ISG) on Mobile Edge Computing (MEC) produces normative Group Specifications that will enable the hosting of third-party applications in a multi-vendor MEC environment. Launched in December 2014, the group plans to deliver the first set of specifications within 2 years.

The initial scope of the ISG MEC focuses on use cases; it specifies the requirements and the reference architecture, including the components and functional elements that are the key enablers for MEC solutions.

When the first documents reach the required maturity level, work on platform services, APIs and interfaces will commence. The MEC platform API will be application-agnostic and will allow the smooth porting of value-creating applications on every mobile-edge server with guaranteed SLA (see Figure 7).

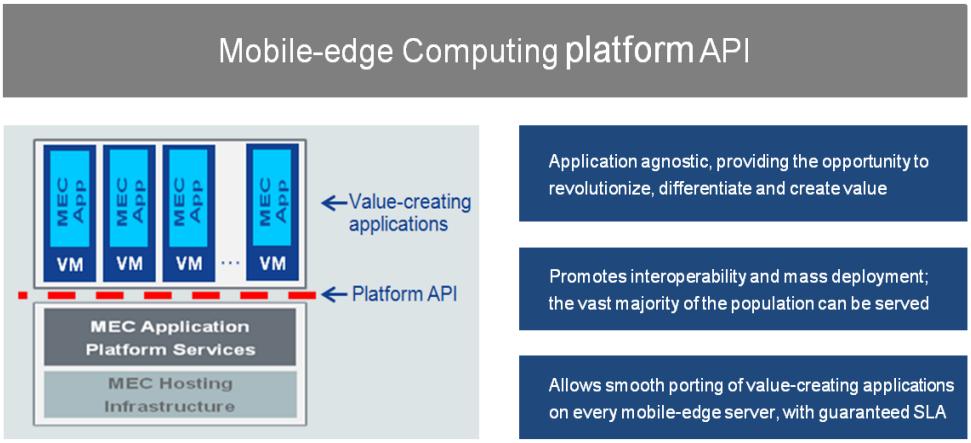


Figure 7: MEC platform API

In addition to the technical work, an industry-enabling working group (IEG WG) has been setup under the ISG MEC which is tasked with advancing Mobile Edge Computing in the industry and accelerating the

adoption of the concept and the standards. This will help to develop favourable market conditions for sustainable business for all players in the value chain.

The IEG group is developing two Group Specifications:

1. ETSI GS MEC-IEG 005 [2] Proof of Concept Framework, specifying the process and criteria that a Proof of Concept demonstration must adhere to. This specification has already been published;
2. ETSI GS MEC-IEG 004 Service Scenarios, which presents a number of examples of service scenarios, business and consumer benefits which can be enabled by Mobile Edge Computing.

The ETSI ISG is open to members and non-members of ETSI to participate and contribute towards this innovative technology. Industry players are invited to actively participate and contribute to the work on Mobile Edge Computing and to take part in the PoC activities. More information on how to participate is available at: http://www.etsi.org/mobile-edge-computing/get-involved-in-mec.

Proofs of Concept

In order to ensure the success and widespread deployment of Mobile Edge Computing, it is necessary to have more than just timely and high quality specifications. It is crucial to validate the specifications that are being developed, and to demonstrate that the use cases have been fulfilled. Furthermore, the MEC concept must be demonstrated to be feasible and valuable to all the major stakeholders in the value chain in order to appeal to the widest possible audience.

To showcase the Mobile Edge Computing concept, the ISG MEC has developed a Proof of Concept process, specified in GS MEC-IEG 005 [2]. A PoC proposal can be submitted by a PoC team consisting of at least one Mobile Network Operator, at least one infrastructure vendor and at least one content or application provider. GS MEC-IEG 005 [2] specifies the process and criteria that a Proof of Concept demonstration must adhere to in order to be accepted as a MEC PoC. A wiki site http://mecwiki.etsi.org has been put in place to assist PoC teams. This site hosts the PoC project proposal templates and the list of PoC Topics, which are specific areas where input or feedback from PoCs is needed. One or several PoC Topics can be addressed by a single PoC project.

The public demonstration of MEC Proofs of Concept helps to build commercial awareness and confidence in this technology, and helps to develop a diverse, open, MEC ecosystem.

Conclusions

Mobile Edge Computing enables innovative service scenarios that can ensure enhanced personal experience and optimized network operation, as well as opening up new business opportunities. A few examples are described in this white paper to demonstrate how proximity to users and objects, together with network and context information can be leveraged by applications to create value. Mobile Edge Computing attracts a new value-chain and energized eco-system, where all players can benefit from tighter collaboration. Mobile operators can play a pivotal role within the new value chain and attract OTT service providers, developers and Internet players to innovate over a new cutting-edge technology, while enabling context-aware applications to run in close proximity to the mobile subscriber. Mobile subscribers can enjoy a unique, truly gratifying and personalized mobile-broadband experience which is tailored to their needs and preferences.

Based on a virtualized platform, MEC complements NFV and is fully aligned with the emerging distributed cloud approach. It is recognised as a key technology of the future 5G era, satisfying the demanding requirements for ultra-low latency and stimulating innovation.

The MEC technology is defined by the ETSI ISG MEC, which was launched in December 2014 with the intention to develop the first set of specifications within 2 years. The deliverables will include service scenarios, requirements, architecture and API specifications, complemented with a PoC Framework specification and White Papers which aim to accelerate the market adoption of the MEC technology.

MEC supports different deployment options, as MEC Servers can be located at different places within the Radio Access Network depending on technical and business requirements.

A MEC Proof-of-Concept (PoC) program has been established to demonstrate the viability of MEC implementations. The results and lessons learned by the MEC PoCs are fed back to the ISG MEC specification activities.

The MEC ISG is developing the foundation to enable an open radio access network which can host third party innovative applications and content at the edge of the network. The ISG is open to members and non-members of ETSI to participate and contribute towards this innovative technology, and to take part in the PoC activities.

References

[1] 5G Vision: The 5G Infrastructure Public Private Partnership: the next generation of communication networks and services. From The 5G Infrastructure Public Private Partnership: https://5g-ppp.eu/wp-content/uploads/2015/02/5G-Vision-Brochure-v1.pdf

[2] ETSI GS MEC-IEG 005; Mobile Edge Computing (MEC); Proof of Concept Framework.

ETSI (European Telecommunications Standards Institute)
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

ETSI White Paper No. 20

Developing Software for Multi-Access Edge Computing

First edition – September 2017

ISBN No. 979-10-92620-14-6

Authors:

Alex Reznik (editor), Rohit Arora, Mark Cannon, Luca Cominardi, Walter Featherstone, Rui Frazao, Fabio Giust, Sami Kekki, Alice Li, Dario Sabella, Charles Turyagyenda, Zhou Zheng

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Alex Reznik (editor)
HPE

Rohit Arora
HPE

Mark Cannon
Virtuosys

Luca Cominardi
UC3M

Walter Featherstone
Viavi Solutions

Rui Frazao
Vasona

Fabio Giust
NEC

Sami Kekki
Huawei

Alice Li
Vodafone

Dario Sabella
Intel

Charles Turyagyenda
InterDigital

Zhou Zheng
Huawei

Contents

Introduction

Edge Computing refers to a broad set of techniques designed to move computing and storage out of the remote cloud (public or private) and closer to the source of data. For the emerging class of "5G Applications" this is often a matter of necessity. Locating such applications in a traditional cloud does not allow one to meet certain stringent requirements, such as roundtrip latency. In other cases, such as the Internet of Things (IoT) and Vehicle to everything communication (V2X), the amount of data is expected to increase rapidly. Edge computing can mitigate this by collecting and processing the data closer to the user.

ETSI MEC (Multi-access Edge Computing) Industry Specification Group (ISG) focuses on enabling edge computing at the access network (mobile or otherwise), thus bringing edge computing as close as possible to the user without it being in the user device. The group was established in September 2014 to standardize APIs that will enable application and content providers to utilize computing capabilities present at the edge of the network. MEC enables successful deployment of new use cases like augmented reality, connected vehicles, etc. and various services can be customized according to the customer requirements and demands.

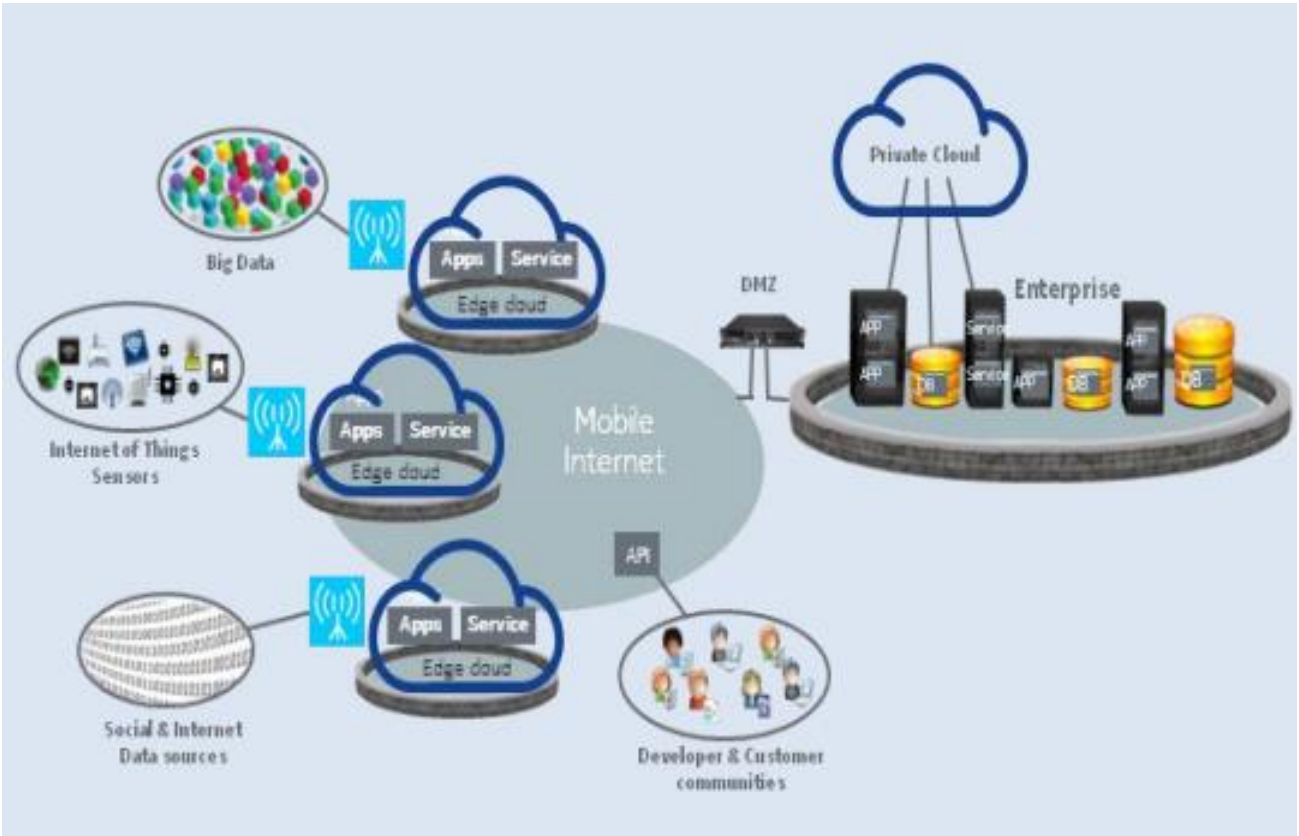


Figure 1: Overview of the MEC system

The current prevalent distributed computing software development model uses a client side to initiate server requests and a remote server side to process these requests (the client-server model). This allows application developers to take advantage of centralized compute and storage and has been a major driver of the emergence of cloud computing. However, for MEC applications, developers need to

identify features of their applications that require processing at the edge as distinct from features that require high compute power or that do not require near real-time response and can, therefore, be deployed at a central location.

This idea is quite recent, although not totally new, and the ecosystem is quickly moving to use systems like Greengrass for Amazon's AWS Lambda, Microsoft's Azure IoT stack and GE's Predix to enable it. Let's take, for example, AWS Greengrass. This consists of the AWS Greengrass core (which is responsible for providing compute capabilities closer to the devices) and the AWS IoT devices enabled with AWS IoT Software Development Kit (SDK). Using this architecture, AWS IoT applications can in real time respond to local events and also use cloud capabilities for certain functions that don't require real time processing of data. An IoT application developer targeting AWS Greengrass has to architect the application in a way that uses these edge systems for certain features that require real time processing or which perform some other useful tasks (e.g. limiting the data flood to the central location), while keeping other features in the traditional cloud.

To provide these new services and to make the most out of MEC it is also important for the application developers and content providers to understand the main characteristics of MEC environment and the additional services which distinguish MEC from other "edge computes", namely: extreme user proximity, ultra low latency, high bandwidth, real time access to radio network and context information and location awareness.

On this basis this white paper provides guidance for software developers on how to properly approach architecting and developing applications with components that will run in edge clouds, such as those compliant with ETSI's MEC standards. The white paper will summarize the key properties of edge clouds, as distinct from a traditional cloud point-of-presence, as well as the reasons why an application developer should choose to design specifically for these. It will then provide high-level guidance on how to approach such design, including interaction with modern software development paradigms, such as micro-services based architectures and DevOps.

The Need for an Evolved Approach

MEC offers to application developers and content providers cloud-computing capabilities and an IT service environment at the edge of the network. As a consequence, MEC introduces a standard for supporting an emerging cloud paradigm for software development communities. In fact, up to now a "traditional" client-server model of application development has been the dominating approach to developing applications for at least 2 decades. The emergence of edge computing, e.g. MEC, evolves this environment, by introducing an intermediate element at the network edge.

A MEC point-of-presence (PoP) is distinct from a traditional cloud PoP. It may offer significant advantages to application components/services running there, while also presenting some challenges, e.g. higher cost, relatively small compute footprint, good local but not global reachability, etc. As such, it is crucial for an application developer to design with specific intent towards running some application components at the network edge when developing for MEC.

This results in a new development model with 3 "locations": Client, Near Server, Far Server (depicted in Figure 2. The client location can be a traditional smartphone or other wireless connected compute elements in a car, smart home or industrial location that can run dedicated client applications. The model is quite new to the majority of software developers, and while modern development paradigms (e.g. microservices) make it easier to adapt to it, a clear and concise summary of this new development model and guidance on how to properly approach it will help accelerate the application development for the network edge and thus accelerate MEC adoption.

As depicted in Figure 2, a MEC Host, usually deployed at the network edge, contains a MEC platform and provides compute, storage and network resources. The MEC platform offers a secure environment where MEC applications may, via RESTful APIs, discover, advertise, consume and offer services.

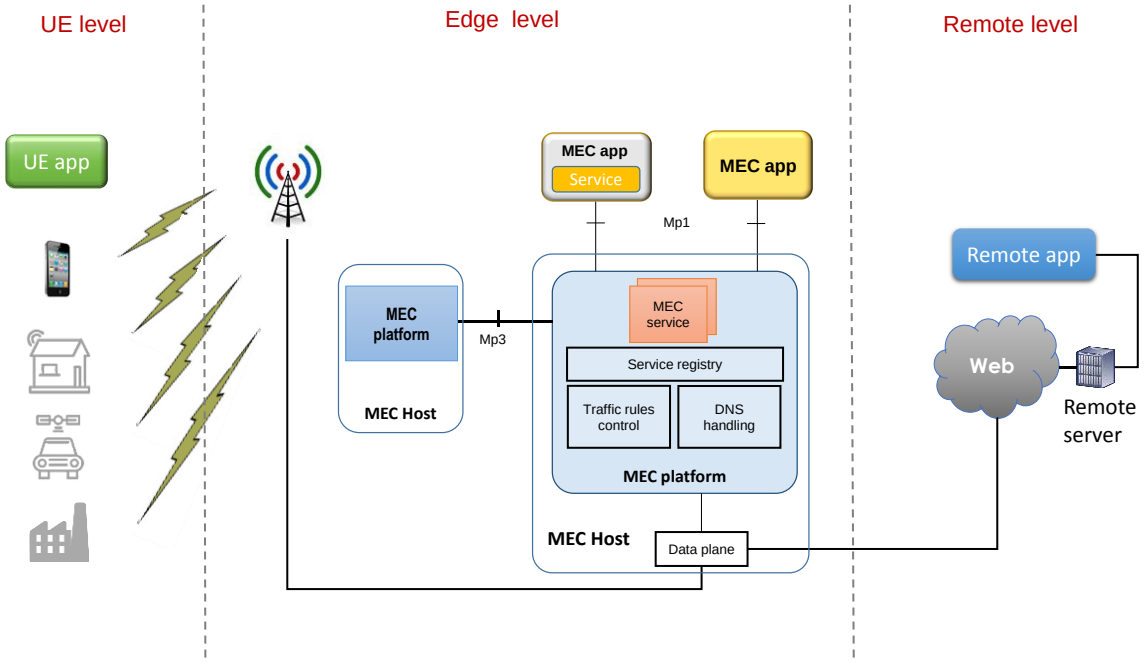


Figure 2: New application development paradigm introduced by MEC.

Designing with the Edge in Mind

A key principle of modern system design, known as the hour-glass model (and realized with IP as the single protocol at the waist of the hour-glass) is that applications and networks should be completely agnostic to each other. This has been key to the success of the Internet and the millions of applications running over it. It allows applications to run over any IP-based network, while any network can use IP to support pretty much any IP-based application. Nevertheless, when it comes to actual application performance, both aspects (the application design and the network) are important and remaining completely agnostic becomes difficult. To date, few traditional Internet applications encountered issues with this approach since the requirements of applications have traditionally been much looser than the performance that networks can deliver. This is about to change. Already practices are emerging that consider network aspects as part of application design. A widely used example is adapting the application behaviour to throughput limitations in the network, e.g. adjusting the video stream compression ratio in response to throughput throttling, while also taking into account the application state, e.g. whether the application is active or idle or in suspended state, etc. Nonetheless, network conditions and topology characteristics are typically considered during the software design phase as an environmental input out of the programmer's control and the application *passively* adapts to these.

The MEC platform is where the network and the applications can converge in a meaningful way without giving up the key benefits of the hour-glass model. MEC can support any application and any application can run in MEC. However, MEC can offer additional services to those applications which have been designed to be MEC-aware. MEC Application Enablement (described in ETSI GS MEC 011) introduces such a service environment, and this can be used to improve the user experience tremendously. Software designed to take advantage of MEC services can leverage additional information about where the application is supposed to run, in terms of expected latency, throughput and other available MEC services. Simply put, with MEC, the environment becomes *less unpredictable* and environmental (i.e. contextual) information can be leveraged to *actively* adjust the application behaviour in run time. The network becomes a resource used to deliver the end to end service. For example, an application is able to not only precisely monitor the radio link via the MEC Radio Network Information Service, but also make a bandwidth request and inform the network of other application requirements via APIs provided by the MEC platform. This allows edge applications to benefit from low latency and high throughput in a fairly predictable/controllable way. In addition, the network itself could also benefit from the MEC services provided by the applications, for instance the network scheduler could also predict the incoming user behaviour to maximize the network efficiency.

A key aspect of software design for MEC is the need to split the application into *terminal device component(s)*, *edge component(s)* and *remote component(s)*. This aspect creates an additional task for developers with respect to a more traditional client-server architecture, since an additional processing stage (at the edge) must be added to the application's workflow, with well-defined characteristics and capabilities. The terminal device can do some preliminary processing to determine the need for further actions. Such preliminary processing requires near zero latency and it requires the terminal device to support some computing capabilities, e.g. to receive and instantiate algorithms or instructions. The edge component(s) include a set of operations that the application performs at the edge cloud, e.g. to offload the computing away from the terminal device while still leveraging very low latency and predictable performance. The remote components implement operations to be carried out in the remote data centre, e.g. to benefit from large storage and database access. This concept should not be confused with

the traditional software modularization and distribution of components. This latter vision takes into account the division of tasks, e.g. to improve the development and maintenance of the code, but, usually, the different components either run in the same data centre, or are sparsely distributed at unpredictable locations (e.g. P2P applications). Software modularization is not only possible in MEC, but also encouraged in order to assign execution tasks at the most appropriate location. In particular, as discussed below, a microservices-based architectural approach is particularly well suited for MEC.

An example of the concept expressed above comes from applications for video analytics from surveillance cameras. Here, the task is to extract information for face recognition from different video streams. In a traditional approach, the video streams are transported up to the data centre where the software components for face recognition, databases, etc. are located. A lot of bandwidth can be saved by splitting the job and performing some functions in the edge or even in the terminal (if the terminal has the necessary computational power). For instance, image areas with suspected faces can be cropped from HD images, compressed, and then sent to the remote cloud, for further processing (i.e. final face identification) and database lookup. When cameras are active, they are constantly searching for pre-defined objects or events. To avoid the need to stream the video constantly in real time to the customer's facilities (e.g. central cloud) the terminal device can do some preliminary processing. Based on the preliminary processing, further action may be necessary on the selected objects, e.g. further analysis of the object's track, detailed identification of the object, etc. Such processing may be delegated to the edge cloud component of the application. The example is illustrated in Figure 3.

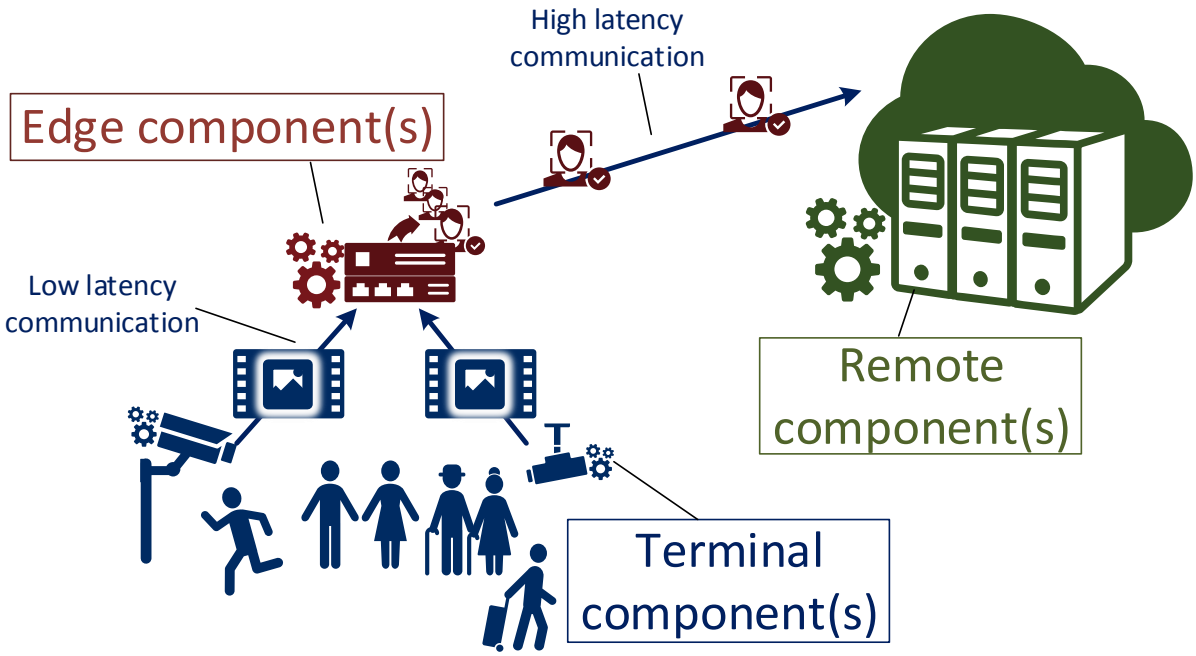


Figure 3: Example of splitting an application into "terminal", "edge" and "remote" components

What is worth highlighting here is that MEC can be exploited to implement computation offloading techniques among all the application's components. In fact, the server can be programmed to dynamically shift the processing among the terminal, the edge and the remote component(s), for a number of reasons ranging from adaptation to network conditions, improving application specific KPIs,

policies, costs, etc. The processing distribution may be driven by certain performance objectives, e.g. providing the best user experience.

However, the concept above should not be disguised as yet another load balancing technique. In its traditional sense, load balancing stands for replicating many instances of the application's server side in order to increase the number of clients in service at the same time. In MEC, load balancing can be performed within the same server instance in order to exploit characteristic of the environment where the server component is running.

Another aspect inherent to MEC is that service providers can exploit the geographical distribution to serve different user populations and thus tailor their service knowing the peculiarities of the covered area. For instance, media content can be adapted to the linguistic and cultural characteristics of a given area, or customized advertising can be tailored to the needs of local businesses. Content Delivery Networks are only one, but perhaps the most straightforward example of a use case that benefits from geospatial distribution. Nevertheless, most of content provided by today's CDNs is not as delay sensitive as the services from other MEC applications, e.g. for virtual/augmented reality, for which it is crucial to run at a specific location to meet the application's stringent latency requirements.

Architecting and Developing for the Edge

ETSI MEC defines a platform that interfaces with MEC applications and other platforms. The MEC platform is responsible for a myriad of functions, including: receiving DNS records and configuring a DNS proxy/server; hosting MEC services, such as: Radio Network Information, Location and Bandwidth Manager. Each application instantiated at the edge would likely only utilize a subset of the totality of functions provided by the MEC platform(s).

Developing a MEC application using monolithic software design principles requires integration of the MEC application and a subset of MEC platform functions into a self-contained, tightly coupled software program. Updates to any MEC platform or program component would necessitate re-writing and re-deploying the entire application.

In contrast, a microservices-based design paradigm would structure a MEC application as a collection of loosely coupled autonomous services working together, i.e. the MEC platform functions could be implemented as microservices with each microservice as a separate entity with no dependency on other microservices forming the MEC application. The microservices interact with each other to implement the logic of the application and communicate via network calls to enforce separation and avoid tight coupling. Consequently, each microservice exposes an application programming interface (API) for other microservices to communicate in collaborative way. One of the most common API paradigms is Representational State Transfer (REST) that provides a uniform and predefined set of stateless operations to access and manipulate resources. The benefits of microservices include.

- A single microservice can be deployed independently of the rest of the system. This allows to deploy some of the components at the edge while keeping others at distinct locations (e.g., at the cloud);
- Microservices permit the use of different technologies inside each microservice and to freely replace the technology stack while the API towards the other microservices remains the same;
- Adapting existing microservices applications does not require refactoring of the whole application;
- Microservices permit scaling of only those microservices that need scaling while keeping the rest of the application untouched;
- Microservices provide better fault isolation, i.e. if one microservice fails, the others continue to function;
- With microservices, it is easy to integrate with 3rd party microservices;
- Microservice-based applications are easily implemented using containers;
- Isolation of the individual microservices simplifies security-related design.

Figure 4, presents the microservices architecture. Each microservice manages its unique data and exposes an API to other microservices. It is worth noting that each microservice may expose a different type of API. Additionally, an API gateway may be used to convey MEC application requests to the appropriate microservice.

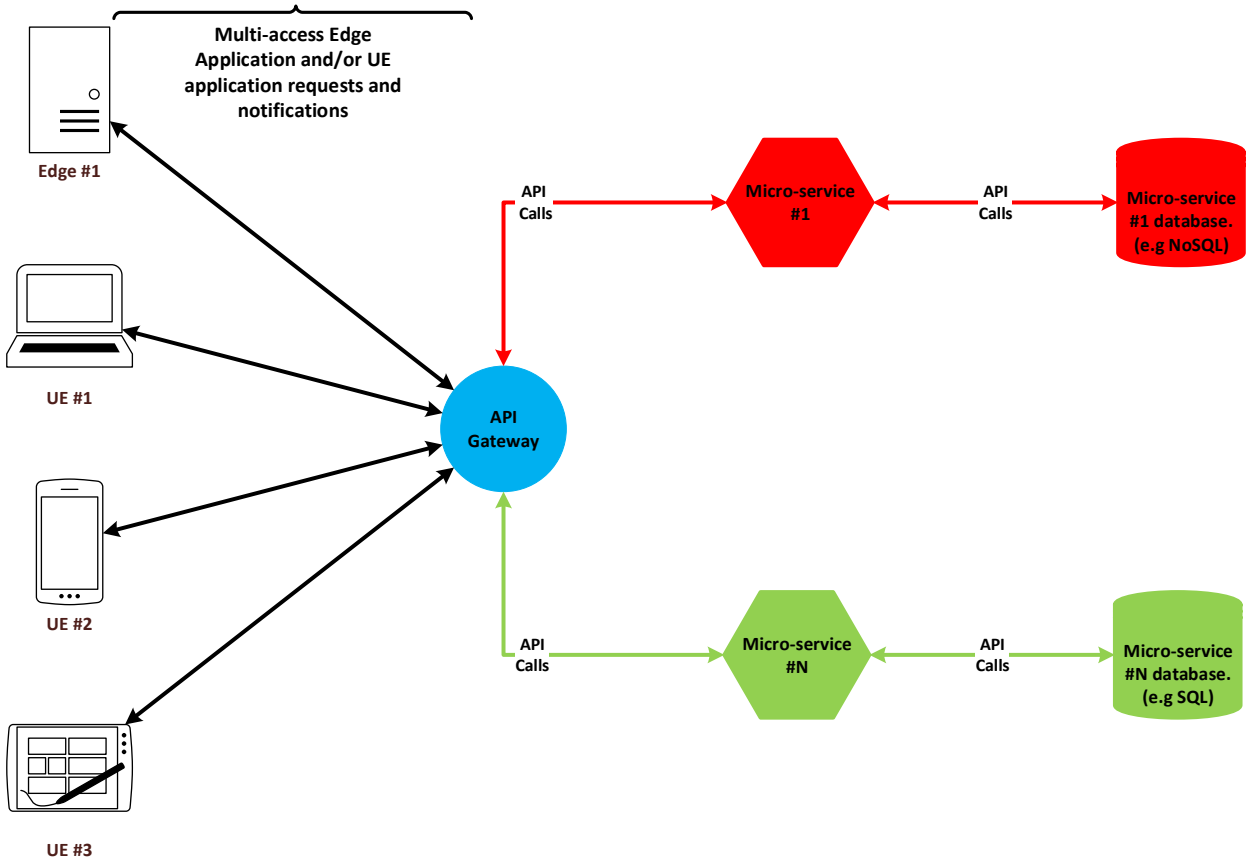


Figure 4: Microservices architecture

Consider an MEC augmented reality application composed of three microservices: one on the client-side, one at the edge, and one at the cloud. The first microservice takes care of collecting the input from the device (e.g., a smartphone) and communicating the data to the microservice residing at the edge. The microservice on the client is also responsible for receiving the computed data from the edge and to displaying it to the user. An API gateway is used in this case to allow the augmented reality logic to support multiple visualization targets at once (e.g., smartphone, tablet, laptop, etc.). The microservice running at the edge is responsible for processing the inputs from the client and combining them to create the augmented reality. This microservice may subscribe to MEC services to improve the quality of experience of the end-user. For example, the microservice may subscribe to the MEC Radio Network Information Service to react to the channel status of the user and enable/disable some augmented reality features depending on the connection quality. Finally, the cloud microservice is responsible for collecting statistics of the users using the augmented reality application.

Edgy DevOps

In the previous sections, it has been highlighted how the microservices variant of the service-oriented architecture (SOA) pattern has direct applicability to the software model envisaged for MEC and its services and applications. The approach's ability to compose a distributed application from separately deployable services was also described. Such services are expected to communicate via web interfaces (such as MEC's RESTful API approach) and perform a specific business function. In combination with this approach DevOps practices are considered as being highly complementary.

DevOps is considered to be a business-driven software delivery approach. It is based on lean and agile principles in which cross-functional teams collaborate to deliver software in a continuous manner. Such teams consist of representatives from all disciplines responsible for developing and deploying software services, including the business owners, developers, operations and quality assurance. This continuous delivery (CD) based approach enhances a business' ability to exploit new market opportunities and quickly adapt to customer feedback. The MEC ecosystem is evolving and presents opportunities for the development of new and innovative services, which means that the DevOps approach is ideally suited and offers a genuine path to deriving new business value.

Adopting the microservices approach results in services that are modularized and small in nature. Given their size, each individual service is easier to develop, test and maintain. The services may also be developed and deployed in parallel. This facilitates continuous integration (CI) and delivery, which are key principles of the DevOps approach. DevOps teams are also well placed to take the individual pieces of functionality offered through microservices and use those as the building blocks for larger applications and systems. Those systems can then be easily expanded by adding new microservices without unnecessarily affecting other parts of the application, thereby offering flexibility and achieving scalability. The consequence is that software development organizations already employing DevOps practices to develop microservices will already be well placed to embrace the MEC software development paradigm, where an overall MEC solution will comprise of multiple software components (i.e. microservices) each providing different capabilities and functions, potentially from different providers, and where those components are expected to continue to expand and evolve thereby benefiting from DevOps CI and CD processes. This expansion will happen at a high and low level, for instance at a high level as the MEC platform's overall capabilities are enhanced to support multiple access technologies and full application mobility and at a lower level as specific applications are further developed to exploit enhanced API functionality, for instance as the Radio Network Information Service capabilities expand into the multi-access domain.

An activity closely related to implementing the DevOps approach is the efforts underway in the MEC ISG to define a test framework to cover aspects such as conformance and interoperability. The framework will deliver a test suite that is ideal for automated testing. This is considered an integral DevOps component, since automated testing supports the delivery pipeline in creating processes that are iterative, frequent, repeatable, and reliable.

Other Issues

Security

With both economic and political cyber-threats on the rise and sophistication of attacks increasing, security considerations should be front and centre in designing a MEC application. In many ways, MEC-enabled applications are similar to other modern applications in terms of security. Approaches, based on modern concepts, such as “zero-trust networking” should be used. In particular, as we noted, a microservices-based design approach (when implemented well) lends itself to isolation of vulnerabilities.

However, MEC can also present unique security challenges as well as opportunities when it comes to the specifics of edge computing. To illustrate this, we provide some examples:

- Challenges:
    - Each MEC “mini-cloud” has limited compute capabilities and these are likely to be costlier. Thus, MEC mini-clouds may be more susceptible to DoS attacks than more central clouds.
    - Physical security must be a real consideration with edge computing even for application developers. An application developer is not likely to worry about what happens if a malicious party accesses the physical infrastructure of a traditional cloud provider – this is an extremely unlikely event. However, this is not necessarily true for MEC – edge compute clusters are often located in much less physically secure locations.
- Opportunities:
    - The collection of edge and cloud taken together presents a much more challenging attack surface than a centralized cloud (even an internally redundant and distributed one). As such, MEC may present an interesting solution for safeguarding critical data and compute tasks that otherwise do no need to be run at the edge.
    - While more vulnerable to a DoS attack, the limited reachability of many MEC PoPs may make it significantly more difficult to perpetrate a DDoS attack against them.

In summary, designing applications for the edge requires careful consideration of security. MEC can be used to improve the overall security of an application – but if not used properly, it can also expose the application to new or increased threats.

Mobility

The terminal is likely to be a mobile device and the current MEC host may not be the best choice for the user due to user mobility. The MEC system allows to relocate the user context from one application instance to another running in a MEC host closer to the user. Furthermore, the MEC system implements a mechanism to relocate the application instance as a whole if necessary. Several use cases for application relocation are documented in ETSI GR MEC 018 “End to End Mobility Aspects”. In these regards, programmers might design their applications with certain capabilities in order to leverage and optimize the UE relocation procedure. Such capabilities might be monitoring the application KPIs to assist the relocation decision, mechanisms to determine the right relocation timing, data synchronization, etc.

Concluding Remarks

This paper addresses the issues that application developers face when developing applications for edge computing, including MEC-based edge computing. While offering only a superficial treatment, the paper provides a guide to developers for further reading and a starting point on how to address the challenges posed by this new type of application hosting environment.

All the authors of this white paper are active in ETSI ISG MEC. As the only international standards committee focused on edge computing, ETSI ISG MEC continues to work on simplifying the application development process and enabling interoperability through the definition of a reference architecture, standardization of APIs and other activities in this space. We invite the readers to learn more by visiting our web pages, http://www.etsi.org/mec. For those interested in participating in our work, the web page also contains a link to information on how to join our group.

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

ETSI White Paper No. 20

Developing Software for Multi-Access Edge Computing

2nd edition – February 2019

ISBN No. 979-10-92620-29-0

Authors:

Dario Sabella, Vadim Sukhomlinov, Linh Trang, Sami Kekki, Pietro Paglierani, Ralf Rossbach, Xinhui Li, Yonggang Fang, Dan Druta, Fabio Giust, Luca Cominardi, Walter Featherstone, Bob Pike, Shlomi Hadad

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Dario Sabella
Intel

Vadim Sukhomlinov
Intel[1]

Linh Trang
Sony

Sami Kekki
Huawei

Pietro Paglierani
Italtel

Ralf Rossbach
Intel

Xinhui Li
VMware

Yonggang Fang
ZTE

Dan Druta
AT&T

Fabio Giust
Athonet

Luca Cominardi
UC3M

Walter Featherstone
VIAVI Solutions

Bob Pike
ACS

Shlomi Hadad
Saguna

[1] The work was done when the author was with Intel. Now he is working with Google.

Contents

Introduction

Edge Computing refers to a broad set of techniques designed to move computing and storage out of the remote cloud (public or private) and closer to the source of data. For the emerging class of "5G Applications" this is often a matter of necessity. Locating such applications in a traditional cloud does not allow one to meet certain stringent requirements, such as roundtrip latency. In other cases, such as the Internet of Things (IoT) and Vehicle to everything communication (V2X), the amount of data is expected to increase rapidly. Edge computing can mitigate this by collecting and processing data closer to the user.

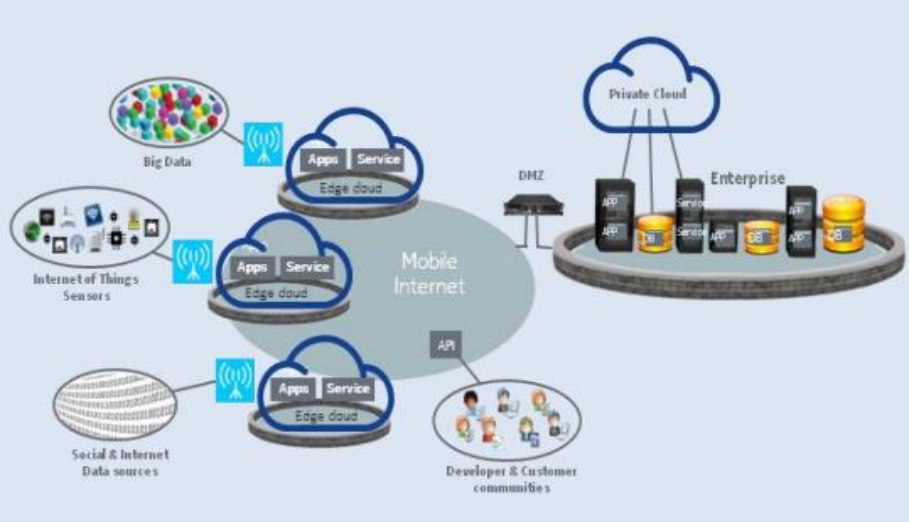


Figure 1: Overview of the MEC system (see references [1][2])

ETSI Industry Specification Group (ISG) MEC (Multi-access Edge Computing) focuses on enabling edge computing at the access network (mobile or otherwise), thus bringing edge computing as close as possible to the user without it being in the user device. The group was established in September 2014 to standardize APIs that will enable application and content providers to utilize computing capabilities present at the edge of the network. MEC enables successful deployment of new use cases and various services that can be customized according to the customer requirements and demands. Some of key applications and use cases are:

- Video content delivery optimization
- Video stream analytics and video surveillance
- Augmented Reality and Virtual Reality (AR/VR)
- Enterprise applications enablement and local breakout
- Applications with critical communication needs such as traffic safety and control, autonomous cars, Industrial IOT and Healthcare
- Connected Cars
- IoT applications and Gateway
- Location and Context aware Services
- Smart City applications

The current prevalent distributed computing software development model uses a client-side to initiate server requests and a remote server-side to process these requests (the client-server model). This allows application developers to take advantage of centralized compute and storage and has been a major driver of the emergence of cloud computing. However, for MEC applications, developers need to identify features of their applications that require processing at the edge as distinct from features that require high compute power or that do not require near real-time response and can, therefore, be deployed at a central location. Applications have to be designed in a way which supports distributed processing, synchronization of contexts, multi-level load-balancing.

This idea is quite recent, although not totally new, and the ecosystem is quickly moving to use systems like Greengrass for Amazon's AWS Lambda, Microsoft's Azure IoT stack and GE's Predix to enable it. Let's take, for example, AWS Greengrass. This consists of the AWS Greengrass core (which is responsible for providing compute capabilities closer to the devices) and the AWS IoT devices enabled with AWS IoT Software Development Kit (SDK). Using this architecture, AWS IoT applications can in real time respond to local events and use cloud capabilities for certain functions that don't require real time processing of data. An IoT application developer targeting AWS Greengrass has to architect the application in a way that uses these edge systems for certain features that require real time processing, or which perform some other useful tasks (e.g. limiting the data flood to the central location), while keeping other features in the traditional cloud.

To provide these new services and to make the most out of MEC it is also important for the application developers and content providers to understand the main characteristics of the MEC environment and the additional services which distinguish MEC from other "edge computes", namely: extreme user proximity, ultra-low latency, high bandwidth, real time access to radio network and context information and location awareness.

On this basis this white paper provides guidance for software developers on how to properly approach architecting and developing applications with components that will run in edge clouds, such as those compliant with ETSI's MEC standards. The white paper will summarize the key properties of edge clouds, as distinct from a traditional cloud point-of-presence, as well as the reasons why an application developer should choose to design specifically for these. It will then provide high-level guidance on how to approach such design, including interaction with modern software development paradigms, such as micro-services - based architectures and DevOps.

The need for an evolved approach

MEC offers to application developers and content providers cloud-computing capabilities and an IT service environment at the edge of the network. Consequently, MEC introduces a standard for supporting an emerging cloud paradigm for software development communities. In fact, up to now a “traditional” client-server model of application development has been the dominating approach to developing applications for at least 2 decades. The emergence of edge computing, e.g. MEC, evolves this environment, by introducing an intermediate element at the network edge.

A MEC point-of-presence (PoP) is distinct from a traditional cloud PoP. It may offer significant advantages to application components/services running there, while also presenting some challenges, e.g. higher cost, relatively small compute footprint, good local but not global reachability, etc. As such, it is crucial for an application developer to design with specific intent towards running some application components at the network edge when developing for MEC.

This results in a new development model with 3 “locations”: Client, Near Server, Far Server (depicted in Figure 2). The client location can be a traditional smartphone or other wireless connected compute elements in a car, smart home or industrial location that can run dedicated client applications. The model is quite new to most software developers, and while modern development paradigms (e.g. microservices) make it easier to adapt to it, a clear and concise summary of this new development model and guidance on how to properly approach it will help accelerate the application development for the network edge and thus accelerate MEC adoption.

As depicted in Figure 2, a MEC Host, usually deployed at the network edge, contains a MEC platform and the compute, storage and network resources for applications in VMs or containers. The MEC platform offers a secure environment where MEC applications may, via RESTful APIs, discover, advertise, consume and offer services.

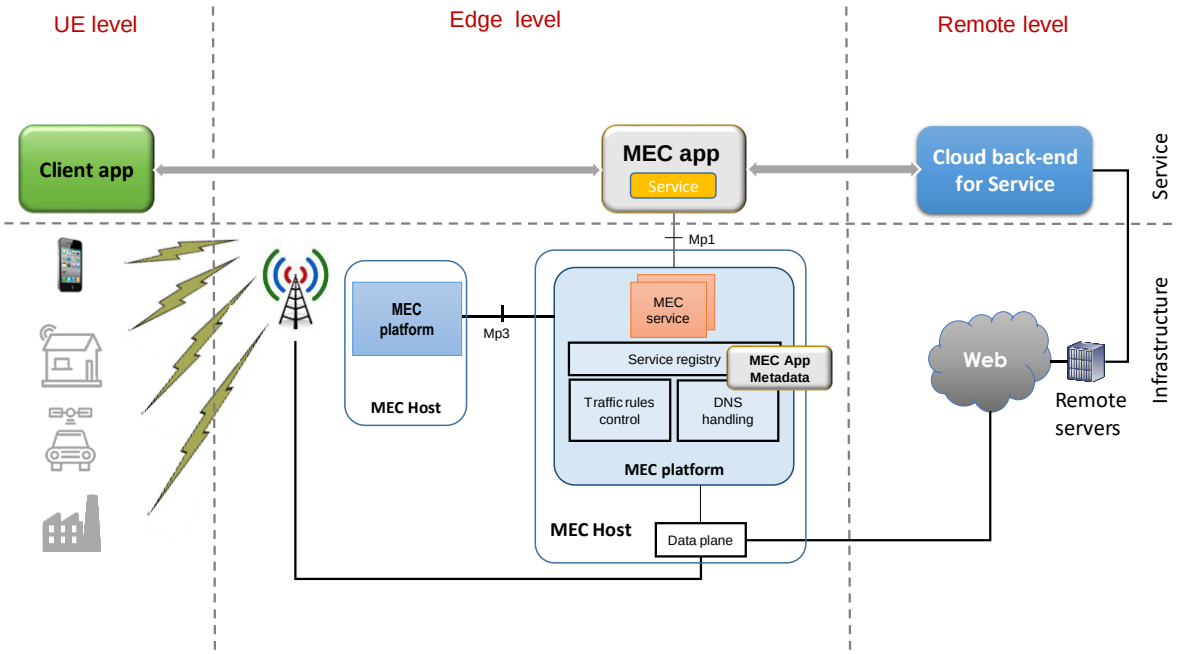


Figure 2: New application development paradigm introduced by MEC.

Designing with the Edge in Mind

A key principle of modern system design, known as the hour-glass model (and realized with IP as the single protocol at the waist of the hour-glass) is that applications and networks should be completely agnostic to each other. This has been key to the success of the Internet and the millions of applications running over it. It allows applications to run over any IP-based network, while any network can use IP to support pretty much any IP-based application. Nevertheless, when it comes to actual application performance, both aspects (the application design and the network) are important and remaining completely agnostic becomes difficult. To date, few traditional Internet applications encountered issues with this approach since the requirements of applications have traditionally been much looser than the performance that networks can deliver. This is about to change. Already practices are emerging that consider network aspects as part of application design. A widely used example is adapting the application behaviour to throughput limitations in the network, e.g. adjusting the video stream compression ratio in response to throughput throttling, while also considering the application state, e.g. whether the application is active or idle or in suspended state, etc. Nonetheless, network conditions and topology characteristics are typically considered during the software design phase as an environmental input out of the programmer's control and the application *passively* adapts to these.

The environment enabled by the MEC platform is where the network and the applications can converge in a meaningful way without giving up the key benefits of the hour-glass model. MEC can support any application and any application can run in MEC. However, MEC can offer additional services to those applications which have been designed to be MEC-aware. MEC Application Enablement (described in ETSI GS MEC 011 [3]) introduces such a service environment, and this can be used to improve the user experience tremendously. Software designed to take advantage of MEC services can leverage additional information about where the application is supposed to run, in terms of expected latency, throughput and other available MEC services. Simply put, with MEC, the environment becomes *less unpredictable* and environmental (i.e. contextual) information can be leveraged to *actively* adjust the application behaviour in run time. This means that the network characteristics can factor in during the design of the end to end service. For example, through the MEC Radio Network Information (RNI) API, it is possible to precisely monitor the radio link, and this information can be obtained by a MEC application that uses it e.g., to drive the behaviour of the client application in the user device, as well as of the application in a central cloud. Similarly, a MEC application can make a bandwidth request using the Bandwidth Management API to reserve networking resources in the MEC system (more details about MEC APIs can be found later in this paper. This allows edge applications to benefit from low latency and high throughput in a predictable/controllable way, and this information can be leveraged at the time of service design to optimize the end-to-end service architecture. In addition, the network itself could also benefit from the MEC services provided by the applications, for instance the network scheduler could also predict the incoming user behaviour to maximize the network efficiency.

From the discussion so far, it emerged already a key aspect of software design for MEC: the end-to-end service can be split into three applications or components: *terminal device component(s)*, *edge component(s)* and *remote component(s)*. This concept should not be confused with the traditional software modularization, but rather seen as a distribution of components to leverage different features of the computing environment. In fact, if the former considers the division of tasks to improve the development and maintenance of the code, the latter is based on distributed computing to meet specific performance figures achievable only at the network edge. In many current services, the different components either run in the same data centre or are sparsely distributed at unpredictable locations (e.g.

P2P applications), and software instances are scattered mostly to address load balancing. Software modularization is not only possible in MEC, but also encouraged to assign execution tasks at the most appropriate location. In particular, as discussed below, a microservices-based architectural approach is particularly well suited for MEC.

This aspect creates an additional paradigm with respect to a more traditional client-server architecture, since an additional processing stage (at the edge) must be added to the application's workflow, with well-defined characteristics and capabilities. For instance, the terminal device can do some preliminary processing to determine the need for further actions. Such preliminary processing requires near zero latency and it requires the terminal device to support some computing capabilities, e.g. to receive and instantiate algorithms or instructions. The edge component(s) include a set of operations that the application performs at the edge cloud, e.g. to offload the computing away from the terminal device while still leveraging very low latency and predictable performance, or offloading high bandwidth load from the network backbone, or extracting some information using RNI API or Location API. The remote components implement operations to be carried out in the remote data centre, e.g. to benefit from large storage and database access.

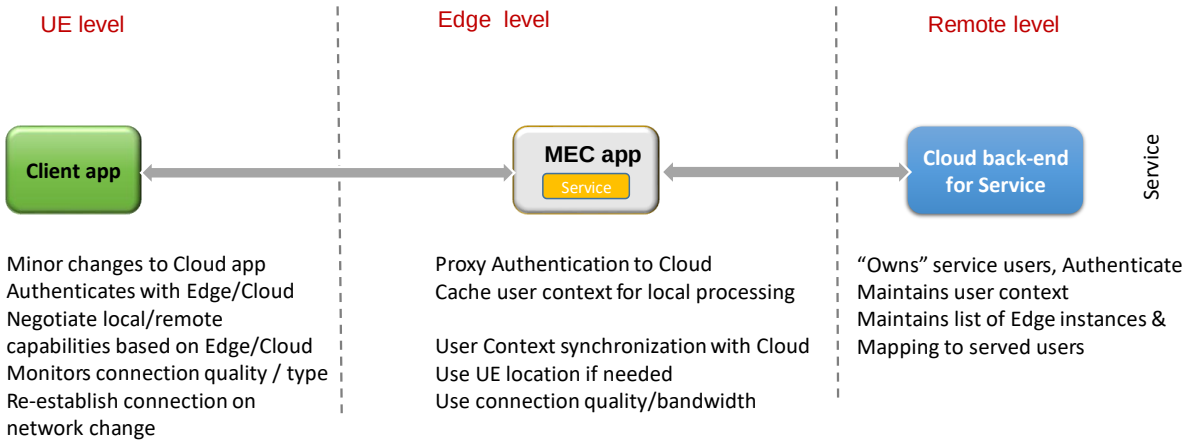


Figure 3: Example of splitting an application into "terminal", "edge" and "remote" components

What is worth highlighting here is that MEC can be exploited to implement computation offloading techniques among all the application's components. In fact, the server can be programmed to dynamically shift the processing among the terminal, the edge and the remote component(s), for a number of reasons ranging from adaptation to network conditions, improving application specific KPIs, policies, costs, etc. The processing distribution may be driven by certain performance objectives, e.g. providing the best user experience.

Another aspect inherent to MEC is that service providers can exploit the geographical distribution to serve different user populations and thus tailor their service knowing the peculiarities of the covered area. For instance, media content can be adapted to the linguistic and cultural characteristics of a given area, or customized advertising can be tailored to the needs of local businesses. Content Delivery Networks (CDNs) are only one, but perhaps the most straightforward example of a use case that benefits from geospatial distribution. Nevertheless, most of content provided by today's CDNs is not as delay sensitive as the services from other MEC applications, e.g. for virtual/augmented reality, for which it is crucial to run at a specific location to meet the application's stringent latency requirements.

Architecting and Developing for the Edge

This section describes how to deploy an application in a MEC system. Some features are fully defined by the MEC standard, whereas others are left to the specific implementation of the MEC provider.

As a first step there are two APIs that the developer should consider, specifically Mx2 API (see ETSI GS MEC 016 [4]) and Mp1 API (see ETSI GS MEC 011 [3]). Mx2 allows a device application to interact with the MEC system and is further described later in this paper. Mp1 is the reference point between MEC applications and the MEC platform, which allows these applications to interact with the MEC system by discovering, advertising, consuming and offering MEC services. In addition, the application may influence the traffic routing by updating the MEC traffic rules. There are also specific service-related APIs such as Radio Network Information API (ETSI GS MEC 012 [9]) and Location API (ETSI GS MEC 013 [10]). Depending on whether the application wishes to consume MEC services, or even produce them, the developer may also want to consider these service-related APIs.

To facilitate MEC application design, MEC communications can be divided in phases (described in detail in the following sub-sections):

- Phase 1 – MEC application packaging & on-boarding
- Phase 2 – MEC application instantiation
- Phase 3 – communication between client-side app and MEC app
- Phase 4 – usage of the MEC platform and services

These phases are described in detail in the following sub-sections. Before analysing them in detail some preliminary high-level considerations for when developing and deploying an application in MEC are provided:

- **DNS-based solution**

  The application must be designed to support a DNS-based solution for traffic redirection. The application sends a DNS query for a registered domain name.

- **Domain name**

  Register a name for the service (FQDN), which is known to client application to gain access.

- **Cloud back-end**

  If there is a requirement for the service to be available regardless of whether a local MEC system exists or not, then a back-end service will be required as a fall-back solution hosted in the cloud or at alternative premises.

- **Sensitive user context data**

  User context data may be sensitive from a legal point of view and may not be allowed to be transferred from one jurisdiction to another. This use context transfer is something the developer must handle in their application if application mobility is to be supported.

- **Packaging the application**

A MEC application runs as a virtualized application, such as a virtual machine (VM) or a containerized application, on top of the virtualization infrastructure provided by the MEC host. Appropriate packaging of the application is inextricably linked to the run-time environment of the MEC system and therefore the developer needs to adapt accordingly, providing the package e.g. as VMs, Docker containers or Kubernetes templates. This is further described in the next section

- **Provide meta-data with application requirements**

The application needs to be provided with its requirements such as latency tolerance, network resources, storage resources, CPU etc. that the MEC system needs to account for. These requirements are provided in the Application descriptor (see ETSI GS MEC 010-2 [5]), which forms part of the application package. Information on whether the application produces a service or is dependent on other services must also be included.

Phase 1: MEC application packaging & on-boarding

MEC applications are packaged by application developers (or in some cases also by MEC operators), and typically set up as a VM or Container (e.g. qcow2/ vmdk images, Docker containers, etc.) with all the necessary dependencies according to a specific MEC platform's requirements and configurations. For security reasons application providers usually sign their application package before sending it to the OSS for set up purposes. Various options exist to package and onboard applications within a MEC environment which likely will have contractual considerations between the interested parties. For this reason, the specific details of application **onboarding** are not defined by the standard and are rather left within the platform and service provider domain. However, the high-level details and steps are provided below.

Figure 4 highlights the entities involved in application onboarding. When the OSS receives requests for managing of applications (e.g. onboarding, instantiation or termination) it makes the decision on whether to grant these requests, or not. Granted requests are forwarded to the MEC Orchestrator (MEO) for further processing. After receiving a request from the OSS, the MEO has the responsibility of onboarding MEC applications into MEC systems, including checking the integrity and authenticity of the signed packages, validating application rules and requirements and if necessary, adjusting them to comply with operator policies, keeping a record of on-boarded packages, and preparing the virtualization infrastructure manager(s) to handle the applications.

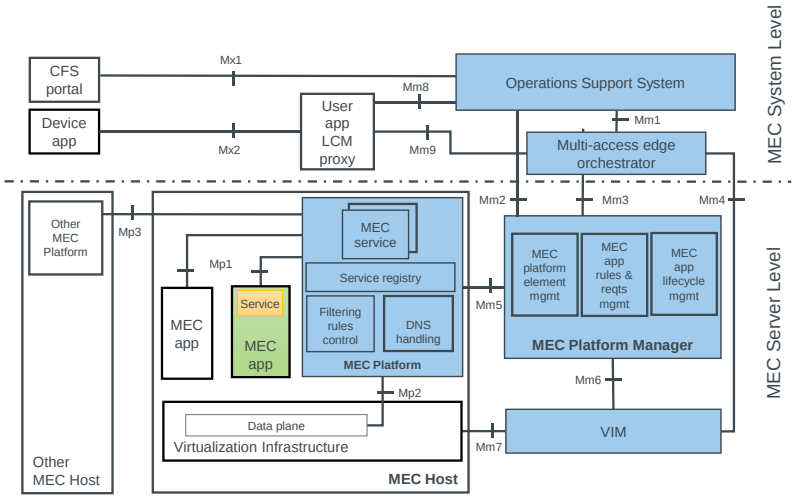


Figure 4: Entities involved in Application on-boarding, instantiation and operation

The MEO assigns an application package ID and provides the MEC Platform Manager (MEPM) with the location of the application image if it is not yet on-boarded in the MEC system. The MEPM prepares the Virtualized Infrastructure Managers (VIMs), selected by the MEO for application instantiation, by providing the necessary infrastructure configuration information and sending the application images, which are then stored by the VIM.

Once on-boarded, the application package is in "Enabled, Not in use" state. From there further application lifecycle management actions may be performed by the OSS in response to application package enable, disable, query and deletion commands.

Phase 2: MEC app instance instantiation and operation

In this section we describe how the developer can get the MEC application instantiated in the MEC system and then make it available to the target audience (the final customers, or in general the app users).

Application initialization may be triggered from a device or from the Operation Support System (OSS). With the first option (figure 4 below), a developer is able to interact directly with the MEC system using a device with a client supporting the Mx2 API (see ETSI GS MEC 016 [4]). In MEC parlance such a client is referred to as a device application. This also requires the MEC system to support the MEC standards defined *UserApps* feature. The User Application LifeCycle Management Proxy (UALCMP) exposes the Mx2 API to the device application (see Figure 4). It allows the device application to request the following application lifecycle management operations from the MEC system: query the available applications, instantiation and deletion of an application and update of an existing application context.

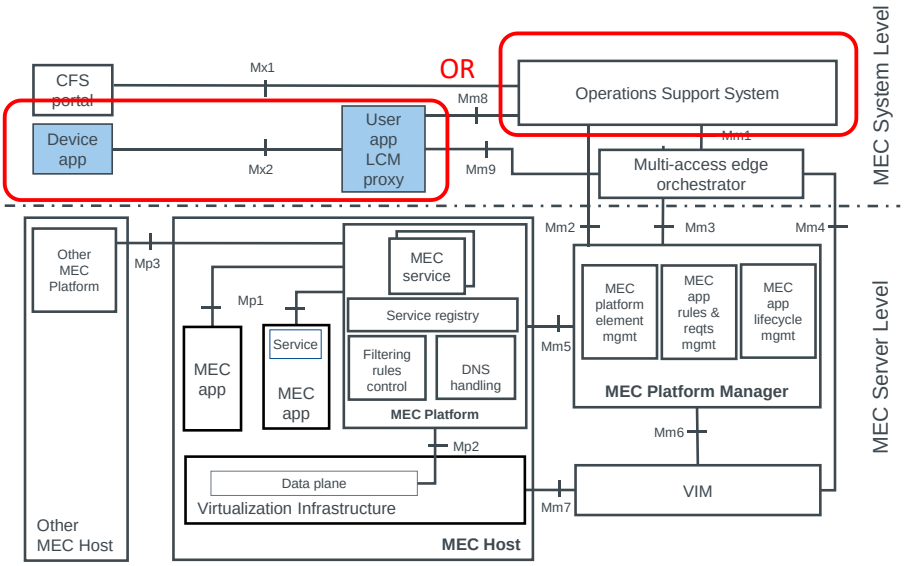


Figure 5: MEC app instantiation options

The other option for the application instantiation is via direct interaction between the developer/service provider and the MEC operator without the involvement of a device application. The MEC operator then triggers the application on-boarding and instantiation directly through the OSS. With this approach the application developer does not have to concern themselves with this initial phase.

The following two subsections describe the instantiation process from a client (external) perspective and give a brief overview of how this is performed from a MEC platform (internal) perspective.

The client perspective

In real life scenarios, for direct interaction with the MEC system, the device application needs a subscription or some other contract with the MEC operator[2]. The subscription authorizes the device application to make use of the MEC services. The ETSI MEC specification relies on OAuth 2.0 in securing the device application's access to MEC system resources. The device application first authenticates and then gets authorized in operator's AA server, which will provide the necessary credential for Mx2 API operations, i.e. the access token. Once it has those and has discovered the UALCMP it can then invoke Mx2 API. In each API call the device application is expected to include the access token in the "Authorization" header field as a bearer token as defined in IETF RFC 6750 [6]. The exact methodology for the device application to acquire the token is beyond the scope of the MEC specification. It is also noteworthy that any given MEC operator may support other means of authentication and authorization, which are also beyond the scope of MEC standard.

With the Mx2 API there are different application instantiation options. If the application image is already available in the MEC system, the developer can request it to be instantiated directly. The Mx2 API also supports querying of all available applications and then pick the one to be instantiated. If, however, the application image is not yet in the MEC system, the developer can provide the MEC system with a link to the application package that contains the image and the descriptor. Using this link, the system can retrieve and then instantiate it. In this latter option, if at all supported by the MEC operator, the image repository most likely will have to be trusted by the operator, or the operator may even have a repository of its own where the developers can make their application images available. In both options the MEC system's response to a successful instantiation includes the address of the instantiated application. The attribute *ReferenceURI* in the Application Context in the response body from the UALCMP conveys the address of the application instance. Once the device application has received the address for the MEC application, the developer can disseminate it among the target audience through their own chosen means. During the lifetime of the application instance the device application will receive notifications of the change, if any, of the application address.

According to the ETSI MEC terminology and specifications, a MEC application that was instantiated in the MEC system in response to a request of a user over Mx2 API is called a user application. In addition to instantiation, the Mx2 API also supports the deletion of the user application. The developer can request the application context to be deleted, resulting in a termination of the application and removal of the application's resources in the MEC system for application context in question.

At this point it is worth emphasizing a couple of essential aspects on the application instantiation in MEC:

1. All operations between the authorized device application and the system proxy (UALCMP) on Mx2 API are management plane operations. There the application instance is *referred to* via the identifier of the context that was created by the MEC system for this application instance. Data type AppContext in ETSI GS MEC 016 [4] represents the information on application context, and there the attribute *contextId* is the identifier of the said application context. Each application instance present in the MEC system has their own unique context ID on Mx2.
2. On the user plane any user with a connected device can attempt to contact the application by using the application's address (the one originally found in the ReferenceURI, e.g. an IP address) as is the case with any internet application. As the MEC system can also configure the DNS server, the application instance may also be addressed by its FQDN (Fully Qualified Domain Name), if

[2] The MEC operator may be different from the Mobile Network Operator (MNO).

available. Any security, authentication, authorization, etc. related to accessing the instantiated application on the user plane is beyond the scope of the MEC specification and will have to be provided by the application/service provider.

3. The developers requesting an application to be instantiated cannot select the location where the application is instantiated, but they shall describe the requirements of the application. These requirement attributes are defined in the Application descriptor data type in ETSI MEC GS 010-2 [5] and they include the C/S/N requirements but also the maximum latency tolerated by the application. It is then the MEC system Management & Orchestration (MANO) that selects the ideal location for the application instance. For an application image made available in a repository for the MEC system to fetch it, the application descriptor is expected to be included in the application package.

The MEC platform perspective

As explained earlier, the instantiation of a MEC application on a MEC host is initiated by the OSS, e.g. in response to a request from the device application, or via direct interaction between the service provider and MEC operator. The OSS forwards the granted instantiation to the MEO, which is responsible for oversight of available MEC apps/services, and app termination. The MEO selects the target MEC platform (MEP) and forwards the request to selected MEPM, which configures the MEP accordingly.

An instantiation request contains information about the application to run, application rules and requirements and possibly other information, such as the target location where the application is to be deployed. MEPM can send lifecycle requests to the VIMs for allocating virtualized resources (compute, storage and networking) and for instantiating the MEC application. Once set up, the MEC application is enabled to interact with MEP over Mp1 to perform certain support procedures related to the lifecycle of the application, such as indicating availability, preparing relocation of user state, etc. MEPM also receives virtualized resources fault reports and performance measurements from VIM for further processing and updates the initial set-up information accordingly.

After an application has been instantiated, the following operations can be performed on the application instance through the application lifecycle management APIs (see ETSI GS MEC 010-2 [5]):

- Start: instruct the application instance to run and produce the service
- Stop: instruct the application instance to stop running and producing service

When an operation on the application instance is invoked, the application lifecycle management will trigger a special task, i.e. Application lifecycle Operation Occurrence, to track the operation and send a notification to the application lifecycle management once the operation finishes. The application lifecycle management then updates the operational state of the application instance accordingly.

The MEC platform and MEC applications or services can subscribe to the notification on application instance operational state change through the application lifecycle management APIs.

Phase 3: client-side app and MEC app communication

A device application is logically separate from the actual client application (see Figure 6) which is the one requesting services from the MEC application. Any end user device may have the client application installed. A client application most often is unaware of the edge deployment of the server application, i.e. the MEC application. The device application, on the other hand, is needed for invoking user application lifecycle management operations on MEC system's management plane, as explained in phase 3.

There are two ways for the client application to connect with a MEC application instance: The first option is that the developer/service provider takes responsibility for making the address of the MEC application, which it receives from the MEC system, available to potential client applications. The other option is that the client application discovers the MEC application instance via DNS look-up. If the domain name is owned by the application provider/developer, the developer needs to ensure that the domain name is managed and updated in the corresponding DNS server. The MEC platform supports the handling of MEC applications' DNS records accordingly.

Depending on what functionalities are supported in the MEC system(s) where the application is being deployed the developer needs to support at least one of the two options above.

Figure 6 depicts the client-side application components; client application(s) and the device application, as well as the relevant MEC system components and the ways how the client-side application components can communicate with their MEC side peers. MEC platform has been authorized to configure MEC application's DNS records for a successful address resolution.

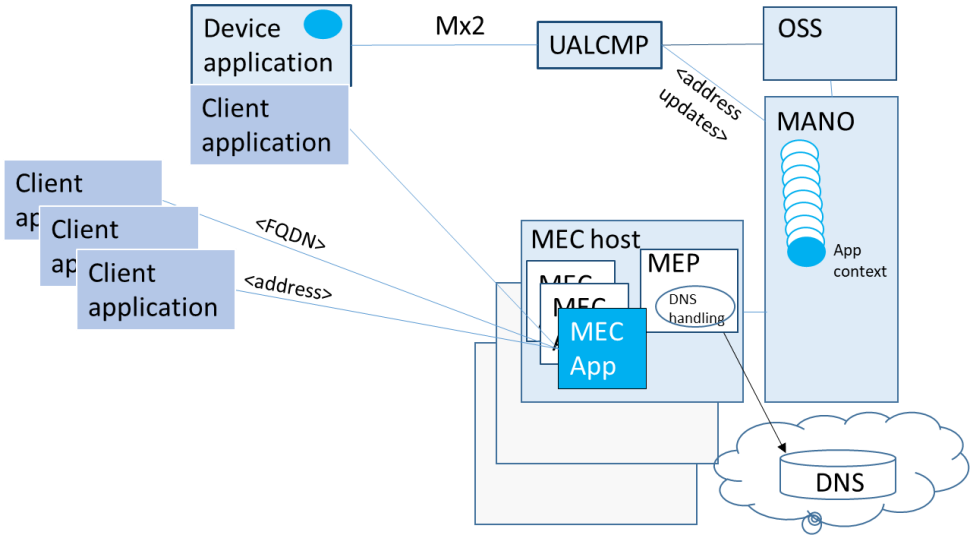


Figure 6: Communications between terminals and the MEC system

Phase 4: usage of the MEC platform and services

Once the MEC application is up and running and becomes operational, it can consume MEC services. These services may be produced by the MEC platform or be produced by another MEC application. MEC services (e.g. RNI, Location, Bandwidth and UE Identity) are available for the MEC platform and authorized MEC applications.

RESTful design

MEC specific APIs are built upon RESTful APIs. The concept of RESTful programming is widely accepted in the industry and many developers are already familiar with the principles of these APIs. A RESTful API is an application program interface (API) that uses the HTTP protocol as a tunnel or transfer mechanism for interaction between remote entities. RESTful refers to a stateless design through REpresentational State Transfer (REST), an architectural style popular for development of web services ([7]).

ETSI's design principles for developing RESTful MEC APIs are outlined in ETSI GS MEC 009 [11], along with http methods, templates, conventions and patterns to create RESTful APIs for MEC. MEC APIs are not only documented in ETSI specifications, they are also available in YAML and JSON format in a Git repository at https://forge.etsi.org/, presented via OpenAPI compliant descriptions.

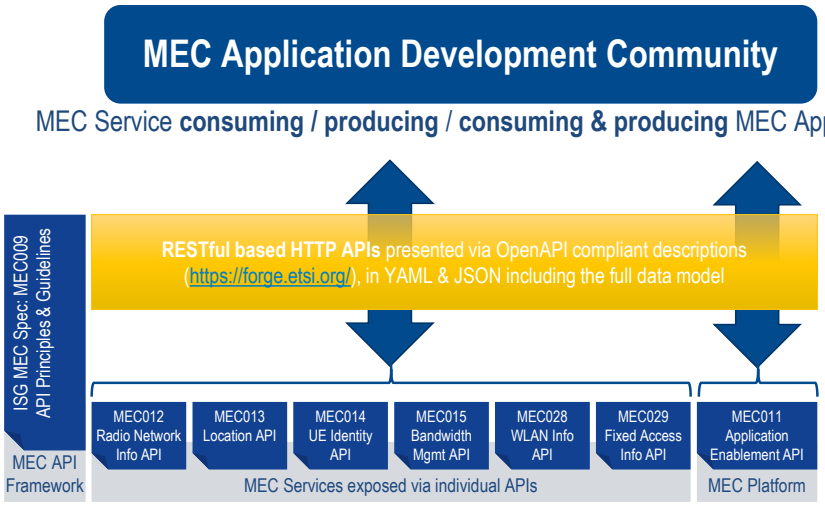


Figure 7: Phase 5 - overview of the MEC Platform and Services from a developer perspective

Radio Network Information API

MEC applications run at the edge of the network where the environment is characterized by low latency, proximity, high bandwidth and exposure to location and up-to-date radio network information. While low-level information on current radio conditions is available in the Radio Access Network (RAN), the MEC server is in relative proximity and connected to the base station. The Radio Network Information Service (RNIS) is a service that provides radio network related information to authorized MEC applications and the MEC platform. The provided information originates from the RAN, based on contents defined in 3GPP specifications. The granularity of Radio Network Information desired by the application can be configured per cell, per UE, per QCI/5QI class, or may be requested over a period of time only. Some of the typical information that can be exposed is listed below:

- Up-to-date information about radio network conditions.
- Layer 2 measurement and statistics information based on KPIs of the user plane, according to 3GPP specifications.
- Information about UEs connected to the radio nodes associated with the MEC host, their UE context, related radio bearers, Quality of Service (QoS) information, throughput, neighbour cells, etc.
- Changes on the information above, based on API subscription, for example to provide notifications on cell change, radio bearer release or reconfigurations, updates to carrier aggregation configuration, UE measurement reports, related to UEs connected to the radio nodes associated with the MEC host.

The Radio Network Information (RNI) may be used by MEC applications and/or the MEC platform to optimize the existing services and to provide new services that are based on up-to-date information on radio conditions. For example, RNI can be used to include bitrate recommendations for video or voice calls based on actual real-time throughput available for a specific connection, or the MEC platform can utilize RNI to optimize mobility procedures required to support service continuity. RNI is also used to optimize network operation. The RNIS supports a wide range of use-cases, some of which are described in ETSI GS MEC 002 [12].

The service consumers interact with the RNIS over the RNI API to obtain contextual information from the radio access network. The Radio Network Information API supports both queries and subscriptions (pub/sub mechanism) that are used over the RESTful API or over the message broker of the MEC platform. More information about the RNIS as well as the APIs is available in ETSI GS MEC 012 [9].

Location API

The availability of accurate location information is crucial to a number of services and enables area-specific application data content. MEC's Location Service (LS) can provide location-related information to a MEC platform or authorized applications.

The LS leverages the Zonal Presence Service described by Small Cell Forum in [16] and [17]. The Location Service is accessible through RESTful APIs originally defined by Open Mobile Alliance (OMA). The incorporated API definitions as well as the full description of the location service are available in ETSI GS MEC 013 [10]. The Location Service is registered and discovered over the Mp1 reference point defined in ETSI GS MEC 003 [13].

Consumers of the Location Service may use the LS API to obtain the location information of a UE, a group of UEs, or the radio nodes associated with a MEC host. For example, the service can perform active device location tracking or provide location-based service recommendations to allow a variety of additional services tightly coupled with a specific place (such as a shopping mall). It is possible to report information such as the distance between a specified UEs or the distance between a specified location and a UE, provide a list of UEs in a particular area of location, or even report when specific UEs move in or out of a particular area.

The service supports both geolocation, such as geographical coordinates, and logical location, such as a Cell ID. Subscriptions to location information are also offered, including periodic location information updates, updates on changes in distance and location updates relating to UEs in a particular area of location, and more. The Location Service API supports both queries and subscriptions (pub/sub mechanism). To facilitate collection of statistics, anonymous location reporting (without related UE ID information) can be used. Operators or third-party services can use the location data for security, safety, and data analytics, or to optimize the network.

Bandwidth Manager API

When a MEC host runs applications in parallel and is using the same network resources, applications are typically competing over available bandwidth. Data traffic associated with different MEC applications (or even specific application sessions) may have different bandwidth requirements with respect to e.g. throughput and priority. A bandwidth management service (BWMS), which likely is produced by the MEC platform, allows a fair distribution of bandwidth resources between applications.

Using the BWMS, different applications, whether managing a single instance or several sessions, may request specific bandwidth requirements for the whole application instance or different bandwidth requirements per session. The BWMS aggregates all the requests to help optimize bandwidth usage and allocates it accordingly.

The Bandwidth Management (BWM) API, described in ETSI GS MEC 015 [14], enables registered MEC applications to request a specific bandwidth allocation. Applications can call the BWM API to register, update or unregister their specific bandwidth requirements (size/priority). It is also possible for the application to query the API to get their configured bandwidth allocation. A specific MEC application or application session is identified using a set of filters within the resource request of the API. The API uses a RESTful API design. Furthermore, the BWMS design might interface with the Radio Network Information Service to use available Layer 2 and QoS information to facilitate the traffic arbitration.

UE Identity API

The MEC platform provides functionality to facilitate the association of IP traffic flows with a particular UE using an externally defined tag rather than the UE Identity directly. The association between a tag and the UE Identity helps preserve user privacy protect identity information at both mobile and enterprise network. A MEC application manages related access control and the integrity of the user content. This is where the UE Identity API, described in ETSI GS MEC 014 [15], comes in.

The UE Identity API provides functionality for a MEC application to register/deregister a tag (representing a UE) or a list of tags in the MEC platform. Each tag has been mapped to a specific UE in the mobile network operator's system and the MEC platform is provided with the mapping information. The MEC platform uses registered tags to apply traffic filters rules based on that tag. This enables authorized UEs to get their user plane traffic routed directly to e.g. a local enterprise network without having to pass through the MEC application. The tag-based traffic filter rules are handled in the MEC platform and described in ETSI GS MEC 011 [3].

Building your first MEC application

The preceding sections have provided a comprehensive description of the key considerations a developer should consider when creating a MEC application. Annex C also contains a couple of exemplary use cases for MEC applications. This section provides some more practical insights on developing a "*Hello World*" style MEC application.

In order to increase the accessibility of the group's specifications, ISG MEC has published OpenAPI Specification (OAS) (https://github.com/OAI/OpenAPI-Specification) compliant descriptions of its service and Platform Application Enablement (Mp1) API specifications on the ETSI hosted Forge site (https://forge.etsi.org/), Figure 8. The OAS offers an open source framework for defining and creating RESTful APIs. YAML (or JSON) interface files compliant to the OAS are both human and machine-readable providing the means to describe, produce, consume and visualize RESTful web services. Based on the interface file clients, i.e. potential MEC applications, can understand and consume services without knowledge of server implementation or access to the server code by reading the declarative resource descriptions. A large ecosystem of tooling has been built up around the OAS, where the Swagger Editor (https://editor.swagger.io/) is of significant note. This enables server and client stubs, in a large variety of languages, to be automatically generated for each of the MEC APIs and used as the starting point in the creation of a MEC application.

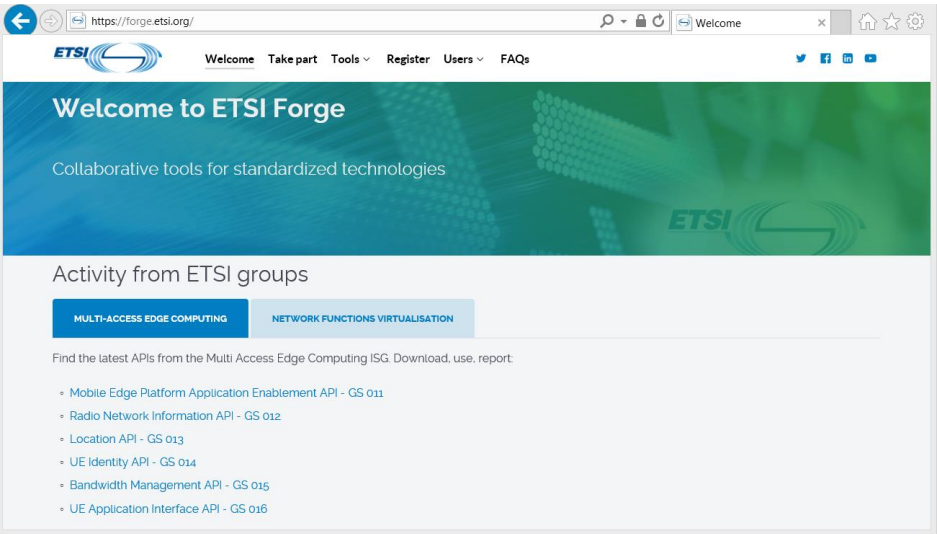


Figure 8: The ETSI Forge landing page

The MEC Location API can be taken as a specific example, where the latest version can be viewed within the Swagger Editor hosted on the Forge site (see Figure 9), or by pasting the YAML directly into the online Swagger Editor. As seen in the figure, this offers the option to "Generate Server" where several languages can be selected. It is likely that Cross-Origin Resource Sharing (CORS) support will need to be enabled for the server to allow resources to be requested from the client domain that will be outside the server domain from which the resource representations are served. By default, a server created using the Location API description will be listening on port 8081 of your localhost once running (reference the "host:" line in the figure). Further capability can then be added to the server. For instance, using it to interface to a database containing additional example data for the responses, rather than just using the example response content provided in the interface file.

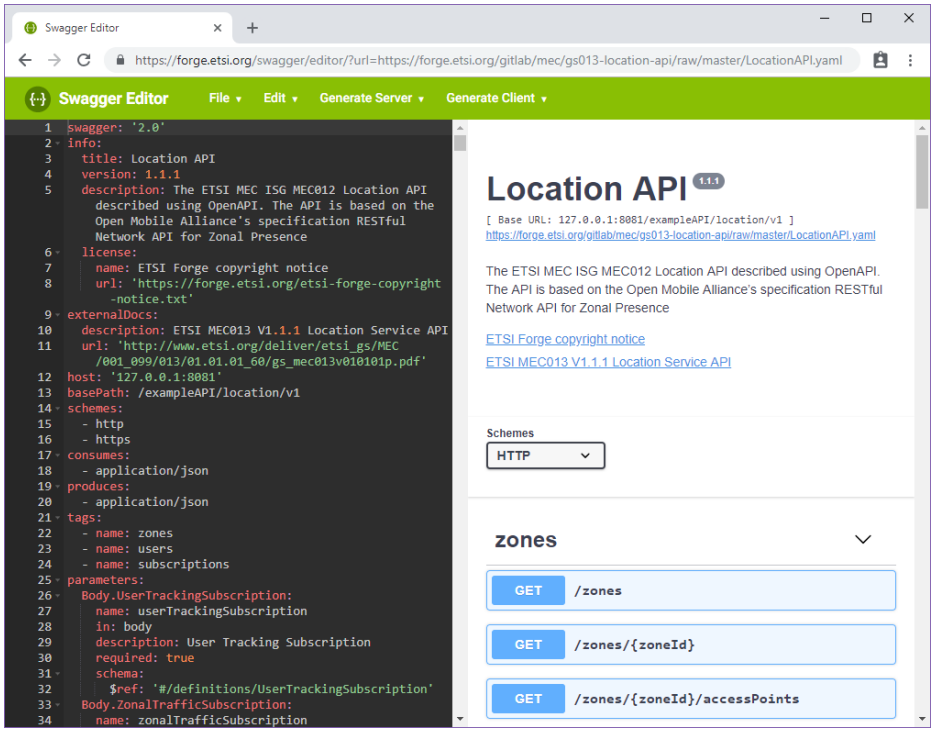


Figure 9: MEC Location API presented using the Swagger Editor

To interact with the server from a client perspective (service consuming MEC application) third party tools such as postman (https://www.getpostman.com/) can be used as an initial step to test out the offered requests and see example responses, see Figure 10. Such tools are rather more user friendly than the Client URL (cURL) command-line tool for transferring data using URL syntax. However, the Swagger Editor does provide the cURL syntax for querying each request endpoint of the API. This is seen, for instance, by clicking GET /zones (Figure 9) and then clicking the "Try it out" button that will appear.

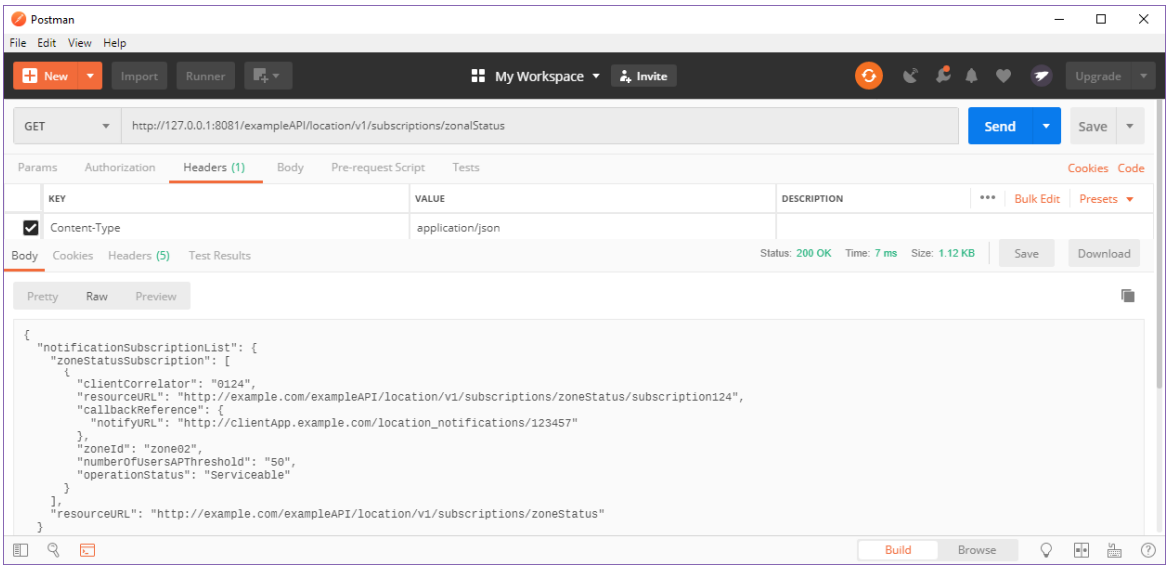


Figure 10: Example MEC Location API query and response

Third party libraries have also been developed to interact with interface files directly, thereby removing the server dependency. For example, Swagger-JS (https://github.com/swagger-api/swagger-js), which just requires a URI link to the interface file. The interface file maybe hosted by the local server, but it is just the file that is required and therefore the version hosted on Forge is equally applicable (e.g. https://forge.etsi.org/gitlab/mec/gs013-location-api/raw/master/LocationAPI.json in the case of the MEC Location API). The JavaScript file can, for example, be included in a HTML file acting as a MEC application:

```
<script src='swagger-client.js' type='text/javascript'></script>
```

This then offers the capability for the various API methods to be called from with the HTML file, where any API requests are validated against the interface file and the associated responses are populated with the example data from that file. This approach has been used in generating the plots in Figure 11 using the Leaflet JavaScript library (https://leafletjs.com/) for the interactive maps.

Finally, in order to generate a client that interacts with a server, "Generate Client" can be selected from the Swagger Editor (Figure 9). As with server generation, many different language options are supported. Once developed, MEC Applications can be packaged, on-boarded and then instantiated according to the phases detailed in another section. This is in the knowledge that the application will be compliant to the MEC API specifications, since its development was based on the client stub produced from those specifications through the OpenAPI compliant interface description file.

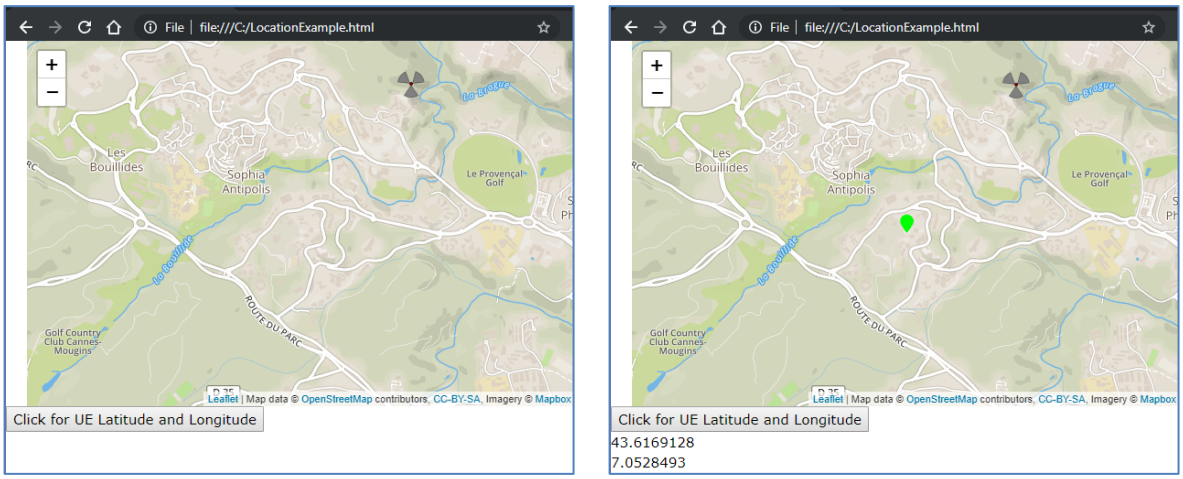


Figure 11: Plotting user location based on calling the GET users by ID via the MEC Location API

Other aspects relevant for developers

Implementation aspects

Different level of Hardware and Software Security mitigations/measures are recommended. For example, for hardware, the environment could use a Hardware Security Module (HSM) for certificates. A Trusted Platform Module (TPM) and trusted boot could also be used. For software, communication internally and externally could be secured with Transport Layer Security (TLS). In most cases, only authorized traffic is allowed in the MEC.

The MEC also must prevent illegal access from dishonest terminals and dishonest MEC application developers. The MEC system must secure the environment for running services for the following actors:

- User
- Network operator
- Third-party application provider
- Application developer
- Content provider
- Platform vendor

The Standard 4G/5G packet core protocols protect from unauthorized users joining the network. Access Control Lists (ACLs) can control policies that are leveraged by sessions and applications running at the Edge. MEC services are configurable for DeMilitarized Zone (DMZ), OAM management, and LTE network.

Security

With both economic and political cyber-threats on the rise and sophistication of attacks increasing, security considerations should be front and centre in designing a MEC application. In many ways, MEC-enabled applications are like other modern applications in terms of security. Approaches, based on modern concepts, such as “zero-trust networking” should be used. As we noted, a microservices-based design approach (when implemented well) lends itself to isolation of vulnerabilities.

However, MEC can also present unique security challenges as well as opportunities when it comes to the specifics of edge computing. To illustrate this, we provide some examples:

- Challenges:
    - Each MEC “mini-cloud” has limited compute capabilities and these are likely to be costlier. Thus, MEC mini-clouds may be more susceptible to DoS (Denial of Service) attacks than more central clouds.
    - Physical security must be a real consideration with edge computing even for application developers. An application developer is not likely to worry about what happens if a malicious party accesses the physical infrastructure of a traditional cloud provider – this is an extremely unlikely event. However, this is not necessarily true for MEC – edge compute clusters are often located in much less physically secure locations.
    - New security attack surface due to more entry points into application, more complicated certificate management.

- MEC Services may contain sensitive information. Therefore, every request for information within the MEC host must be authenticated and authorized.
- As different MEC application vendors may be installed side by side on the same MEC infrastructure, data separation policies must be taken in account carefully.

- Opportunities:
  - The collection of edge and cloud taken together presents a much more challenging attack surface than a centralized cloud (even an internally redundant and distributed one). As such, MEC may present an interesting solution for safeguarding critical data and compute tasks that otherwise do no need to be run at the edge.
  - While more vulnerable to a DoS attack, the limited reachability of many MEC PoPs may make it significantly more difficult to perpetrate a DDoS attack against them.
  - Knowledge of subscriber by network, location awareness can be used to strengthen authentication process, introduce new types of heuristics in fraud detection, availability of Edge-focused application firewalls
  - Since the MEC host location is much closer to the access network than the core network, every network blocking resulting from attack will have a much smaller impact on the overall network. Using MEC, wide scale server connectivity shutdown will be avoided.

In summary, designing applications for the edge requires careful consideration of security. MEC can be used to improve the overall security of an application – but if not used properly, it can also expose the application to new or increased threats.

Mobility

The terminal is likely to be a mobile device and the current MEC host of the user session may not be the best choice due to a change in the device's location. The MEC system allows the relocation of the user context for a session from one application instance to another running in a MEC host closer to the user. This facilitates service continuity and offers programmers the opportunity to design their applications with the capability to leverage and optimize the user context relocation procedure. Such capabilities include monitoring the application KPIs to assist the relocation decision, mechanisms to determine the right relocation timing and data synchronization. Details of what constitutes user context are provided later in this section.

Application mobility may be triggered by a change in the device bearer path in the underlying transport network. For example, device handover from one 4G or 5G cell to another associated with a different MEC host. This situation requires the MEC system to make the application instance available in this new MEC host to ensure the optimum lowest latency service to this device. If a user context existed in the source application instance, it must be transferred to the new target application instance.

Support by the MEC system for application mobility is aimed at optimizing the performance of applications and reducing service latency by relocating user sessions to the application instance closest to the end user. This facilitated through the MEC application mobility service API. This service allows applications to manage the relocation of user sessions between application instances by informing the source instance of the target instance and by enabling user context transfer between the two instances.

The entities involved in application mobility are highlighted in Figure 12. Here, the client app runs on a device such as a mobile phone or on equipment installed in vehicle. The device hosting the client app can

communicate via cellular network with the MEC hosts, i.e. MEC Host (A) and (B). These hosts are where the MEC app' and app'' instances run. At the remote level, the application is implemented in the cloud backend to serve the client app. The client app can communicate with either the edge level MEC application instances or the application instance(s) in cloud backend via the underlying network to receive the service produced by the application.

Initially, the client app may communicate with the remote level application. Then when the MEC system identifies that the device is within the serving area of MEC Host (B) it may create an edge level application instance (e.g. MEC app'') to serve the user and offer an enhanced service to the client app, e.g. contextual based on the device location with overall lower latency. If the MEC system identifies device handover to a new location associated with MEC Host (A) through the application mobility service, it will then create another new application instance (i.e. MEC app') to continue serving the user. The application may need to synchronize the user context between application instances if they are running on the cloud backend, MEC Host (A) or (B) to ensure that the application service continuity is maintained.

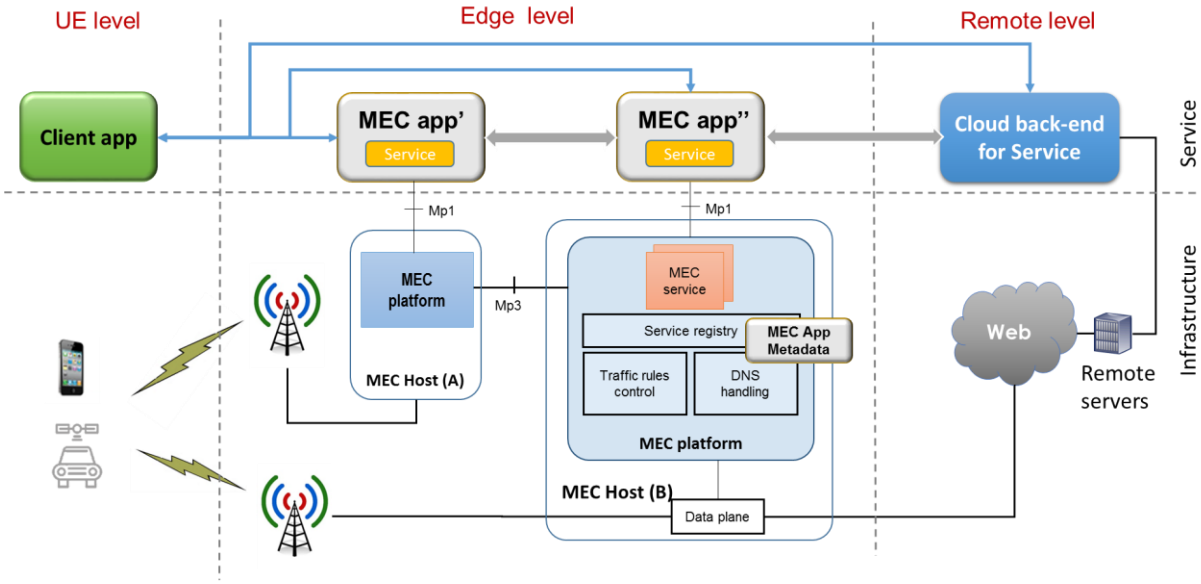


Figure 12: Application mobility support transferring the service between the Cloud back-end and MEC instance

User context is the application-specific runtime data maintained by the MEC application that is associated with a specific user or a group of users being serviced by that application. It may contain the client application identifier, the status of the application instance serving the user, the communication link and other runtime information. Such information is specific to and belongs only to the application. User context synchronization depends on the application implementation:

- For a stateless application, i.e. an application that does not retain the service state or recorded data about the user for use in the next service session, the application can be relocated to another MEC host without using the mobility service.
- For a stateful application, i.e. an application that stores information about service state during an application session change, it will require further application logic to transfer the user context from the source to the target application instance to maintain the service continuity.

MEC application mobility service can facilitate the support of service continuity for stateful applications, specifically by providing the necessary communication connections and support information exchange

between application instances. However, the synchronization of application service state and other specific runtime information relies on the application SW implementation, which could take one of two approaches:

- Store it in the client app of user device
- Store it in the MEC application instance

If the user context is stored in the client app, the client app may be required to send the user context to the target application instance after the application service is moved to another MEC host.

If the user context is stored in the MEC application, the application is required to support the capability to transfer the user context between MEC application instances. In this case the MEC application mobility service API can assist since it allows the application to subscribe to notifications on relevant mobility events. This enables the application to prepare the user context for transfer to the target instance in a timely manner. The mobility service can also be used to notify the source application instance of the address of its peer target instance and therefore where the user context needs to be transferred.

Future evolutions: Edgy DevOps

In the previous sections, it has been highlighted how the microservices variant of the service-oriented architecture (SOA) pattern has direct applicability to the software model envisaged for MEC and its services and applications. The approach's ability to compose a distributed application from separately deployable services was also described. Such services are expected to communicate via web interfaces (such as MEC's RESTful API approach) and perform a specific business function. In combination with this approach DevOps practices are considered as being highly complementary.

Currently the ecosystem is quickly moving to use Function as a Service (FaaS) approaches like Greengrass for Amazon's AWS Lambda, Microsoft's Azure IoT stack and GE's Predix to enable it. Also, DevOps seems to be a greenfield for MEC, as it is considered to be a business-driven software delivery approach. DevOps is based on lean and agile principles in which cross-functional teams collaborate to deliver software in a continuous manner. Such teams consist of representatives from all disciplines responsible for developing and deploying software services, including the business owners, developers, operations and quality assurance. This continuous delivery (CD) based approach enhances a business' ability to exploit new market opportunities and quickly adapt to customer feedback. The MEC ecosystem is evolving and presents opportunities for the development of new and innovative services, which means that the DevOps approach is ideally suited and offers a genuine path to deriving new business value.

Adopting the microservices approach results in services that are modularized and small in nature. Given their size, each individual service is easier to develop, test and maintain. The services may also be developed and deployed in parallel. This facilitates continuous integration (CI) and delivery, which are key principles of the DevOps approach. DevOps teams are also well placed to take the individual pieces of functionality offered through microservices and use those as the building blocks for larger applications and systems. Those systems can then be easily expanded by adding new microservices without unnecessarily affecting other parts of the application, thereby offering flexibility and achieving scalability. The consequence is that software development organizations already employing DevOps practices to develop microservices will already be well placed to embrace the MEC software development paradigm, where an overall MEC solution will comprise of multiple software components (i.e. microservices) each providing different capabilities and functions, potentially from different providers, and where those components are expected to continue to expand and evolve thereby benefiting from DevOps CI and CD processes. This expansion will happen at a high and low level, for instance at a high level as the MEC platform's overall capabilities are enhanced to support multiple access technologies and full application mobility and at a lower level as specific applications are further developed to exploit enhanced API functionality, for instance as the Radio Network Information Service capabilities expand into the multi-access domain.

An activity closely related to implementing the DevOps approach is the efforts underway in the MEC ISG to define a test framework to cover aspects such as conformance and interoperability. The framework will deliver a test suite that is ideal for automated testing. This is considered an integral DevOps component, since automated testing supports the delivery pipeline in creating processes that are iterative, frequent, repeatable, and reliable.

Figure 13 shows an overview of different cloud business models where different key touch points are expected to be required and standardized for the different cases.

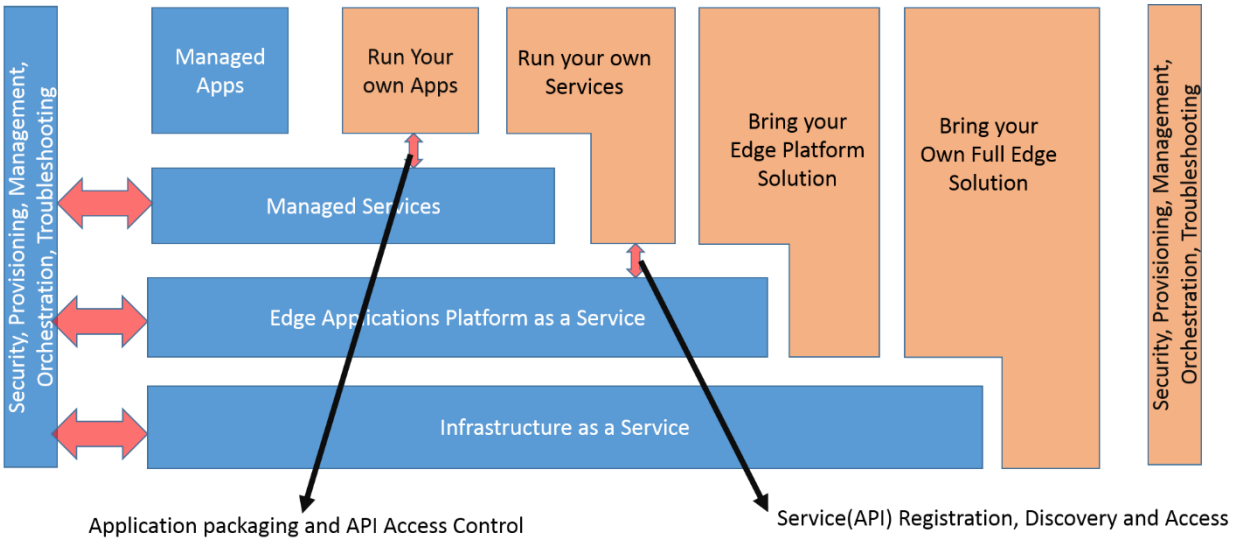


Figure 13: Overview of different cloud business models

Based on this layered model, while the fundamental principles of DevOps such as Continuous Integration and Continuous Delivery (CI/CD) apply across the board, the realization of such practices will differ significantly depending on the software layer where the development is performed and the management boundaries of the Edge Stack. Developers of enterprise applications or operator centric Network Functions will have a different deployment environment compared to consumer application developers and the tools used to deploy an application to a CPE will be different than for one deployed in an Edge Data Centre. One of the key operational requirements for Edge Computing is also the concept of Zero-Touch provisioning. This applies to all the layers of the edge stack and requires full management automation and service assurance.

To address these variations in the broad scope for edge computing for telco and enterprise applications, Akraino a relatively new Linux Foundation project attempts to break up the end to end framework into functional modules and use case driven blueprints (integrated stacks). This is an integration project in the sense that most of the code is being integrated from upstream projects via CI/CD tools. The resulting Edge Stack deliverables are tested, deployable platforms for telco and enterprise edge applications. Akraino Edge Stack brings enterprise application developers cloud deployment and management tools tailored for the edge application development.

Consumer application developers will look at a more abstract deployment model where the application follows an intent based, declarative approach that decouples the functions from the platform and execution environment. In this scenario, much of the DevOps process is being pushed to the platform including application package validation, testing, dependency checking, app update or roll back. To achieve this, the application must adhere to strict packaging specifications including well defined manifests, descriptors and operational models for scaling/support.

There are many requirements that make automation mandatory for the Edge Computing ecosystem to enable Zero Touch provisioning and while this improves the overall integration, DevOps is not eliminated. It just gets shifted into the platform at much higher layers in the stack.

FaaS make development of applications easier with less code to write and with a clear separation of concern between the function implementation and its consumers.

While application developers can and should be able to compose applications out of available functions using less code it does not eliminate the need to address compatibilities of their app with Function/Microservice versions deployed and consequentially deal with updates/upgrades of the platform.

Concluding Remarks

This paper addressed the issues that application developers face when developing applications for edge computing, including MEC-based edge computing. While offering only a superficial treatment, the paper provided a guide to developers for further reading and a starting point on how to address the challenges posed by this new type of application hosting environment.

All the authors of this white paper are active in ETSI ISG MEC. As the only international standards committee focused on edge computing, ETSI ISG MEC continues to work on simplifying the application development process and enabling interoperability through the definition of a reference architecture, standardization of APIs and other activities in this space. We invite the readers to learn more by visiting our web pages, http://www.etsi.org/mec. For those interested in participating in our work, the web page also contains a link to information on how to join our group.

Annex A - useful material for SW developers

This annex contains a list of useful material for software developers for MEC, e.g. references to repositories, open source communities and software frameworks.

In addition, few examples of collaborative projects related to MEC are provided.

Name	description	Link
ETSI MEC specifications	Hub with the latest published MEC specifications.	https://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing
MEC API definitions	GIT Repository with an OpenAPI compliant implementation of MEC APIs	https://forge.etsi.org/
ETSI NFV specs	Hub with latest Network Functions Virtualisation specifications	https://www.etsi.org/technologies-clusters/technologies/nfv
NEV SDK	NFV platform for MEC applications and services	https://networkbuilders.intel.com/network-technologies/nev
OpenAPI specification	OpenAPI GitHub	https://github.com/OAI/OpenAPI-Specification
OpenAPI Initiative	OAI Developer Community	https://www.openapis.org
Data Plane Development Kit (DPDK)	Library to accelerate packet processing	https://www.dpdk.org/
MEC wiki	ETSI wiki for MEC	https://mecwiki.etsi.org
Openstack	Open Infrastructure website and introduction to MEC	https://www.openstack.org/edge-computing/cloud-edge-computing-beyond-the-data-center/
Development resources for OpenStack clouds	Resources for application development on OpenStack	https://developer.openstack.org/ https://www.openstack.org/edge-computing/

Annex B - Collaborative projects related to MEC

Name **5GMedia (H2020)** http://www.5gmedia.eu/

Description 5G-MEDIA aims at delivering an integrated programmable service platform for the development, design and operations of media applications in 5G networks by providing mechanisms to flexibly adapt service operations to dynamic conditions and react upon events (e.g. to transparently accommodate auto-scaling of resources, VNF re-placement, etc.) by using an ETSI MANO framework.

The role of ETSI MEC is relevant to manage and extend the core cloud-computing capabilities towards the edge of the mobile network, within the Radio Access Network (RAN) and in close proximity to mobile users. 5GMedia aims at demonstrating a variety of media related use cases, such as the dynamic instantiation and management of media caches to serve Ultra High Definition contents to moving mobile devices.

Name **5GCity (H2020)** https://www.5gcity.eu/

Description The 5GCity project is working to design, implement and deploy a distributed cloud, edge and radio platform for smart cities and infrastructure owners acting as 5G Neutral Hosts. In 5GCity, cloud and edge computing technologies are integrated and extended to address 5G requirements and implement network slicing and end to end service orchestration from the radio up to the core data centres. To validate the 5GCity platform we are developing challenging media-related use cases which will be soon deployed in live 5GCity infrastructures in the cities of Barcelona (ES), Bristol (UK) and Lucca (IT).

The 5GCity platform is designed to leverage both ETSI MEC and ETSI NFV architectures and interfaces to implement the neutral hosting features. We have followed the guidelines provided by ETSI MEC (ETSI GR MEC 017 V1.1.1) and designed a concrete integration of ETSI MEC and NFV through a thin orchestration layer on top of the NFV and MEC orchestrators. This new orchestrator enables the coexistence of the different descriptors used in the two frameworks and a smooth communication between them across the different layers and scopes of action. In 5GCity, the MEC orchestrator controls the edge applications and the edge platform management, while the NFV orchestrator performs the actual functions deployment on the NFV Infrastructure (NFVI). The NFV orchestrator is aware of the deployment of MEC/Edge Apps which is achieved by their deployment as VNFs (as recommended by ETSI MEC specifications.

The 5GCity consortium is formed by 18 partners from 7 European countries, including leading vendors, network and infrastructure operators, research centres and SMEs all heavily involved in 5G, MEC and NFV development. The 5GCity project is a 5G PPP Phase 2 project, funded by the European Commission under the of the Horizon 2020 programme.

Name **NRG-5 (H2020)** www.nrg5.eu

Description The NRG-5 project envisages contributing to the 5G PPP/5G Initiative research and development activities and participation at the relevant 5G Working Groups by delivering a novel 5G-PPP compliant, decentralized, secure and resilient framework, with highly availability able to homogeneously model and virtualize multi-homed, static or moving, hardware constrained (smart energy) devices, edge computing resources and elastic virtualized services over electricity and gas infrastructure assets combined with the telecommunications infrastructure covering the full spectrum of the communication and computational needs.

The ultimate project goal is to render the deployment, operation and management of existing and new communications and energy infrastructures (in the context of the Smart Energy-as-a-Service) easier, safer, more secure and resilient from an operational and financial point of view.

Name **5GEssence** http://www.5g-essence-h2020.eu/

Description 5G ESSENCE addresses the paradigms of Edge Cloud computing and Small Cell as a Service by fuelling the drivers and removing the barriers in the Small Cell market, forecasted to grow at an impressive pace up to 2020 and beyond and to play a key role in the 5G ecosystem. 5G ESSENCE provides a highly flexible and scalable platform, able to support new business models and revenue streams by creating a neutral host market and reducing operational costs by providing new opportunities for ownership, deployment, operation and amortization.

The technical approach exploits the benefits of the centralization of Small Cell functions as scale grows through an edge cloud environment based on a two-tier architecture: a first distributed tier for providing low latency services and a second centralized tier for providing high processing power for computing-intensive network applications. This allows decoupling

the control and user planes of the Radio Access Network (RAN) and achieving the benefits of Cloud-RAN without the enormous fronthaul latency restrictions. The use of end-to-end network slicing mechanisms will allow sharing the 5G ESSENCE infrastructure among multiple operators/vertical industries and customizing its capabilities on a per-tenant basis. The versatility of the architecture is enhanced by high-performance virtualization techniques for data isolation, latency reduction and resource efficiency, and by orchestrating lightweight virtual resources enabling efficient Virtualized Network Function placement and live migration.

Name	**Matilda (H2020)**	http://www.matilda-5g.eu/
Description	The vision of MATILDA is to design and implement a holistic 5G end-to-end services operational framework tackling the lifecycle of design, development and orchestration of 5G-ready applications and 5G network services over programmable infrastructure, following a unified programmability model and a set of control abstractions. It aims to devise and realize a radical shift in the development of software for 5G-ready applications as well as virtual and physical network functions and network services, through the adoption of a unified programmability model, the definition of proper abstractions and the creation of an open development environment that may be used by application as well as network functions developers. Intelligent and unified orchestration mechanisms will be applied for the automated placement of the 5G-ready applications and the creation and maintenance of the required network slices. Deployment and runtime policies enforcement is provided through a set of optimisation mechanisms providing deployment plans based on high level objectives and a set of mechanisms supporting runtime adaptation of the application components and/or network functions based on policies defined on behalf of a services provider. Multi-site management of the cloud/edge computing and IoT resources is supported by a multi-site virtualized infrastructure manager, while the lifecycle management of the supported Virtual Network Functions Forwarding Graphs (VNF-FGs) as well as a set of network management activities are provided by a multi-site NFV Orchestrator (NFVO). Network and application-oriented analytics and profiling mechanisms are supported based on real-time as well as a posteriori processing of the collected data from a set of monitoring streams. The developed 5G-ready application components, applications, virtual network functions and application-aware network services are made available for open-source or commercial purposes, re-use and extension through a 5G marketplace.	

Name	**5G-Coral**	http://5g-coral.eu/
Description	5G-CORAL project leverages on the pervasiveness of edge and fog computing in the Radio Access Network (RAN) to create a unique opportunity for access convergence. It envisions a distributed edge computing platform also suitable for constrained devices.	

Name	**5G-Transformer**	http://5g-transformer.eu
Description	5G-Transformer aims to transform today's rigid mobile transport networks into an SDN/NFV-based mobile transport and computing platform. It envisions an edge computing platform capable of offering services tailored to the specific needs of vertical industries.	

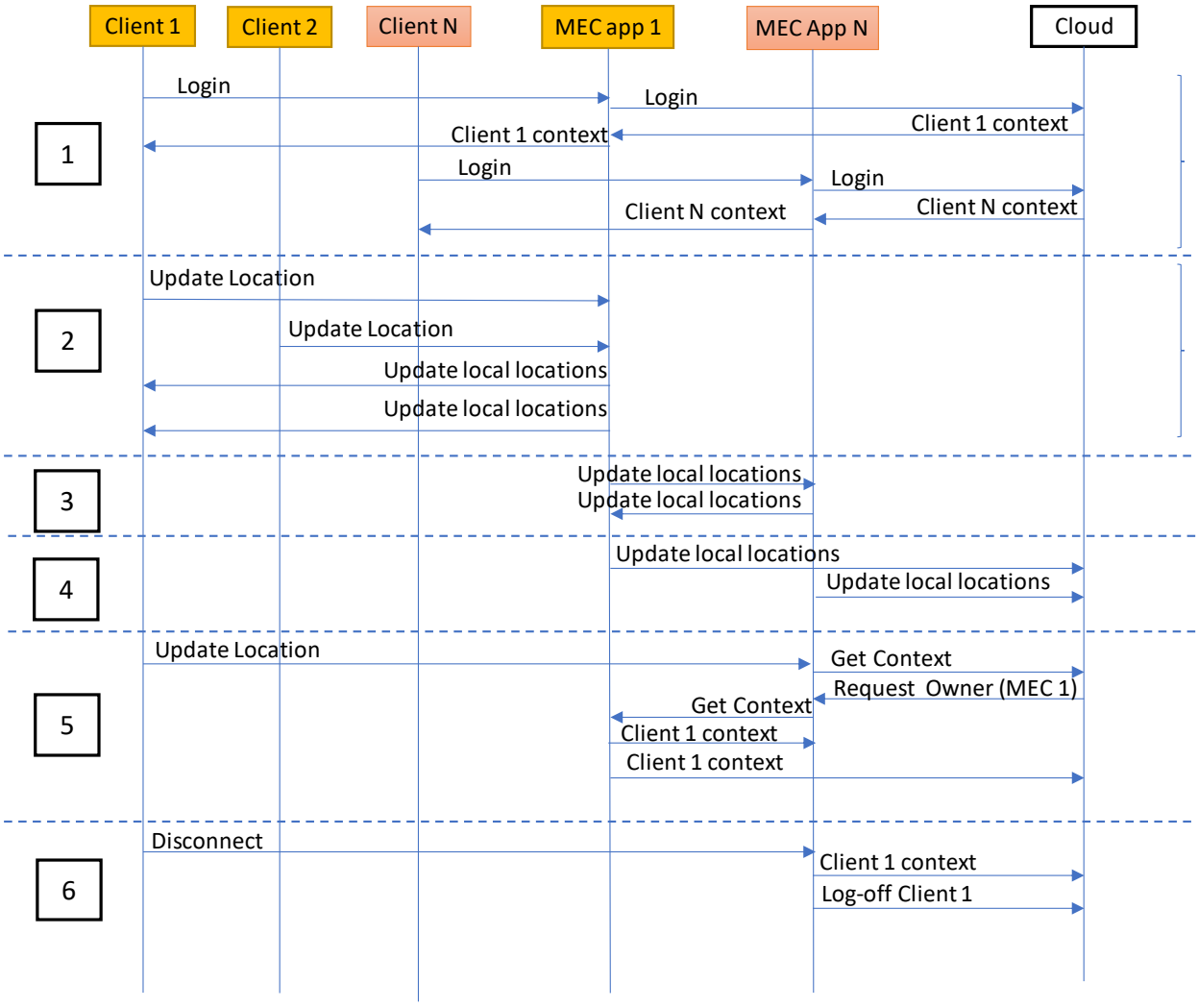
Annex C - Exemplary use cases for developers

V2X service on collision prevention

This use case illustrates a possible implementation of collision prevention V2X service, where all client apps running on vehicles regularly share their locations, directions and speed with others in proximity, so each client can detect potential collision and make preventive actions like change direction or speed.

This use case is very latency critical, where delayed decisions could be the difference between whether a collision happens or not. For instance, at 60 km/h a vehicle moves by 16 cm every 10 ms.

In the example below clients 1 and 2 are served by MEC instance 1 and client N by MEC app instance N. Client 2 is already connected, but client 1 has to initiate connection.



1. Client 1 connects to local MEC app instance 1. It establishes TCP connection to MEC Application using DNS redirection by MEC platform to resolve known domain name. Once connection is established, it uses Login API and provides credentials. MEC app 1 can't authenticate the user locally and therefore forwards the request to Cloud back-end, which verifies credentials and

responds with a Single Sign-On token and Client 1 Context. The same process takes place with Client N and MEC app N instance.

2. Service Apps running on clients regularly shares location, time of measurement, direction and speed with MEC instances. MEC app accumulates the locations of served Clients and updates each locally served Client with recent updates containing locations of clients in close-proximity (e.g. within a 100m/300ft radius).
    - Each client computes the possibility of collision using its current location and the extrapolated location of nearby clients.
3. MEC regularly sends the list of Clients which are in proximity of areas served by other MEC app instances to those instances (which is important for smooth hand-overs). Prior to this step, MEC apps discover peers and their physical location with assistance of MEC platform.
4. MEC instances updates the cloud-backend with latest information about clients on a regular basis, though significantly less frequently than updates to MEC app peers and clients.
5. Client 1 moved to area served by MEC app instance N and its traffic is now being handled by that instance. MEC Instance N doesn't yet know this client but receives its "single sign on" (SSO) token and forwards the request to Cloud backend to validate it and get associated context. The cloud back-end maintains a directory of which client is served by which MEC app instance and refers to MEC 1 to request context. MEC N sends request to MEC 1 which responds with context to both MEC N and Cloud and drops local context as it's now owned by MEC N.
6. Once client 1 disconnects, the MEC instance updates the cloud with the recent context and sends a Logoff message to drop the authentication token and Client's 1 context.

Video transcoding use case

With the advent of 5G, the media sector is witnessing the emergence of a number of innovative services. Among these, advanced video applications are attracting a particular attention. Many factors can explain such interest. According to Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016-2021, White Paper [8], video data traffic will grow to 82% of all internet traffic by 2021. Moreover, consumers' preferences have shown a clear shift from traditional services towards advanced video activities and social networking, carried out through a variety of personal devices, such as smartphones, tablets, smart watches, personal computers, etc.

During sport events or concerts, consumers increasingly often wish to originate and share high quality media contents in groups of peers, besides merely receiving them from a centralized distribution system. The produced contents must be available as flash events – i.e. in the shortest possible time - and adapted to the consumers' device capabilities, to optimize the perceived Quality of Experience.

In touristic locations or relevant cultural sites, the use of ad hoc multimedia contents, 3D reconstructions (of monuments, statues, or buildings etc.), and panoramic videos can be combined with geo-localization services, augmented reality, and object recognition capabilities, to offer highly immersive experiences to end users. For instance, in a historical location, tourists could see on their personal devices the reconstruction of an antique building overlapping the real landscape.

Using the architecture depicted in another section, Figure 2, edge applications can be developed and used to provide an immersive experience to end users. To do so several different interacting components are

required, concurrently running on the end users' devices, in the virtualization infrastructure at the edge, and in one or more core data centres.

Consumers can access the immersive video services through a specific application, downloadable on their personal device, or, alternatively, through any browser. This way, they can register to the service and create groups of peers sharing media contents, off-line or in real-time. The end user app also provides geo-localization data to the MEC App components running at the edge and sends and receives the multimedia streams that originate the immersive experience.

The MEC App running at the edge consists of three different components (that can run both as Virtual Machines and containers), namely the dataBase (dB), the eStore and the Media Processing Unit (MPU) components. Such components are independent entities, which offer their services through Restful APIs registered on the MEC platform.

The dB component collects and monitors relevant information about users and groups. When consumers connect to the local access network for the first time, they can download the end user app. The next step is the registration to the system, which identifies each user by a name and/or nickname. Through the interaction between end-user app and the dB component, consumers create groups that can share media contents. For each group member, the dB component saves the relevant information about identity, group belonging, connectivity, and Service Level Agreement, to provide the required type of service. Moreover, it provides basic security functionalities, such as user password management, and records the geo-localization data periodically sent by the end-user device app.

The eSTORE component offers local storage capabilities to the groups of users. This way, the operation of content upload and/or download to the storage system can be significantly speeded-up, since it does not require any access to the core network. This feature is particularly relevant during crowded events, when the backhaul connection becomes a contended resource that can originate bottlenecks for any activity involving the core network.

The basic task of the MPU component is converting audio/video streams from one format to another, to adapt to the capabilities of the receiving end-user device. A multimedia content could reside as a pre-recorded file in the storage system or come as a real-time packet stream. The requested transcoding service can be mono-directional, as in video stream distribution applications, or bi-directional, as in real-time videoconferencing.

The MPU supports the most popular video codecs, such as H.265 and Google VP9. It can take advantage of the potential presence of GPUs at the edge to improve overall performance and efficiency. Video streams originated by consumers are saved in the distributed storage system (eSTORE), to be shared later. Pre-recorded contents, for instance coming from a broadcaster archive, can be streamed on demand to single users, to an entire group, or to all the users.

The MPU offers the advanced audio/image/video processing capabilities needed to provide immersive video services. Specifically, it can stream to consumers 360° video data originated from specific multi-sensor cameras. To this end, the MPU processes the different video contributions coming from the camera, performing all the needed adjustments in terms of image stitching, colour correction, projection in the equi-rectangular format, etc, - so that consumers can select in real-time on their device the viewing direction in the observed scene.

To enhance the immersive experience, the MPU can provide 3D reconstructions of spaces (such as a square or a view) or objects (monuments, building etc.). In addition, it can recognize objects in the images and video streams sent by end users, to perform automatically visual searches in ad hoc image or video databases. The visual search allows matching images or videos captured by the user, such as buildings, statues, paintings, with contents present in a database, exploiting visual similarities and without the need for manual query.

Finally, certain specific components of the overall immersive video application can run in the core cloud. For instance, they allow the permanent storage of multimedia contents originated during a crowded event, temporarily saved at the network edge. In addition, specific back-end components can run in the core cloud, to process large data sets during the training phase of machine learning algorithms, or to run the most computationally intensive and time-consuming activities in 3D reconstruction.

The MEC framework can significantly contribute to the successful implementation and provision of the immersive video services above described. To this end, it can offer not only low latency, local computing and storage resources, and high data-rates, but also localization awareness and flexibility to radio link conditions.

From the perspective of MEC App developers, the capabilities offered by the MEC Platform and the MEC management blocks can dramatically simplify both the development phase of new applications, and the entire lifecycle of the offered services.

As an example, the coordinated activities of the MEC management blocks and the MEC Platform can straightforwardly solve in a fully automated way the complex problem of properly steering the user data towards the corresponding MEC App running in the local edge infrastructure. To this end, App developers must only specify, during the on-boarding process of an MEC App, a Fully Qualified Domain Name, which identifies the implemented service; the overall MEC framework has the capability to produce and enforce into the MEC data plane the corresponding DNS and traffic steering rules.

References

[1] Mobile-Edge Computing – Introductory Technical White Paper, September 2014; https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge_computing_-_introductory_technical_white_paper_v1%2018-09-14.pdf

[2] ETSI White Paper No. 11: "Mobile Edge Computing: A key technology towards 5G", ISBN No. 979-10-92620-08-5, first edition, September 2015; https://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp11_mec_a_key_technology_towards_5g.pdf

[3] ETSI GS MEC 011: "Mobile Edge Platform Application Enablement", http://www.etsi.org/deliver/etsi_gs/MEC/001_099/011/01.01.01_60/gs_MEC011v010101p.pdf

[4] ETSI GS MEC 016: "UE application interface", http://www.etsi.org/deliver/etsi_gs/MEC/001_099/016/01.01.01_60/gs_MEC016v010101p.pdf

[5] ETSI GS MEC 010-2: "Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management", http://www.etsi.org/deliver/etsi_gs/MEC/001_099/01002/01.01.01_60/gs_MEC01002v010101p.pdf

[6] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage", https://tools.ietf.org/html/rfc6750

[7] https://restfulapi.net/

[8] Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2016–2021 White Paper, Updated:March 28, 2017; https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[9] ETSI GS MEC 012: "Mobile Edge Computing (MEC); Radio Network Information API", V1.1.1, July 2017. https://www.etsi.org/deliver/etsi_gs/MEC/001_099/012/01.01.01_60/gs_MEC012v010101p.pdf

[10] ETSI GS MEC 013: "Mobile Edge Computing (MEC); Location API", V1.1.1, July 2017. https://www.etsi.org/deliver/etsi_gs/MEC/001_099/013/01.01.01_60/gs_MEC013v010101p.pdf

[11] ETSI GS MEC 009: "Multi-access Edge Computing (MEC); General principles for MEC Service APIs", V2.1.1, January 2019. https://www.etsi.org/deliver/etsi_gs/MEC/001_099/009/02.01.01_60/gs_MEC009v020101p.pdf

[12] ETSI GS MEC 006: "Mobile Edge Computing (MEC); Market Acceleration; MEC Metrics Best Practice and Guidelines", V1.1.1, January 2017. https://www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/006/01.01.01_60/gs_MEC-IEG006v010101p.pdf

[13] ETSI GS MEC 003: "Multi-access Edge Computing (MEC); Framework and Reference Architecture", V2.1.1, January 2019. https://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/02.01.01_60/gs_MEC003v020101p.pdf

[14] ETSI GS MEC 015: "Mobile Edge Computing (MEC); Bandwidth Management API", V1.1.1, October 2017. https://www.etsi.org/deliver/etsi_gs/MEC/001_099/015/01.01.01_60/gs_MEC015v010101p.pdf

[15] ETSI GS MEC 014: "Mobile Edge Computing (MEC); UE Identity API", V1.1.1, February 2018. https://www.etsi.org/deliver/etsi_gs/MEC/001_099/014/01.01.01_60/gs_MEC014v010101p.pdf

[16] SCF 084.07.01: "Small cell zone services - RESTful bindings".

[17] SCF 152.07.01: "Small cell services API".

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

ETSI White Paper No. 23

Cloud RAN and MEC: A Perfect Pairing

First edition – February 2018

ISBN No. 979-10-92620-17-7

Authors:
Alex Reznik, Luis Miguel Contreras Murillo, Yonggang Fang, Walter Featherstone, Miltiadis Filippou, Francisco Fontes, Fabio Giust, Qiang Huang, Alice Li, Charles Turyagyenda, Christof Wehner, Zhou Zheng

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Editor: Alex Reznik (HPE)

Luis Miguel Contreras Murillo (Telefonica)

Yonggang Fang (ZTE)

Walter Featherstone (Viavi)

Miltiadis Filippou (Intel)

Francisco Fontes (Altice Labs / Portugal Telecom)

Fabio Giust (NEC)

Qiang Huang (ZTE)

Alice Li (Vodafone)

Charles Turyagyenda (InterDigital)

Christof Wehner (Artesyn)

Zhou Zheng (Huawei)

Contents

Executive Summary

CRAN and MEC are highly complementary technologies. Collocating these helps make the economics of each of them significantly more attractive. Collocating CRAN and MEC also helps an MNO to support (and generate revenue from) some of the key 5G applications that it would not be able to support otherwise.

However, to realize these advantages, mobile operators have to overcome challenges associated with co-location, as well as maximize the return that can be made from MEC. We identify and discuss challenges in the management, security, networking and regulatory domains. We argue that these issues are surmountable and the industry is well-positioned to deploy this potentially revolutionary new technology.

Moreover collocation can also enable MEC services (e.g. the ETSI defined Radio Network Information API, Location API, UE Identity API and Bandwidth Management API) to exploit CRAN and enable MEC applications to exploit CRAN information. Mobile operators could, for example, resolve the management complexities associated with multiple IaaS stacks, while monetizing services like RNIS, which are unique to MEC edge clouds.

CRAN and MEC: benefits of co-deployment

Edge presence is viewed as absolutely necessary to enable certain use case classes defined for 5G. The 5G use cases have been classified into three service types (see, e.g. [1]): eMBB (enhanced Mobile Broad Band), URLLC (Ultra Reliability and Low latency Communications), and mMTC (massive Machine Type Communications). In particular, the URLLC service type includes use cases related to Tactile Internet, Interactive Gaming, Virtual Reality, automotive, industry and automation. A common characteristic of these use cases is the need for low end-to-end latency. Physical limitations (i.e., speed of light) prohibit execution of these use cases in the traditional "deep" or "remote" cloud. The eMBB service type encompasses another kind of challenge - a previously unseen volume of upstream data associated with, for example, high-definition video sharing. Finally, the mMTC set of use cases covers applications where a large number of IoT devices, such as sensors, are sending data upstream, collectively creating a significant data volume passing through the network. Moreover, this data is highly localized and is often associated with a requirement (due to privacy, data ownership, etc.) that it shall not cross certain domain boundaries. It can, therefore, be concluded that the 5G use cases all call for some processing of data and/or proximity at the edge of the Radio Access Network (RAN).

From a Mobile Network Operator's (MNO) point of view, a major challenge in enabling applications associated with the 5G use cases is the significant investment required to deploy a sufficiently extensive network of edge computing Points-of-Presence (PoPs), so that it becomes attractive to develop applications exploiting the edge processing infrastructure in mind. Moreover, this investment must be made in advance of applications being ready to take advantage of it – i.e., this is an investment in anticipation of future revenue, but without any guaranteed near-term returns. One way to mitigate the significant cost (and risk) of such strategic investment is to bootstrap a Multi-Access Edge Computing (MEC) deployment to the deployment of a Cloud RAN (CRAN): the cost of providing additional processing power across an already planned pool of centralized processing points (e.g., a pool of Base Band Units (BBUs)), should be significantly lower than a standalone MEC deployment.

Conversely, deployment of a CRAN across generic computing infrastructure (as opposed to dedicated, RAN-optimized hardware) is itself a significant investment for an MNO. In addition to the costs of deploying CRAN processing units themselves, there is the cost of moving towards virtualized RAN appliances, testing, integration and maintenance of these new solutions. While the operational flexibility and network re-configurability offered by virtualization may carry significant long-term benefits, the near-term effort and costs can make it a tough pill to swallow. The significant strategic benefits of MEC can make the decision a much clearer one.

Among the investments in mobile network infrastructure, the RAN represents the major part of the MNO's Capital Expenditure (CAPEX) - this is in addition to the cost of the spectrum itself, while maintenance, possible use of leased transport network lines and network optimization add significant additional Operational Expenditure (OPEX). Given this situation, a CRAN deployment which virtualizes much of the RAN functionalities on standard General-Purpose Processors (GPPs), is seen as an important technology enabler for reducing the Total Cost of Ownership (TCO), associated with the RAN. The amount of investment and the Operation and Maintenance (OAM) costs are expected to decrease fast thanks to maturing cloud technologies and deployment experience. The CRAN approach facilitates a faster radio deployment, drastically reducing the time needed in conventional deployments. There are evident CAPEX and OPEX benefits derived from a more efficient site management (less rented space and energy, easier negotiation with owners, etc.), energy savings, network simplicity (for current and advanced

functionalities like radio coordination) and higher levels of security. On top of that, CRAN also facilitates the introduction of Artificial Intelligence (AI) in RAN to truly turn it into a “smart” RAN.

The business models of MNOs have already changed fundamentally from offering bit pipes through their networks towards a data centric network driven by the services offered there. With regards to such a new data centric network, the network resources are efficiently utilized by virtual datacentres (vDCs). In contrast to past deployments, where an MNO had to deploy hard infrastructure everywhere from the city centre to the far end, in a modern service-driven network some of the traditional network sites and functions are becoming redundant. A RAN functionality can be flexibly deployed across multiple different locations over the same generic compute substrate. For example, a CRAN processing node can be deployed anywhere from what used to be the Central Office (CO) of a Public Switched Telephone Network (PSTN) (now more like a datacentre) to an in-field aggregation site for several cell-sites, to a cell-tower co-located hut. And because a CRAN deployment requires a substantial amount of processing power, any such site automatically becomes a MEC site – easily scalable to support other workloads.

To summarize, CRAN and MEC are highly complementary technologies. When considered together, they make the near term economics of deploying CRAN hubs based on generic processing components much more attractive, while positioning an MNO to support (and generate revenue from) some of the key 5G applications that it would not be able to support otherwise.

Co-location: Deployment Scenarios and Architecture

As noted above, CRAN and MEC are perfectly paired to accommodate emerging services, especially those requiring low latency or high bandwidth. Moreover, an attractive aspect of a cloud-based approach is that it enables a scalable solution, in particular making the capacity of the CRAN dynamic. How close a CRAN/MEC site is located to cell-sites will often determine how well it can support certain applications – or whether it can support them at all. As with most other things, it is a trade-off between cost and performance – locating a CRAN/MEC site in a CO is often less expensive than doing so in the field, but the cost is higher latency. Therefore, a careful understanding of the use-cases – i.e. which applications are likely to run at such a site – is critical. Table 1 presents a summary of such a use case analysis from [2].

Table 1: Exemplary use case analysis

Service	Content Sever	Characteristic			Cloud-Edge Coordination	Possible Location
		Latency	Bandwidth	Privacy		
AR/VR	Local	<5ms	100Mbps~ 9.4Gbps	No	Sync but not real-time	Access ring (Edge DC)
V2X	Local	<10ms	>100Mbps	No	Processed data real-time Sync	Access ring (Edge DC)
Video Surveillance	Local	Variable	>20Mbps	No	Processed data real-time Sync	Access ring (Edge DC)
Smart factory	Local	<10ms	Variable	Yes	Only in private Cloud	Factory (Edge DC)
Enterprise Cloud (e-health)	Local	<10ms	Variable	Yes	Only in private Cloud	Enterprise (Edge DC)
IOT management	Local /Cloud	Variable	Variable	No	Processed data but not real-time Sync	Access ring or Collector ring (Edge DC or Local DC)
Entertainment (8K TV and Gaming)	Cloud	10ms	>100Mbps	No	Local caching	Collector ring (Local DC)

Given the various application types and requirements that may be present at a CRAN/MEC site it is conceivable that the infrastructure is portioned into multiple domains, as shown in Figure 1.

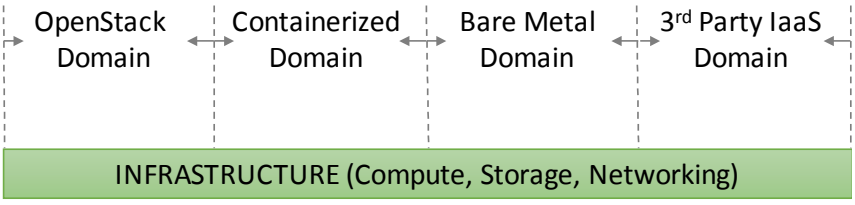


Figure 1: CRAN/MEC site architecture example

The following scenario illustrates the usage of such an infrastructure. Suppose an MNO wants to run the following applications at the edge:

- An MNO's own applications, which are Virtualized Network Functions (VNFs) that run on, e.g., OpenStack
- Third-party Cloud Applications using a cloud-native based stack such as Microsoft's Azure Stack
- A CRAN implementation, which runs on bare-metal
- Cloud-native containerized applications (using, e.g., Docker, AWS Greengrass, Microsoft Azure IoT Stack)

Since both OpenStack and AzureStack are complete stacks, co-located deployment of these features requires support for at least two independent OAM stacks. Moreover, at least one of these stacks must support the "Containerized Domain" (or else a third OAM stack may be needed). Finally, bare metal resources need to be managed by one of these OAM stacks (e.g. using OpenStack's Ironic), or else also be considered a separate (fourth!) "OAM stack".

To summarize, co-location of MEC and CRAN – while technologically and economically attractive – presents several challenges which much be addressed so that the value of the co-location can be fully realized. The next sections delve deeper into this subject.

Challenges in co-location

Repurposing existing CRAN deployments as MEC deployments is a frequent subject of on-going discussions. Ideally, this should result in cost as well as rack space and power savings. But is this really the case? Or does it make sense to deploy MEC next to, or near, CRAN instead? The answer depends on the actual use case and deployment scenario.

For deployment, we need to think about *brownfield* vs *greenfield.* In a *brownfield* deployment, existing equipment is being partially replaced, or augmented, with new MEC capable equipment. On the other side, in a *greenfield* deployment all-new equipment and virtual functionality is deployed.

The use case dependency is more complex. For instance, in a shopping mall, systems are probably going to find themselves in a difficult-to-handle environment at the edge, rather than a well-controlled datacentre-style site. Such a challenging environment drives the need for dense, high performance compute outside of the datacentre and creates a need for diversity in MEC infrastructure. Datacentre like MEC deployments can easily use standard computing equipment; however, such infrastructure is not optimized for the environment we described above. Such environments require Commercial-of-the-Shelf (COTS) designs that have the capability to meet strict space constraints as well as to operate in dusty, less-well maintained, less temperature-regulated conditions. What follows is an analysis of important considerations to take into account when thinking about co-locating CRAN and MEC.

Management

The MEC system consists of functional as well as management and orchestration (MEC-MANO) entities, which enable applications to run as virtual machines in a virtualized computing environment, following the Infrastructure-as-a-Service (IaaS) model. Through the MEC management interfaces [3], [4], the MEC system supports operations such as on-boarding applications, creating instances and orchestrating services. In an incremental *brownfield* deployment approach, the MEC system would likely appear as a standalone managed object sitting next to the existing CRAN implementation. Jointly managing the two would require some kind of *a posteriori* intervention aimed at harmonizing the MEC-MANO part with the CRAN. Such activity can be extremely challenging if the CRAN part requires its own components for infrastructure management, e.g., when the CRAN sits on bare metal, which shares little with the MEC components for virtualized infrastructure management.

Network Function Virtualization (NFV) is a powerful emerging technique in the telecom industry used to decouple the network functions from dedicated physical network hardware and allow the network services to be operated in a virtualized environment. Therefore, NFV is deemed to facilitate CRAN deployments; in fact, CRAN contains non-real time functions (such as RRC, PDCP, etc.) and real time functions: whereas the latter would be still implemented as physical network functions (PNF) the non-real time functions could be virtualized as per the NFV paradigm.

ETSI ISG NFV is the leading Standard Developing Organization (SDO) in the NFV space, with the NFV management and orchestration (NFV-MANO) system being one of the achievements most relevant to the market. Like NFV, which provides the virtualized infrastructure to run network functions, MEC also uses a virtualization platform to run the applications. Therefore there is the possibility for MEC and NFV to share the same MANO and NFV Infrastructure (NFVI) systems to manage, orchestrate and execute the applications and services. ETSI ISG MEC has issued a study [5] on MEC operating in an NFV environment to allow MEC to re-use NFVI as virtualized infrastructure in either standalone or shared with NFV. Assuming

both MEC and CRAN are built on the top of same NFVI, an approach under investigation of ISG MEC is to have MEC MANO to communicate with NFV MANO to invoke the services by NFVI at IaaS layer. Therefore MEC would be able to use the MANO and VNFs of CRAN to orchestrate its applications and services. But the challenge of this approach is to maintain two MANO systems to manage the applications.

The MEC system not only interfaces to CRAN, but also interacts with applications. In Internet services, many applications are running in clouds, such as Amazon Web Service (AWS) Greengrass, Microsoft Azure and Google Cloud. Those clouds may use different virtualization stacks then CRAN. Therefore, it would be a challenge for MEC to support porting such applications from the cloud and managing them running on NFV based MEC without any modification.

A potential approach under investigation in ISG MEC is support of container-based virtualized environment for MEC applications. For example, the orchestration engine of container-based virtualization would be treated as a resource of IaaS and the Operating System (OS) image for containers could be run as a virtual machine. Therefore the container-based virtualized environment would be able to run as an independent Platform-as-a-Service (PaaS). As it is operated on different virtualization infrastructure from the NFV, the MEC MANO may not be the same as NVF MANO, which would be helpful to separate two different virtualization environments. But the challenge to this approach remains in the management systems, i.e. the CRAN operator needs to maintain two separate MANO systems.

There are several factors limiting the codeployment of existing RAN or CRAN systems. One of them is service availability. To date, few CRAN system have been deployed with sufficient provisioning for both radio access and service delivery capacity. As radio access is the current revenue driver, it is unlikely that MNO service providers will be inclined to risk service disruption. Thus, delivery of services requires a separate virtualized computing environment that has access to radio data, such as location and possibly user-plane data, and billing systems tied to the service consumer or the advertiser.

In private environments, e.g., shopping malls, the most likely scenario is that initial deployments will have equipment placed separately from the existing (working) RAN, following the IT motto “never change a running system”. Additionally, this enables a seamless cutover to add the new functionality, whereas a replacement may result in a day-long period without local connectivity, which is undesirable. Integration of CRAN into the fully managed virtual environment would still happen, but at a later time.

The benefits of augmenting existing systems are many, but the primary driver is revenue generation. The ability to provide location-aware services sets the stage for an entirely new revenue stream based on local advertisers addressing local shoppers. Provided security concerns are addressed the compute platform needs only connection to appropriate data from the radio system to be application ready.

In contrast, in a *greenfield* scenario, one can take advantage to jointly deploy equipment for both radio and MEC services, implementing the appropriate optimizations and/or customizations. In the shopping-mall or stadium environment, where most early deployments are expected, opportunities for applications and promotion within applications based on location arise (such as: “come buy dinner get free dessert”).

The key role of a MEC platform is to provide the necessary baseline support for applications, including the ability to route traffic to and from them, pointers to the appropriate Domain-Name-System (DNS) records and persistent storage. In addition, through the service registry the platform can keep track of and advertise the services available in that MEC host, so that consumer applications (or the platform itself) can discover services, and producer applications can make their services visible.

The scope of the MEC platform is, generally speaking, the MEC host, which is defined as the entity managed by MEC MANO stack. This implicitly restricts the host to a single IaaS domain under the control of a single VIM instance. Nevertheless, with the appropriate level of abstraction and interfaces, complex MEC deployments may span across multiple IaaS domains. As an example, one may think of an application in a containerized environment like AWS Greengrass which consumes the Radio Network Information (RNI) Service from an application running in the conventional MEC environment.

This abstraction and interfacing layer turns out to be necessary in order to grant inter-domain communication, where different coupling levels can be envisioned:

- *Loose coupling* refers to the ability of the MEC platform to make the service registry usable by non-MEC applications, i.e., those applications managed through a MANO stack different from MEC. In other words, loose coupling exposes MEC service APIs to applications sitting in other clouds.
- *Tight coupling* requires additional logic to propagate MEC-defined management instructions to other cloud domains, in order to concentrate management decisions at a single entity, e.g., to use the capabilities of the MEC platform manager in order to program the data plane from another cloud domain.

As already mentioned, MEC is already tackling the tight coupling problem between MEC and NFV domains. This should cater for exposing MEC platform services to the CRAN when the latter is deployed as a VNF, as per the ETSI-defined NFV system, including an optimized integration of the CRAN and MEC data planes.

Security

MEC allows the provision of new types of services, which also introduces potential security threats and vulnerabilities.

A likely model for MEC in a CRAN architecture is that MEC applications will run on the same physical platforms as some network functions. These applications may be third party applications, not controlled by the MNO directly. There are risks that these applications may exhaust resources needed by the network functions. There are also risks that some poorly designed applications could offer hackers an attack vector to infiltrate the platform and, hence, affect the network functions running on the platform – or even risks that malicious applications do the same thing themselves.

In particular, some MEC applications are intended to influence the mobile network configuration (including both RAN and Core Network (CN) parameters) in real time in order to improve network efficiency and customer experience. If this influence is too large, it could cause severe degradation, or denial of service to other users. Some applications might starve competitor applications (and their customers) of radio resources, either accidentally or maliciously.

It is, thus, paramount for service providers to be sure that a system malfunctioning in the MEC environment does not impact the CRAN part. Whether this malfunctioning comes from application malicious behaviour or a system crash is up to the service provider to troubleshoot and fix. In all cases, protection and isolation mechanisms should be in place in order to ensure that the CRAN components are still in service and able to deliver the mobile coverage to end users.

Meanwhile, where MNOs host third party applications in a MEC system, there is an opportunity for MNOs to provide security / assurance services for those applications. The example services include performing integrity assurance checks on applications at installation or upgrade, or after a server restart, and

exposing security services APIs to sufficiently trusted third party MEC applications, e.g. for user identification.

Networking

In order to benefit from all expected MEC advantages (e.g. low latency, backhaul traffic reduction and local breakout), the closest point to UEs for MEC deployments is close to the eNBs, which may be distributed or centralized. That location presents challenges related to mobility event handling (item being addressed by ETSI MEC) and guaranteeing execution of operations required on the mobile traffic that, with MEC close to eNB, may not reach the SGi interface.

CRAN, via the centralization it provides, has the potential to reduce the number of MEC hosts required to provide a service to the same population. However, different CRAN deployment strategies may be followed. The splitting point of functions between the distributed Remote Radio Heads (RRHs) and the centralized BBUs is just one aspect. For 4G, CPRI is commonly adopted, with RRH units only executing RF functions. For 5G this may be different due to the higher demanded bandwidths and the recently specified eCPRI [15] that will contribute to the solution. While this aspect shall not influence MEC, the type of BBU centralization (hostelling vs. pooling) determines the number of mobility events to be handled by MEC and, thus, overall system efficiency and the user experience.

By concentrating BBUs (hostelling) or reducing the BBU number (pooling), actions to be taken related to user mobility are local or even inexistent whenever the UE moves inside the geographical area covered by the centralized BBUs. Thus, the geographical area a UE can move without changing servicing MEC host gets bigger.

Running MEC close to CRAN in a scenario where the CRAN is virtualized, creates the conditions for MEC to share the same virtualization infrastructure. In such scenario the interfaces to be handled are established inside that infrastructure, provided the required handling and forwarding are available. In a scenario where the BBU may be implemented as a chain of modular functions, MEC may be inserted in that chain prior to S1 encoding.

Many of these challenges will also exist with 5G, which also uses GTP, and with other access technologies, whenever users' traffic crosses the edge encapsulated, with the sessions anchor point, QoS enforcement, IP addresses assignment, Lawful Interception (LI) and usage accounting being done deeper in the network, e.g. at a Broadband Network Gateway. The difference for 5G is that, being a technology currently under specification, it already accounts for edge computing, with 5G architecture entities (e.g. the User Plane Function) close to the UE, supporting those functions and allowing traffic to be steered by the Application Functions (AFs). Other features are being added, like a 'Common API Framework', to complement that.

Regulatory

MNOs are required to provide Law Enforcement Agency (LEA) support including Lawful Interception (LI) and Retained Data (RD) capabilities for traffic carried on their networks. Typically this functionality is supported by core network elements for all data passing through these elements.

When implementing MEC, some traffic may be generated or manipulated inside the MEC system or may come from a local breakout connection, thus not passing through the core network and not supported by the existing LI solution in the network.

Moreover, in the context of MEC, placing multiple additional LI points around the network edge raises security risks:

- there will be many more LI points than in traditional deployments
- edge nodes are likely to be more exposed to attack than core nodes.

It is therefore recommended that LI and RD collection functions are implemented at the edge of the network, alongside or as part of the functionality being intercepted. Any edge node including LI / RD collection features must support strong physical security requirements similar to core network sites. Further work would be required to examine specifically how and where the LI/RD functionality should be included in a network architecture.

Enabling and Exposing RAN Services in MEC

Two key issues of CRAN deployment with MEC are a) the ability of the CRAN to exploit the MEC service APIs within the CRAN, and b) the exposure of CRAN information to MEC applications. In the remainder of this section, we will focus on the role and possible benefits of exploiting the following MEC service APIs within a CRAN towards service optimization (e.g., measured by means of the Quality-of-Experience – QoE):

- Radio Network Information (RNI) API [6];
- Location API [7];
- UE Identity API [8];
- Bandwidth Management API [9].

The reference provided for each API refers to the respective ETSI MEC Group Specification (GS). To complement the specifications, ETSI ISG MEC also provides a supplementary description file compliant to the OpenAPI specification [10]for each of these APIs. OpenAPI is a specification for machine-readable interface files for describing, producing, consuming and visualizing RESTful web services. The description files are hosted on the ETSI Forge site: https://forge.etsi.org/.

RNI API

The Radio Network Information Service (RNIS) is a service that provides radio network related information to MEC applications and to MEC platforms. Typical information provided by RNIS includes radio conditions, user plane related measurements, radio access bearer information and corresponding change notifications.

In further detail, radio network information can be broadly classified into the following groups.

- **Radio Access Bearer (RAB) information**, which contains data about existing E-RABs associated with a specific MEC application instance. In addition to existing E-RABs, RNIS also provides information on RAB establishment, RAB modification and RAB release.
- **Public Land Mobile Network (PLMN) information**, which contains data about the underlying mobile network that the MEC application is associated to.
- **S1 Bearer information**, which represents data about the S1-U bearer. In addition to existing S1 bearers, the RNI service also provides information on S1 bearer establishment, modification and release.
- **Cell change information**, which includes the following information elements: handover status, PLMN information and E-UTRAN cell global identifier
- **UE RRC measurement reports**
- **UE timing advance**, which is necessary to ensure that uplink and downlink sub frames are synchronized at the eNB.

The RNI service exposes the radio network information to the RNI service consumers using a RESTful API. The standard RESTful methods, i.e., GET, PUT, POST and DELETE, can be utilised for RNI requests and responses, which support individual requests for information as well as subscription to notifications.

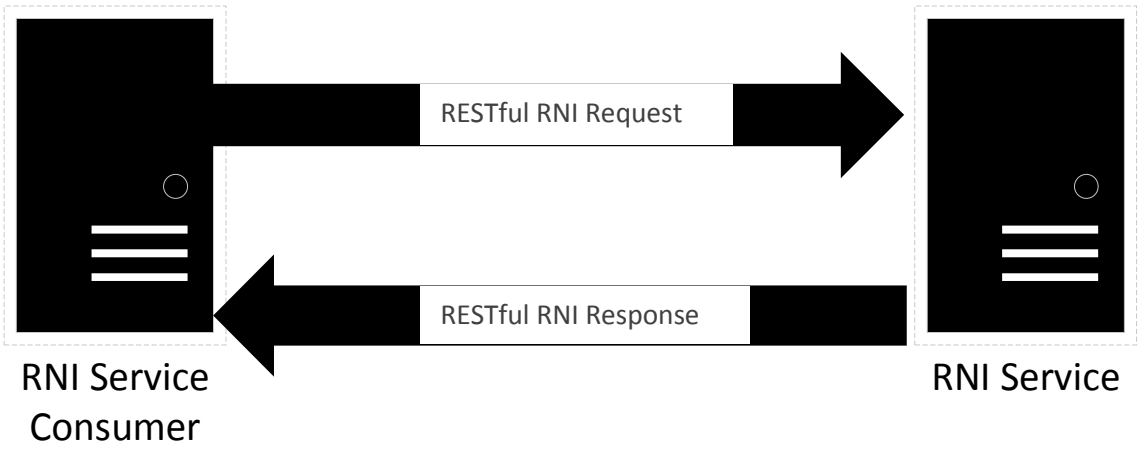


Figure 2: RESTful RNI API

A MEC application instance which is running at a MEC host can leverage the RNIS to optimize performance or to provide new types of services based on up-to-date radio network information. A typical example is Virtual Reality/ Augmented Reality (VR/AR) applications that can adjust TCP congestion windows and video formats that are adaptive to the air interface throughput. The key point of optimization of those services is the accuracy of radio network information which leads to a real-time interaction requirement between CRAN and MEC.

5G CRAN supports centralizing the upper layers of the radio stacks at a Central Unit (CU), while distributing the lower layers into Distributed Units (DUs). Different protocol stack functional splits with ideal/non-ideal fronthaul are also supported between the CU and the DUs. This flexible architecture of 5G CRAN ensures that the radio information from not only long-term Radio Resource Management (RRM) but also short term RRM could be pulled out from the CU directly. With the enhanced real-time 5G RAN L3/L2/ L1 status information (e.g., beam info, Sounding Reference Signal (SRS) measurements), RNIS will be widely used in many scenarios, such as indoor positioning navigation in shopping malls, rapid RAT selection in V2X, etc.

Since MEC is deployed much closer to the RAN, the best way for RNIS to get radio information is direct interaction with CRAN rather than taking a long route through a network function of the core network (e.g., Network Exposure Function). Especially when co-located with CRAN, MEC may share the same NFVI with CRAN. The interaction between MEC and CRAN could be performed via internal interfaces, which could improve communication efficiency and support real time applications effectively by leveraging the performance advantage of the shared infrastructure.

Figure 3 shows how the RNIS could obtain radio information from co-located CRAN with a shared NFVI. F1 is an interface between the CU and the DU defined by 3GPP. Mp1 is a reference point defined by ETSI MEC for exposing the RNIS to authorized applications. The interaction between CRAN and MEC calls for further investigation.

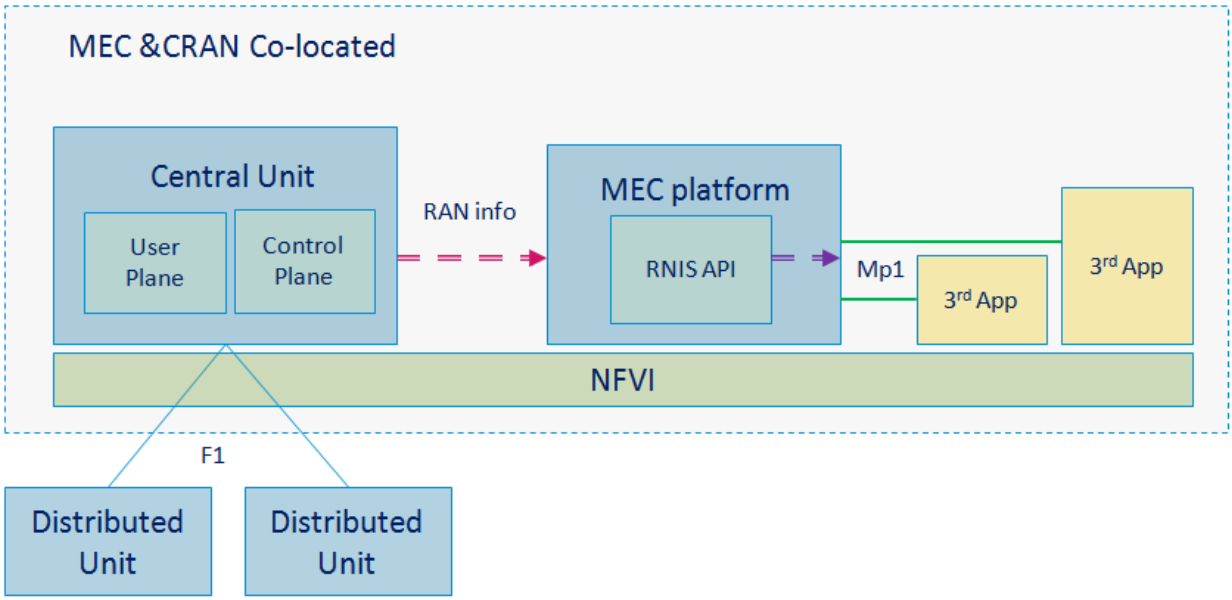


Figure 3: MEC obtains RAN information directly from CRAN

Location API

The Location Service is a service that provides location related information to MEC platforms and MEC applications, e.g., AP/eNodeB location, UE location, UEs in a specific area, or notifications of UE(s) entering an area. It leverages the Zonal Presence service described by the Small Cell Forum [11], [12] and is accessible through the API defined in the Open Mobile Alliance (OMA) specification "RESTful Network API for Zonal Presence" [13]. In the context of MEC, the Anonymous Customer Reference [14] may address particular user categories via the API and may also discover how the 3GPP Cell Identifiers can be mapped to the Access Point identifier of the OMA API.

With respect to a MEC deployment at the edge of the CRAN domain, what is of interest are the potential sources of UE location of which there are two: firstly, from the UE itself, e.g., a Global Navigation Satellite System (GNSS) fix, which may benefit from network originated assistance information such as A-GPS data; and secondly network based location with low or high precision. An example of low precision would be serving cell alone, or, higher precision could be provided using techniques such as network based trilateration and triangulation, which may be augmented with UE originated information such as time difference measurements as requested by the network.

The UE may provide GNSS-like fixes via higher layer protocols, for instance at the application layer and particularly in an Over the Top (OTT) context, but that would not be directly available to the RAN. It may also be specifically requested to provide it via Radio Resource Control (RRC) signalling through features such as Radio Link Failure (RLF) and Minimization of Drive Tests (MDT), in which case such information would be available in the RAN and, therefore, could be exposed by the RAN and used as a source of information for the Location API. With respect to low precision information, such as the serving cell, the RAN is also well placed to make that information available to the MEC domain, where it is worth noting only the serving eNodeB is generally exposed to the Evolved Packet Core (EPC). In the absence of a GNSS fix from the UE, the network may offer higher precision location estimates through the location services architecture, but then the key network elements are typically located outside of the RAN. Therefore, an

alternative may be for the MEC platform or MEC application to generate the information necessary for the Location API itself based on the raw information available to it, e.g., that obtained from RNIS.

Focusing on security and anonymity challenges , the Location API responds to a number of issues that need to be resolved in the MEC environment. Such privacy and security-related issues are expected to be prominent when MEC hosts are deployed across the CRAN.

- Question 1: Is this safe? Can a wrongdoer hack this?

Location data is privacy sensitive. Knowing where people are, on the other hand, is important for a business, e.g., if a merchant knows someone is currently in front of his/ her store (or a competitor's store), this is of extra business value. In order for this information availability to be accepted in the wide public, there need to be mechanisms within the CRAN to keep this information safe.

- Question 2: Can it be done anonymously?

This can be a clear, acceptable answer to the public's concerns – keep the location data anonymized, or the location expressed in a way that ensures a user can select to remain anonymous.

In order to address these challenges, data should be shared between applications using a generic, controlled API to avoid extraction and unintended leakage of private information. This also responds to further privacy concerns by avoiding a growing database of personal information in multiple locations.

Also importantly, a multi-tenant security model, potentially even by separating processes not only through virtualization but even running in separate physical domains, enables extended privacy protection by avoiding extraction of data from processes sharing the same physical domain. Such security-driven multi-tenancy needs to be investigated when it comes to a CRAN deployment.

UE Identity API

The UE Identity feature is provided to allow authorized MEC application instances to invoke UE specific traffic rules within the MEC platform. Each UE is identified by a unique "tag" which is provided to the application. The tag acts as an intermediary identifier between a UE's mobile IP 5-tuple (where the mobile identity may be used as a further intermediary identifier, i.e., the International Mobile Subscriber Identity / IMSI) and its external identifier, e.g., its enterprise identity (Figure 4). In this manner, masking is achieved between the MNO's identification system and that of any external network. The MEC platform is provided with the UE to tag mapping information, but how that mapping is realized is currently outside the scope of the MEC specifications. In order to trigger the UE traffic rules, the MEC application instance registers the relevant tag with the MEC platform via the UE Identity API. Following successful registration, the MEC platform then activates the corresponding traffic rule(s) linked to the tag. Later, if the application instance no longer wishes to use the traffic rule for that user, it may de-register the tag by invoking the de-registration procedure.

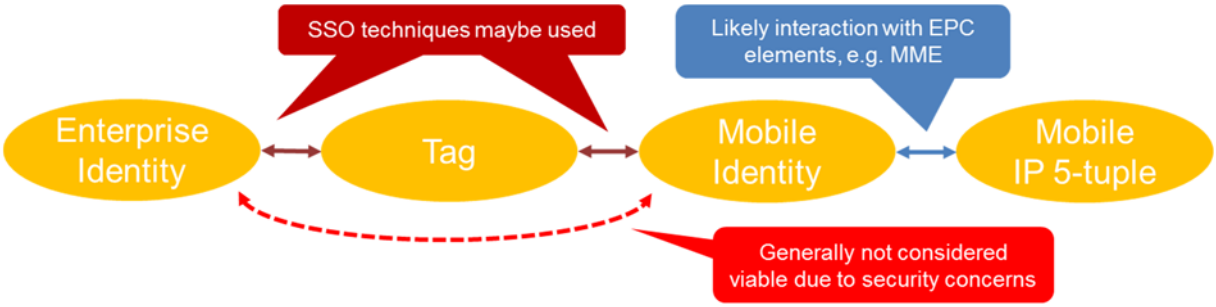


Figure 4 Identity mapping using the Tag

A significant factor with respect to a CRAN approach is that the mobile identity (IMSI) would not inherently be available in the CRAN domain. This is certainly the case for LTE if it is assumed that the northbound interface from the CRAN is S1, or the N2/N3 interface with regards to 5G NG-RAN. The IMSI is generally only exposed within the LTE EPC and communicated infrequently by the UE. The implication is that a secondary means is required to provide the information necessary to link the traffic flows supported within the CRAN with specific UE identities. The CRAN would only be aware of temporary identifiers such as the System Architecture Evolution Temporary Mobile Subscriber Identity (S-TMSI), associated Radio & S1 Bearer IDs and S1-Application Protocol (AP) UE IDs. Therefore, the MEC Platform would need to be provided with the information to link a specific tag with the appropriate temporary identifiers. A solution would be to deploy probe based agents in the EPC. Specifically, the role of the agent would be to extract IMSI/IMEI identifiers with their associated temporary identities for each connection session by probing key interfaces within the EPC. The agent would then provide this IMSI based Customer Experience Management (CEM) data feed with the necessary paring information to the MEC platform. Using this information, the MEC platform could then fulfil the UE specific traffic rules as required.

With such a solution in place, a clear advantage with a CRAN type approach is the centralization it affords. This allows traffic rules to be applied at the edge of the RAN across a wide area and for many UEs, rather than at an intermediate point along the S1 interface that would require S1 de-encapsulation and re-encapsulation.

Bandwidth Management API

The Bandwidth Management service (BWMS) API is another means of supporting the use cases and requirements relevant to MEC technology. The mission of this service API lies in the effective and timely satisfaction of bandwidth requirements tailored towards a single MEC application instance, or multiple sessions of the same application. Such bandwidth requirements may refer to the bandwidth required to support the instantiated application (or session), to the bandwidth priority of the application, or to both. Given that different MEC applications concurrently instantiated at a MEC host may have different bandwidth requirements, or the same bandwidth requirement may characterize MEC applications tailored to UEs located at different distances from the edge of the RAN, the BWMS API can aggregate all requests and share the available resources in an optimal and fair manner.

In a non-CRAN deployment involving MEC hosts located at the edge, an advantage of enabling and exposing the BWMS API consists in making “local” decisions on bandwidth allocation, which enhance performance and are reached in a timely manner due to proximity. However, with the lack of any centralized coordination provided by a central entity of the equivalent CRAN, only intra-cell UEs are expected to benefit from decisions upon bandwidth allocation. This means that, for example, out-of-cell interference might affect the performance of some of the MEC applications.

On the other hand, consider a CRAN deployment without any MEC hosts existing at the edge (and, consequently, the non-availability of the BWMS API). Centralized bandwidth allocation is expected to have a positive impact on performance due to coordination, albeit largely depending on the characteristics of the fronthaul connection. In other words, this means that the quality of the fronthaul interface may negatively influence the decisions upon bandwidth management.

Motivated by the above arguments, one would expect that a joint CRAN/ MEC deployment, together with the enablement of the BWMS API, will address the coordination efficiency / timeliness trade-off, by properly exposing network-wise bandwidth information to the MEC host. Such information could be considered by the BWMS API with the aim of boosting the performance of the MEC application instances.

Conclusion

In this paper, we present a case that MEC and CRAN are highly complementary technologies. Each makes the economics of deploying the other significantly more attractive. However, such co-location requires solving a number of technical challenges as well as figuring out how to maximize the revenue generating potential of co-location.

This paper is meant as an introductory guide to the industry as it works on resolving these challenges. We highlight the importance of understanding the deployment scenarios and use-cases and the architectural trade-offs that these may impose. We then provide a summary of key technical challenges and high-level overview of solution approaches. The last part of the paper provides an overview of how ETSI MEC services, such as Radio Network Information Service, can be used as revenue generating tools in a CRAN-MEC environment.

This paper is but a brief, and, thus a high-level overview. Our hope, however, is that it is a useful starting point on the journey towards effective and highly profitable joint CRAN/MEC deployments.

Abbreviations

AI	Artificial Intelligence
AF	Application Function
API	Application Programmers Interface
AR	Augmented Reality
AWS	Amazon Web Services
BBU	Baseband Unit
BWMS	Bandwidth Management Service
CAPEX	Capital Expenditure
CN	Core Network
CO	Central Office
COTS	Commercial-of-the-Shelf
CRAN	Cloud RAN
CU	Central Unit
DC	Data Centre
DNS	Domain Name System
DU	Distributed Unit
eMBB	enhanced Mobile Broadband
EPC	Evolved Packet Core
GNSS	Global Navigation Satellite System
GPP	General Purpose Processor
GS	Group Specification
IaaS	Infrastructure-as-a-Service
IMSI	International Mobile Subscriber Identity
ISG	Industry Specification Group
LEA	Law Enforcement Agency
LI	Lawful Intercept
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
mMTC	massive Machine Type Communication
MNO	Mobile Network Operator
NFV	Network Function Virtualization
NFVI	NFV Infrastructure
OAM	Operation and Management
OPEX	Operational Expenditure
OS	Operating System
OTT	Over-the-Top
PaaS	Platform-as-a-Service
PLMN	Public Land Mobile Network
PSTN	Public Switched Telephone Network

PoP	Point-of-Presence
QoE	Quality of Experience
QoS	Quality of Service
RAB	Radio Access Bearer
RAN	Radio Access Network
RD	Retained Data
RNI	Radio Network Information
RNIS	RNI Service
RRC	Radio Resource Control
RRH	Remote Radio Head
SDO	Standard Developing Organization
SRS	Sounding Reference Signal
TCO	Total Cost of Ownership
URLLC	Ultra Reliability and Low Latency Communication
V2X	Vehicular-to-Everything (as in car-based communication)
vDC	virtual DC
VNF	Virtualized Network Function
VR	Virtual Reality

References

[1] ITU-R Recommendation M.2083-0, "IMT Vision – Framework and overall objectives of the future development of IMT for 2020 and beyond", Sep. 2015. https://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.2083-0-201509-I!!PDF-E.pdf

[2] Huawei White Paper, "5G Unlocks A World of Opportunities – Top Ten 5G Use cases," Huawei, Nov 2017. http://www.huawei.com/us/industry-insights/mbb-2020/trends-insights/5g-unlocks-a-world-of-opportunities

[3] ETSI GS MEC 010-1, "Mobile Edge Computing (MEC); Mobile Edge Management; Part 1: System, host and platform management" (2017-10). http://www.etsi.org/deliver/etsi_gs/MEC/001_099/01001/01.01.01_60/gs_MEC01001v010101p.pdf

[4] ETSI GS MEC 010-2, "Mobile Edge Computing (MEC); Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management" (2017-07). http://www.etsi.org/deliver/etsi_gs/MEC/001_099/01002/01.01.01_60/gs_MEC01002v010101p.pdf

[5] ETSI GR MEC 017 V1.1.1, "Mobile Edge Computing (MEC); Deployment of Mobile Edge Computing in an NFV environment" (2018-02) http://www.etsi.org/deliver/etsi_gr/MEC/001_099/017/01.01.01_60/gr_MEC017v010101p.pdf

[6] ETSI GS MEC 012 V1.1.1, "Mobile Edge Computing (MEC); Radio Network Information API" (2017-07) http://www.etsi.org/deliver/etsi_gs/MEC/001_099/012/01.01.01_60/gs_MEC012v010101p.pdf

[7] ETSI GS MEC 013 V1.1.1, "Mobile Edge Computing (MEC); Location API" (2017-07) http://www.etsi.org/deliver/etsi_gs/MEC/001_099/013/01.01.01_60/gs_MEC013v010101p.pdf

[8] ETSI GS MEC 014 V1.1.1, "Mobile Edge Computing (MEC); UE Identity API" (2018-02) http://www.etsi.org/deliver/etsi_gs/MEC/001_099/014/01.01.01_60/gs_MEC014v010101p.pdf

[9] ETSI GS MEC 015 V1.1.1, "Mobile Edge Computing (MEC); Bandwidth Management API" (2017-10) http://www.etsi.org/deliver/etsi_gs/MEC/001_099/015/01.01.01_60/gs_MEC015v010101p.pdf

[10] OpenAPI Specification https://github.com/OAI/OpenAPI-SpecificationOAS

[11] SCF 084.07.01: "Small cell zone services - RESTful bindings" https://scf.io/en/documents/084_-_Small_Cell_Zone_services_RESTful_Bindings.php

[12] SCF 152.07.01: "Small cell services API" http://scf.io/en/documents/152_-_Small_cell_services_API.php

[13] RESTful Network API for Zonal Presence V1.0 (2016) http://www.openmobilealliance.org/release/REST_NetAPI_ZonalPresence/V1_0-20160308-C/OMA-TS-REST_NetAPI_ZonalPresence-V1_0-20160308-C.pdf

[14] RESTful Network API for Anonymous Customer Reference Management V1.0 (2013) http://www.openmobilealliance.org/release/REST_NetAPI_ACR/V1_0-20131224-A/OMA-TS-REST_NetAPI_ACR-V1_0-20131224-A.pdf

[15] eCPRI Transport Network V1.0, “Common Public Radio Interface: Requirements for the eCPRI Transport Network,” Oct. 2017. http://www.cpri.info/downloads/Requirements_for_the_eCPRI_Transport_Network_V1_0_2017_10_24.pdf

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

ETSI White Paper No. 24

MEC Deployments in 4G and Evolution Towards 5G

First edition – February 2018

ISBN No. 979-10-92620-18-4

Authors:

Fabio Giust, Gianluca Verin, Kiril Antevski, Joey Chou, Yonggang Fang, Walter Featherstone, Francisco Fontes, Danny Frydman, Alice Li, Antonio Manzalini, Debashish Purkayastha, Dario Sabella, Christof Wehner, Kuo-Wei Wen, Zheng Zhou

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Fabio Giust
NEC - Editor

Gianluca Verin
Athonet - Editor

Kiril Antevski
UC3M

Joey Chou
Intel

Yonggang Fang
ZTE

Walter Featherstone
Viavi Solutions

Francisco Fontes
Altice Labs

Danny Frydman
Saguna

Alice Li
Vodafone

Antonio Manzalini
TIM

Debashish Purkayastha
InterDigital

Dario Sabella
Intel

Christof Wehner
Artesyn

Kuo-Wei Wen
ITRI

Zheng Zhou
Huawei

Contents

Introduction

Multi-access Edge Computing is regarded as a key technology to bring application-oriented capabilities into the heart of a carrier's network, in order to explore a wide range of new use cases, especially those with low latency requirements. When it comes to deploying MEC, there are many potential scenarios where MEC can fit in, and – as the name clearly spells out – these are not limited to 4G or 5G at all! As a universal access technology, MEC offers itself to any application that has locality requirements like a shopping mall or a sports arena, or wherever low latency is required such as 5G V2X or autonomous vehicle applications.

Nevertheless, starting from the fact that the MEC's original target was the mobile network, when it comes to its deployment, MEC is often considered as a 5G-only feature. However, 4G is expected to still be successful in the years to come, thus a large part of the industry is working towards running MEC in existing 4G networks. In fact, the MEC reference architecture, defined in ETSI GS MEC 003 [1], is agnostic to the mobile network evolution, so that a MEC host deployed in a 4G network can be reused to support 5G services as well.

Therefore, understanding the impact of deploying an ETSI MEC system into current 4G and future 5G systems is crucial for mobile network operators (MNOs) in order to carefully plan their network upgrades. This way, MEC can be not only a technology ready for 4G, but also a driver to motivate 5G adoption, as it can allow operators to retain the investment made in 4G deployment. Indeed, from a mobile evolution perspective, products based on current MEC specifications can be smoothly migrated to support 5G networks through software update. This way, flexibility in the deployment architecture allows planning for the introduction of MEC services as the milestone to build the edge cloud, which is key for the success of 5G services such as URLLC (Ultra Reliable Low Latency Communications).

In light of the considerations above, the purpose of this white paper is to show the compatibility of an ETSI MEC system with 3GPP 4G and 5G architectures, aiming at:

- shedding some light on the potential deployment options available for operational 4G systems;
- providing a technical insight of MEC operations under such scenarios;
- showing how the creation of the mobile edge infrastructure in 4G can pave the way for 5G deployment and therefore protect investments and smoothly transit to future and more advanced service offerings.

The present document will first showcase different options to cast the MEC system into a running 4G system, maintaining backward compatibility with the 3GPP-specified architecture. In other words, such options explore how the "MEC box" can be drawn into the 4G system architecture, showing the challenges associated to each choice.

Moreover, a section devoted to the transition to 5G will demonstrate how and why deploying MEC in 4G can accelerate network transformation, leveraging on compliance to 3GPP standards and use of the cloud computing paradigm to bring future-proof added value to service providers.

Deploying MEC in 4G networks: scenarios and challenges

As per the GS MEC 011 [2] specification, a key baseline functionality of the MEC platform is to route IP packets to MEC applications which are meant to handle the traffic in the following different ways:

- In **Breakout** mode, the session connection is redirected to a MEC application which is either hosted locally on the MEC platform or on a remote server. Typical breakout applications include local CDN, gaming and media content services, and enterprise LAN.
- In **In-line** mode, the session connectivity is maintained with the original (Internet) server, while all traffic traverses the MEC application. In-line MEC applications include transparent content caching and security applications.
- In **Tap** mode, specified traffic is duplicated and forwarded to the tap MEC application, for example, when deploying virtual network probes or security applications.
- In **Independent** mode, no traffic offloading function is needed, but still the MEC application is registered in the MEC platform and will receive other MEC services, such as DNS, Radio Network Information Service (RNIS), etc.

Steering traffic to/from MEC applications is achieved by configuring the MEC's local DNS and the MEC host's data plane accordingly. From the list above, it appears straightforward that the implementation-specific details of the data plane within the MEC host (as per the MEC architecture in GS MEC 003 [3]) and the MEC platform, which is meant to program the data plane through Mp2 interface, are impacted by the point where the MEC host is installed in the 4G architecture. Many choices are possible, but all in all they can be condensed down into some base scenarios discussed in the following sections.

Bump in the wire

The expression "bump in the wire" encompasses all the scenarios in which the MEC platform installation point ranges in locations between the base station itself and the mobile core network. These options were first proposed in the MEC Introductory White Paper [1] and reproduced in Figure 1.

When the eNB implementation bundles the MEC platform into a single implementation, this latter is able on the one hand to route plain IP packets to/from the MEC applications (i.e., local switching mode), and on the other hand to route GTP-encapsulated packets to/from the Serving Gateway (SGW) for regular traffic as per the operator-configured Packet Data Networks (PDNs - i.e., the legacy S1-U mode). This deployment is very convenient e.g., in enterprise scenarios to allow intranet traffic to breakout to local services (similar to Local IP Access - LIPA), and also when MEC is co-located with a CRAN deployment (see the ETSI white paper "Cloud RAN and MEC: a Perfect Pairing" [4].)

In all the other locations, either in proximity of the radio node or at an aggregation point, the MEC platform sits on the S1 interface of the 4G system architecture. In this scenario, the MEC host's data plane has to process user traffic encapsulated in GTP-U packets, thus requiring the appropriate handling of these tunnels. This non-trivial operation poses some challenges, as a portion of the data may be generated or manipulated internally in the MEC hosts or may come from a local breakout, without passing through the core of the network. For this traffic, a dedicated solution may be required (e.g., the MEC GW in Figure 1) to handle operations such as lawful interception and charging. This solution can support CUPS, which ensures a 3GPP-compliant solution (see sections below). Also, in this solution, low

latency is supported by installing the MEC platform all the way down to the eNB, or in locations that ensure minimal latency. Additionally, it offers the capability to steer traffic on a per session and/or packet granularity, with flexible filtering support.

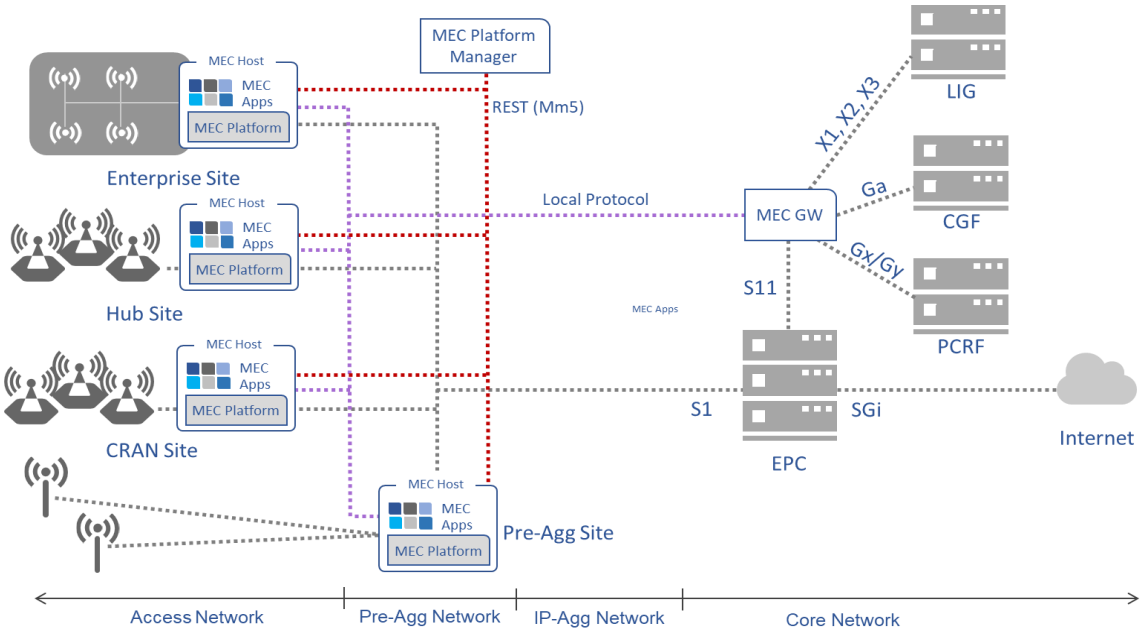


Figure 1: MEC deployment using the "Bump in the wire" approach.

Distributed EPC

Unlike the “bump in the wire” approach, in this deployment the MEC host logically includes all or part of the 3GPP Evolved Packet Core (EPC) components, as specified in the 4G system architecture in ETSI TS 123.401 [5], and the MEC data plane sits on the SGi interface. By doing so, in order to steer U-plane traffic towards the MEC system, two elements, the local DNS of MEC and the PDN Gateway (PGW) of a distributed EPC, play critical roles. In fact, as the UE subscribes to the distributed EPC co-located with the MEC host, the PGW thereupon terminates the PDN connection and assigns the IP address and local DNS information to resolve the MEC applications’ IP address. This scenario requires less changes to the operator’s network as standard 3GPP entities and interfaces are leveraged for operations such as session management, charging, etc.

This type of deployment can well serve Mission Critical Push to Talk (MCPTT), and M2M communications, where the communication with the operator’s core site is optional (see for example the upper diagram in Figure 2). In this case, the Home Subscriber Server (HSS) is co-located with the EPC as well, and there is no need for a working backhaul to keep the local service running. This type of deployment is typically used by first responders, public safety, and mission critical industrial sites.

In some other cases, the HSS is unique and centrally managed by the operator at the core site and the operator’s core site PGW can be used for some selected APNs (e.g. IMS or roaming). This allows the local management of the entire subscriber database and the use of the local EPC in the MEC to offload the entire APN traffic. Additionally, the distributed EPC offers the ability to deliver exactly the QoS and

configurability features that, e.g. an enterprise customer requests from the particular network service purchased (see the lower diagram in Figure 2).

The MEC applications can be co-located with the evolved packet core (EPC) functions in the same MEC host. This option can reduce costs as the EPC and its components can run e.g. as Virtual Network Functions (VNFs) on the same Network Functions Virtualisation (NFV) platform with the MEC components in order to improve scalability and better utilize network resources (see the example in Figure 3).

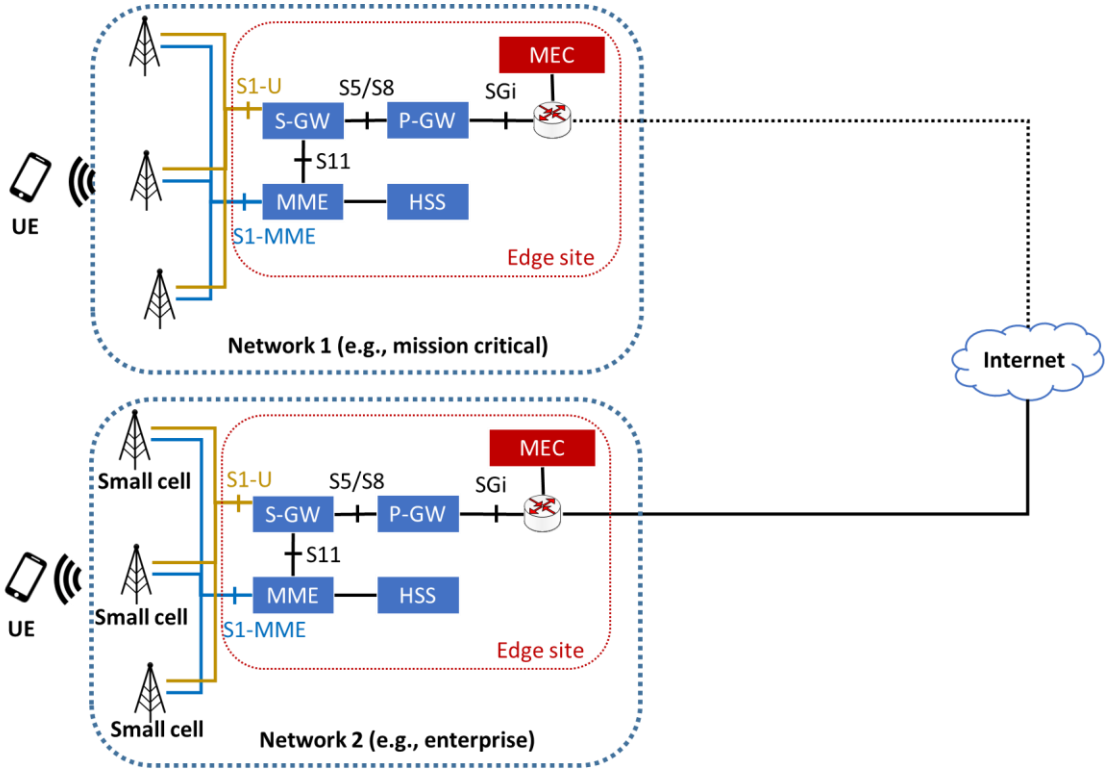


Figure 2: MEC deployment with distributed EPC

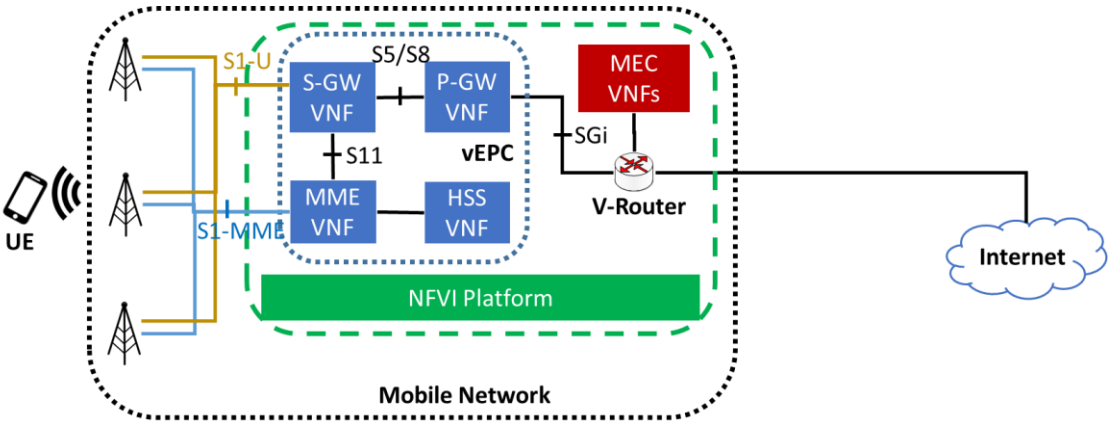


Figure 3: MEC deployment with EPC and MEC application on the same NFV platform (same MEC host).

Distributed S/PGW

The distributed S/PGW deployment option is similar to the previous one, except that only SGW and PGW entities are deployed at the edge site, whereas the control plane functions such as the Mobility Management Entity (MME) and HSS are located at the operator's core site. Still, the MEC host's data plane connects to the PGW over the SGi interface.

Similarly to the previous option with the whole distributed EPC, the SGW and PGW can also run as VNFs together with the MEC application on the NFV platform as part of the same MEC host. The local SGW selection is performed by the central MME according to the 3GPP standard DNS procedures and based on the Tracking Area Code (TAC) of the radio where the UE attaches to. This architecture allows offloading the traffic based on the APN, which means, for example, that the IMS for VoLTE APN and roaming APNs may not be offloaded.

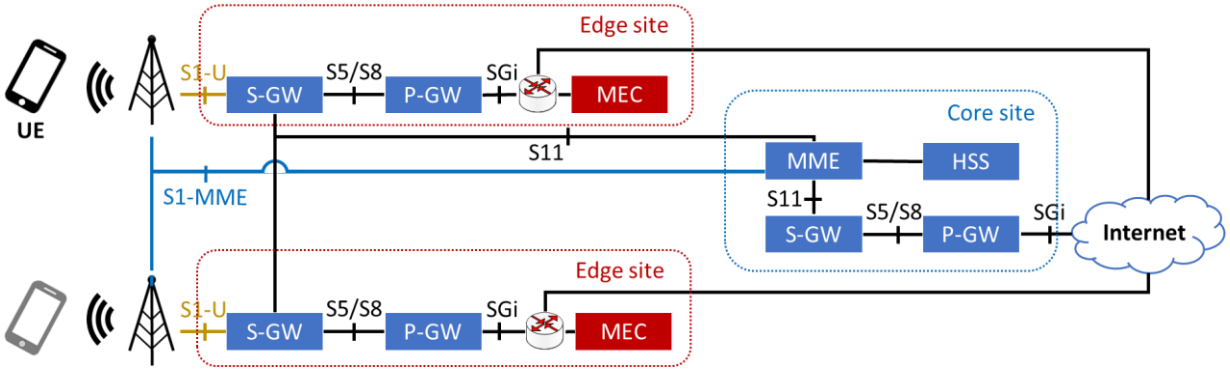


Figure 4: S-GW and P-GW MEC deployment

The diagram above shows the deployment with the SGW and PGW co-located at the network edge, which requires the operator to extend the S5 interface to the MEC site. This type of deployment allows the operator to retain full control over the MME.

Distributed SGW with Local Breakout (SGW-LBO)

Local breakout at the SGWs is a new architecture for MEC that originates from operators' desire to have a greater control on the granularity of the traffic that needs to be steered. This principle is dictated by the need to have the users able to reach both the MEC applications and the operator's core site application in a selective manner over the same APN.

With the Distributed SGW deployment, one of the optional MEC deployment scenarios is to co-locate MEC hosts with the SGW. Both the SGW-LBO and the MEC application may be hosted as VNFs in the same MEC platform. The following figure describes co-locating MEC hosts with the SGW in a mobile network where the MEC system and the distributed SGW are co-located at the edge.

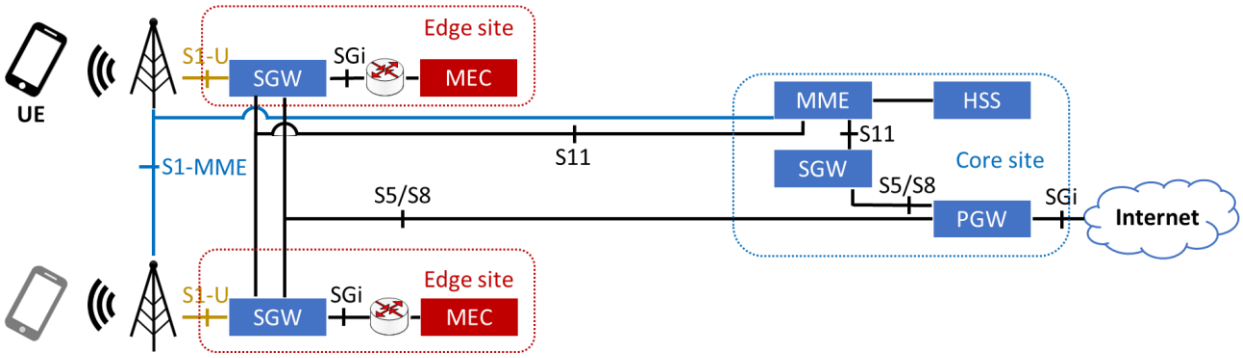


Figure 5: SGW-LBO MEC deployment

The traffic steering uses the SGi - Local Break Out interface which supports traffic separation and allows the same level of security as the operator expects from a 3GPP-compliant solution. This solution allows the operator to specify traffic filters similar to the uplink classifiers in 5G, which are used for traffic steering. This architecture also supports MEC host mobility, extension to the edge of CDN, push applications that requires paging and ultra-low latency use cases.

The SGW selection process performed by MMEs is according to the 3GPP standard and based on the geographical location of UEs (Tracking Areas) as provisioned in the operator's DNS.

The SGW-LBO offers the possibility to steer traffic based on any operator-chosen combination of the policy sets, such as APN and user identifier, packet's 5-tuple, and other IP level parameters including IP version and DSCP marking.

Control/User Plane Separation (CUPS)

The deployment options above which distribute the EPC gateways at the edge, either co-located with or within the MEC host, can also be built using the CUPS paradigm standardized in 3GPP Release 14 and have the new User Plane built in the MEC host.

Local User Plane (UP) distribution allows the use of the CUPS architecture to locally steer the traffic. SGW-C and PGW-C are the end points that populates the UP routing tables.

Challenges in the different approaches

From the deployment scenarios outlined above, a clear distinction emerges of two major categories, depending on whether the MEC host leverages the EPC packet gateways' functionalities or not. This section examines the impact of the different types of deployment scenario with respect to session and mobility management, security, charging and Lawful Interception. As expected, approaches that use standard 3GPP NFs to support MEC show the least impact.

Session management

In the bump in the wire scenario, MEC is located on the S1-U reference point. The eNB and SGW are not MEC-aware as MEC components are not involved in the standard 3GPP procedures of session management, including PDN connection setup, deletion and paging. However, it is necessary for MEC to get the UE context for the right traffic routing, which makes it more challenging. There are at least two feasible approaches to manage the UE context for MEC:

1. User plane packet inspection: MEC creates the UE context according to the S1-U tunnel IP addresses and Tunnel Endpoint Identifiers (TEID-Us) learned from the user plane packets (see also

the section below “Identifying specific subscribers at the MEC platform”). For the traffic that needs to be offloaded, MEC routes specific packets to specific applications by a traffic offload function. For traffic flows that do not need to be offloaded, MEC behaves as a transparent device. In addition, a dedicated yet not standard mechanism is necessary to trigger paging from the MEC application, e.g., for push notifications.

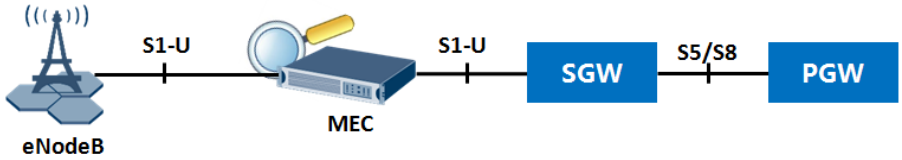


Figure 6: User plane packets inspection

2. Controlled by the PGW: The enhanced PGW controls the session management of MEC to create, update, delete the UE context and delivers the charging characteristics/LI information. Although a new reference point needs to be created between PGW and MEC, charging and LI are supported and it can be easily upgraded to the CUPS deployment mode and then evolved smoothly to 5G. The reference point between PGW and MEC will be Sx in CUPS deployment mode and N4 in 5G.

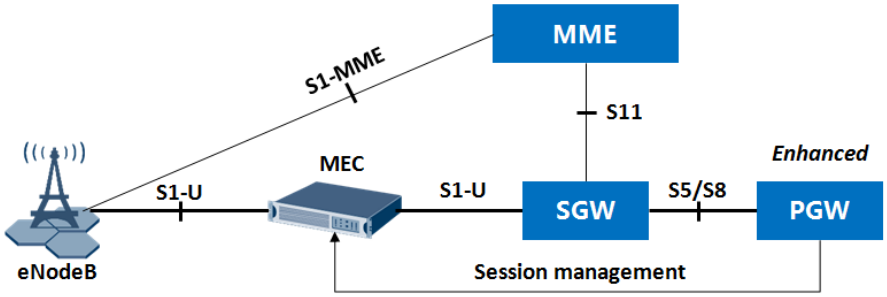


Figure 7: PGW-Controlled MEC

In the other MEC deployment options which make use of **EPC co-location,** there is much less impact on session management, as it is handled by the EPC functions installed along with the MEC host. In particular we observe the following

- EPC MEC, SGW+PGW MEC: Session management is not impacted, even for inter-MEC handover since the standard 3GPP procedures are used to keep the original PGW as anchor. This assures session continuity as well as paging idle UEs. Application level mobility is achieved by reassigning the IP address to the user or enforcing a breakout policy into the target SGW. Charging and lawful interception are supported natively by the solution.
- SGW-LBO MEC: The connectivity to the standard PGW assures the creation and deletion of the UE context similarly to the approach above.
- CUPS MEC: All the MEC options can incorporate the CUPS solution which requires a UP capable of performing traffic offload in order to steer traffic to/from the MEC applications.

Mobility management

Mobility is concerned with service continuity when the UE is moving intra or inter MEC. MEC needs to be aware of the handover of the UE in the underlying network and updates the UE context to keep the service continuity. Two scenarios appear relevant:

1. The UE moves from one eNodeB to another, but is still in the coverage of the same serving MEC host, i.e., intra MEC mobility. The MEC system should be able to route the traffic to the UE via the correct eNodeB and tunnel.
2. The UE moves out of the coverage area of the source MEC host to enter the coverage area of target MEC, i.e. inter MEC mobility or MEC Handover. This scenario may result in interruption of service to the UE. In order to provide service continuity to the UE, the MEC system needs to relocate the service to the UE from the source to the target MEC.

Depending on the selected solution, MEC Handover is handled in different ways. In the bump in the wire approach, mobility is not natively supported. One solution is to have the MEC implementation to detect the UE handover and act accordingly. An alternative solution is to update the UE context in MEC by the PGW as described in the session management part.

In the EPC MEC, SGW + PGW MEC, and CUPS MEC, the MEC handover is supported using 3GPP standard S1 Handover with SGW relocation by maintaining the original PGW as anchor. The same considerations apply for the SGW-LBO MEC deployment. In the latter case, the target SGW enforces the same policy towards the local MEC application. It is the MEC application's responsibility to synchronize at application level and maintain the session in the case of a stateful application.

Lawful interception

Lawful Interception (LI) and Retained Data (RD) play a crucial role in helping law enforcement agencies to combat terrorism and serious criminal activity. Providers of public telecommunications networks and services are legally required to make available to law enforcement authorities information from their retained data which is necessary for the authorities to be able to monitor telecommunications traffic as part of criminal investigations.

Typically, this functionality is supported at nodes within the core network. However, traffic carried from the UE to an application at the network edge is currently designed to avoid the core, and hence would avoid the usual intercept points. In the context of MEC, it is recommended that LI and RD collection functions are implemented at the edge of the network, alongside or as part of the functionality being intercepted. Any edge node including LI/RD collection features must support strong physical security requirements similar to core network sites. In these regards, ETSI MEC is collecting informative and normative aspects in GS MEC 026 (work in progress), as this has strong impact on the MEC entities especially under the bump in the wire approach.

On the contrary, the solutions that include an EPC gateway, such as EPC MEC, SGW+PGW MEC, SGW-LBO MEC, and CUPS MEC are compliant with LI requirements as they natively support the usage of X1, X2, X3 interfaces towards the operators' Mediation Device, responsible to connect to the agency and transfer the requested data for the target.

Security

MEC offers an IT service environment and cloud-computing capabilities for hosting applications at the edge of the mobile network. As illustrated above in some deployment models the MEC applications run on the same physical platforms as some network functions. The third party applications are not controlled by the operator directly, so there are risks of these applications exhausting resources that are needed by the network functions. There are also risks of poorly designed applications allowing hackers to infiltrate the platform and hence affect the network functions running on the platform – or even of malicious

applications doing the same thing themselves. One solution to tackle these issues is to run both the MEC applications and the network function(s) in robustly segregated virtual machines, providing an assurance of confidentiality for sensitive data/information between VMs running on the same physical platform, and between a hypervisor and the host operating system. In addition, there is an opportunity for the MEC system to provide security/assurance services for the hosted applications. One example is to perform integrity assurance checks on applications at installation and upgrade, or after a server restart. Another is to expose security services APIs to sufficiently trusted third party MEC applications, e.g. for user identification.

Another option to enforce security is to allow deployment types that run applications on segregated hardware. This is particularly relevant for CDN, which usually have strict hardware requirements for copyright and privacy issues. Security is also enforced with an appropriate network design of the edge site where the MEC platform is connected with the use of L2/L3 traffic separation and firewalls.

Moreover, IPSec is envisaged to protect packets on the S1 interface. This creates additional challenges for bump in the wire approaches, which may require special, not yet standardized network design and more complex approval process by operators.

One last observation concerns the Distributed EPC deployment when including the HSS at the edge: this scenario requires special care on handling the confidentiality of subscribers' data and extending standard 3GPP core network interfaces at the edge of the network.

Charging

MEC needs to support off-line and on-line charging:

- Off-line charging: MEC periodically collects and reports the data records to the off-line charging function for aggregation and correlation. Billing systems use the aggregated/correlated event records to charge the consumer at the end of the billing cycle.
- On-line charging: Upon the first chargeable event of a consumer, MEC triggers an on-line charging request towards the on-line charging function to get a quota granted. When the allocated quota is almost fully used, MEC reports the usage of the resource and requests for an additional quota from the on-line charging function. The charging function may allocate a new quota or deny it. In case of denial, MEC will reject the resource usage request.

When it comes to charging, the bump in the wire approach natively supports the case of traffic passing through MEC application and then further to the CN, for which charging is taken care of by the 3GPP functions. Conversely, for traffic that is either terminated at MEC applications or breaking out to an external network, alternative solutions need to be considered to provide the necessary charging support. For instance, an alternative, yet not standardized solution needs the cooperation of MEC and CN functionalities, whereby MEC reports charging data to the PGW (as in Figure 7), or to the MEC Gateway based on the charging policies from the PGW (see Figure 1). Then the PGW aggregates and reports them to the billing system.

The other deployment options leverage the EPC data plane functionalities, so that both offline and online charging are supported natively as in the standard EPC, for all packets terminated locally or forwarded to external APNs configured in the core site's EPC for home and roaming traffic. This applies also to solutions like SGW + PGW MEC, SGW-LBO MEC, and CUPS MEC. In the latter case, the User Plane component employed for traffic offloading is able to forward usage statistics to the SGW-C and PGW-C according to

the standards. However the PGW-C needs to be customized to support both the PGW-U and the customized MEC-UP.

Identifying specific subscribers at the MEC platform

Traffic routing is part of the MEC platform's (MEP) essential functionality [1] and is enabled by applying configurable traffic rules. The functionality supports use cases such as breakout of encrypted user-plane traffic to a local network (e.g. enterprise network) by the MEP. Such traffic routing enables e.g. employees using authorized smartphones and tablet PCs to enjoy a fast broadband connection directly to their enterprise LAN, rather than such traffic having to traverse the mobile core network via a latency-inducing transport network. A key aspect of the routing is the identification of the packets to be filtered, where several filtering options are provided, including source/destination IP, port and tunnel addresses.

In order to filter based on UE identity, ETSI MEC has specified the UEIdentity feature [7]. This includes a dedicated API to trigger the MEP to route specific UE traffic flows to specified end points, without having to route the traffic via a MEC application. For local breakout, for example, the process would begin with the UE entering the serving area of the MEP within the enterprise zone. Detection could be provided by a BYOD client application on the UE, which would then initiate a connection to the BYOD server MEC application hosted by the MEP. Once the connection to the BYOD server had been established, it would be responsible for invoking the traffic routing at the MEP.

A challenge with this approach is that identification of individual traffic flows for a specific UE at the MEP can be problematic since the necessary information to do so may be obfuscated due to the MEP location within the mobile network architecture. For instance, considering a CRAN or bump in the wire deployment, identifiers, such as the UE's International Mobile Subscriber Identity (IMSI), are generally not exposed over the S1 interface. Therefore, the MEP must be provided with alternate identification information that it has direct access to, such as the temporary connection identifiers used on the S1 interface. Considering the user-plane only, such temporary identifiers include the pair of S1 GTP-U Tunnel Endpoint Identifiers (TEIDs). These are dynamically allocated and may change even while the connection is active due to factors such as UE mobility. The consequence is that the EPC sourced mapping information must be provided in near real time. One solution is to deploy probe agents within the EPC to capture temporary identifier information for a given UE identity, e.g. that are provided by the BYOD system. For example, by monitoring the S11 interface, the TEID assignments per IMSI can be recovered by the probe and then forwarded to the MEP or to the BYOD server to invoke at the MEP.

MEC as driver to 5G adoption

Multi-access Edge Computing makes no assumptions on the underlying radio infrastructure, which makes it a highly flexible element in the communications networks. As the delivery technology, together with the underlying hardware of the MEC platform, remains open, this enables new levels of adaptability to the chosen deployment scenario. Therefore, service providers (SPs) can use MEC as a revenue generator and application test bed (including service producing applications) without being forced to wait for full ratification of the 5G standard and the associated capital investment. This approach enables SPs to offer third parties a cost effective way to trial their applications. Using an "edge cloud", the SP can host applications in a virtual retail space, test the revenue return, and scale up or remove as appropriate. So, starting out as a 4G edge test bed with limited deployments at first, MEC allows a smooth transition into the 5G network rollout, removing the need for major upgrades when the time for transition arrives.

Another focus area for transitioning from today's 4G to 5G networks is re-using the existing deployed systems in the process. Due to the virtualized characteristics of MEC, it has never been easier to monitor performance and resource needs of an application, which, in turn, enables more accurate pricing for operators towards application providers for hosting the applications, as well as dimensioning the edge equipment exactly as required for the application set proposed.

The common feature set of providing much-improved capabilities at the edge of the network, improved intelligence about resources needed at the edge, and the ability to charge for service delivered by cycles, memory, storage and bandwidth delivered, makes it very attractive to start the deployment now in early test sites, roll out to sites that show promise and need for MEC based applications, and then roll out as part of the 5G transition without losing any upfront investment from the earlier test deployments.

Taking into account the above considerations, in the next sections we illustrate how MEC compatibility towards 5G networks may involve:

- Integrating the MEC data plane with the 5G system's one for routing traffic to the local data network and steering to an application;
- An Application Function (AF) interacting with 5G control plane functions to influence traffic routing and steering, acquire 5G network capability information, and support application instance mobility;
- The possibility of reusing the edge computing resources and managing/orchestrating applications and/or 5G network functions, while MEC still orchestrates the application services (chaining).

Deploying MEC in the 5G system architecture

The 5G Service Based Architecture (SBA) specified by 3GPP TS 23.501 [6] contains multiple control plane functional entities, like the Policy Control Function (PCF), the Session Management Function (SMF), the Application Function (AF), etc., and data plane functional entities like the User Plane Function (UPF).

In contrast to the current mobile network architecture, the 5G system was conceived to allow a more flexible deployment of the data plane, aiming to natively support edge computing. As a consequence, the MEC architecture can easily be integrated into that defined for 5G. Figure 8 illustrates an example MEC mapping to the 5G system architecture, where for example the MEC host's data plane can be mapped to 5G's UPF element.

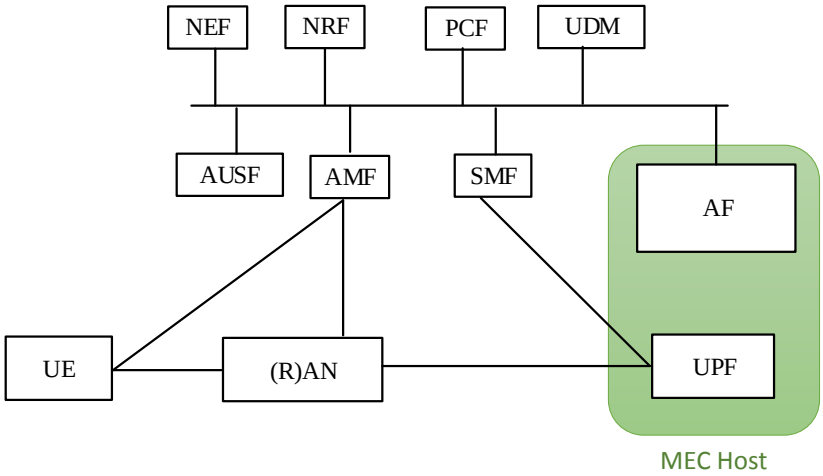


Figure 8: An example of MEC mapping with 5G system architecture

In the example above, the MEC platform would leverage the 5G network architecture and performs the traffic routing and steering function in the UPF. For example, a UL Classifier of UPF is used to divert to the local data plane the user traffic matching traffic filters controlled by the SMF, and further steer to the application. The PCF and the SMF can set the policy to influence such traffic routing in the UPF. Also the AF via the PCF can influence the traffic routing and steering. Therefore MEC in 5G is able to influence the UPF through the standardized control plane interface in SMF similarly to some of the EPC MEC deployment scenarios that we examined in 4G.

Although the position of MEC at the edge site is left to the operators' choice, similarly to what we have done for the 4G MEC deployment, here are a few migration examples to 5G selected architectures. The pictures below show how the MEC host, which includes the 4G core network functions, can be transformed to support 5G by software upgrading the relevant network functions. In the transition to 5G the MEC functionalities introduced with the 4G technology are preserved, fulfilling key requirements such as:

- reusing the edge computing resources;
- interaction with 5G control plane;
- integration with the 5G network.

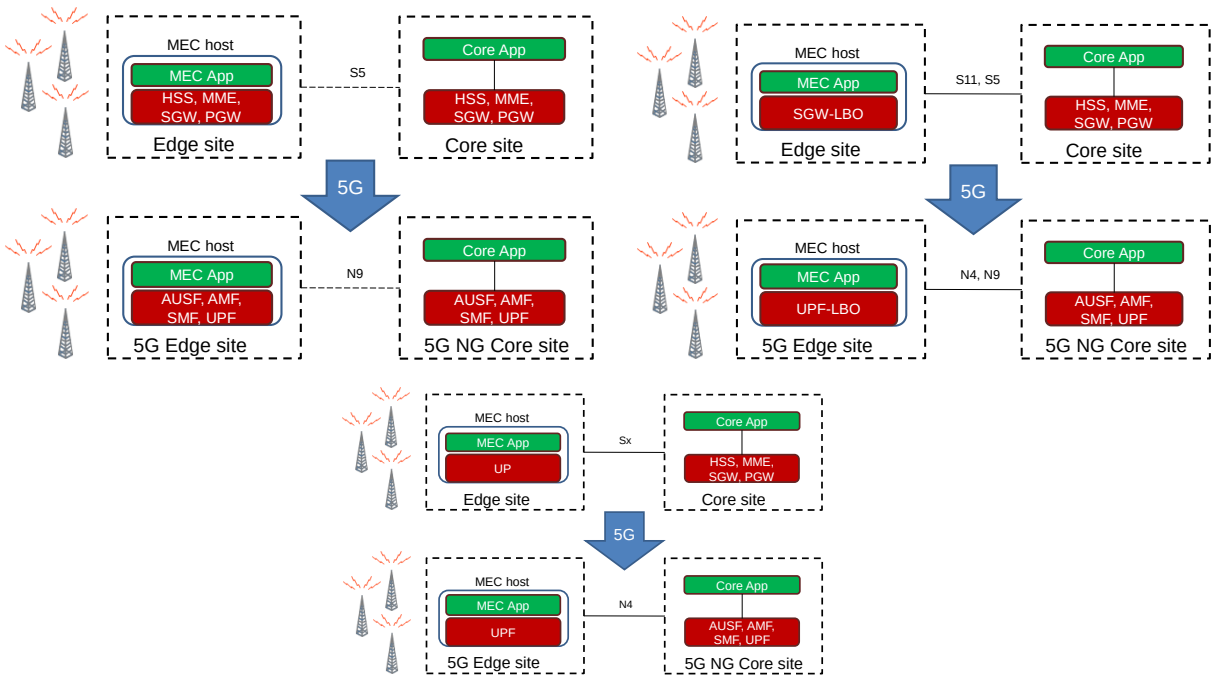


Figure 9: Migration patterns for MEC deployments from 4G to 5G. In the top left diagram, MME, SGW, PGW and HSS migration e.g. to support private networks and mission critical applications. At the top right, SGW-LBO MEC migration to 5G for selective traffic offloading. In the bottom diagram, CUPS migration to 5G.

Management and Orchestration of Cloud vs Edge resources

There is a growing consensus that in the long term, 5G deployments will increasingly integrate fixed-mobile networks infrastructures with cloud computing and MEC. In these future scenarios, the borders between cloud and MEC virtual resources will blur, paving the way towards a sort of "continuum" of logical resources and functions, offering flexibility and programmability through global automated operations. This will require that the orchestration capabilities, which are already a key element for exploiting cloud computing capabilities, become an essential part of the operation of future 5G infrastructure.

In cloud computing, orchestration is a mature concept and is generally referred to as the automation of tasks involved with arranging, managing and coordinating services deployed across different applications and enterprises, i.e., administrative domains, with the purpose of exposing them as single service instances. In 5G service scenarios integrating cloud and MEC, orchestration will have to span across the different service levels: Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS).

On the other hand, in 5G future service infrastructures, IaaS will assume a new meaning, as the CPU, storage, and network resources are not only provided by a collection of data centres, but also by the CPU, storage, and network resources deployed into the Points of Presence (PoPs) of the telecommunications network (in its core, edge and access segments). The PaaS layer will feature a pool of software appliances that facilitate the end-to-end lifecycle of developing, testing, deploying, and hosting services and applications. Some examples of these appliances are databases, web servers, application servers, Apache Hadoop, Apache Storm, and load balancers, each of which will be integrated with other network

appliances (e.g., telecom/internet middle-boxes) to design complex service chains, functions and applications. SaaS will integrate multiple, interoperable PaaS and IaaS resources to deliver services and applications to the end users.

In this broader perspective, orchestration should satisfy both horizontal and vertical interoperability requirements: horizontal, when considering the interoperability between the same tiers in different infrastructure stacks (such as cross-SaaS, cross-PaaS, or cross-IaaS); and vertical when addressing downstream-compatible infrastructure tiers in different stacks.

The integration of 5G management, control and orchestration processes is expected to facilitate applications/services development by providing controlled access to high-level abstractions of 5G resources (e.g. abstractions of computing, memory/storage and networking) thus enabling any vertical application. Moreover just like a true operating system, it should provide automated resource management, scheduling process placement, facilitating inter-process communication, and simplifying installation and management of distributed functions and services, spanning from cloud computing to MEC. This implies a shared data structure capable of supporting multi-vendor systems and applications, which together enable sharing of common data amongst different protocols. Data structures include network state information, i.e., data about system and interface state, forwarding information base state, neighbours table and routing information base and policies. Also, standardized data models are required using, for example, a data-modelling language such as YANG.

The key to a successful management integration of MEC platforms, regardless of their deployment site, is to support standard modes of management, be it IPMI-based management at a very low level, SNMP, modern REST-based protocols or DMTF's Redfish, all of which connect into the operators' network management solutions. Whether Open Source MANO, ONAP or other open- or closed-source solutions, they all offer interfaces into these standard technologies. As MEC moves closer to standard data centre practices, wherever there is a non-standard management approach, typically the integration will become tedious, and, in some cases, even fail.

MEC and NFV

A consolidated vision of the MEC system is about deploying it as part of an NFV environment where MEC applications would be deployed as Virtual Network Functions (VNFs). In this deployment scenario, we have already illustrated some examples of how the MEC and EPC functions may co-operate to create the end-to-end network service.

In order to fully understand the implication of deploying MEC in an NFV environment, ETSI MEC has already started looking at how the two technologies can blend together, resulting in a proposed architecture available in the GR MEC 017 [8]. The MEC system would be virtualized as well and offered as a Network Service which introduces additional challenges in all life-cycle and enablement procedures for the MEC application (VNFs). Also, the management and orchestration systems from both MEC and NFV are meant to co-operate in order to carry out their respective functions.

Support to third party service providers

Third party service providers may own, deploy and manage compute and storage resources to provide MEC service. In order to manage the service, they require a control interface to the mobile network's OAM system. However, no single standardized interface or open API specification exists to interconnect a 3rd party MEC service with a specific MNO network. Each third-party cloud service provider must work

independently with each operator's network, where they intend to provide service, conforming to the operator's or vendor's interfaces, when available. As an alternative to offering edge services, third party service providers may utilize the Service Capability Exposure Function (SCEF) in a 4G system to monitor, gather network information and create innovative services in the cloud. But these are not edge services in true sense.

In order to avoid this ugly scenario, providers of MEC platforms may support standardized and non-standardized interfaces as they feel necessary, but with a focus on the standards-based management functions delivering access to the complete set of capabilities.

Management of MEC applications

The MEC application life-cycle consists of procedures such as: on-boarding, enablement, instantiation, termination, query, disablement and deletion. The Mm3 interface (connecting MEC orchestrator and MEC Platform Manager) is a key component for MEC application on-boarding and enablement. A MEC Orchestrator is the brain, makes placement decisions, whereas a MEC Platform Manager is the executor, allocating resources through VIMs and instantiating applications through a MEC Platform on each MEC host. For each scenario the placement decision is based on the demands of a MEC application and real-time monitoring capabilities of a MEC host.

The MEC platform specification assumes a completely virtualized environment. This is a key requirement in order to enable seamless application lifecycle management paired with seamless platform management. Some applications, however, require hardware acceleration in order to perform certain tasks that are too difficult to achieve in a fully virtualized regime. A resulting requirement for this is the possibility to add access to the acceleration function as part of the virtualization platform. It would be even better if these requirements can be fulfilled in a single box, and can be configured upon start to allow communality of units across multiple deployments, while matching the local requirements when the unit is started.

A dynamic start-up and shutdown of applications across multiple machines, selecting the best-cost solution that matches the application's requirements, enables telecoms operators to select the best match between application, performance needed and delivered without adding unnecessary overhead and upfront investment until it is really required.

Conclusions

Multi-access Edge Computing brings a network technology featuring a whole set of application-oriented functionalities, such as: policy-based traffic forwarding control, DNS policy management, application enablement and orchestration, and, optional services, like RNIS, location and bandwidth management. The key element in the MEC architecture is the MEC host, a general purpose edge computing facility that provides the computing, storage and other resources required by applications such as IoT data pre-processing, VR/AR, video streaming and distribution, V2X, etc.

In this document, we have explored how the MEC system can be deployed in existing 4G networks, by showing different options to install the MEC host along with the 4G system architecture components, and observing how such installation choices impact on the running system and architecture.

Moreover, we have demonstrated how the MEC system deployed for 4G networks could be migrated to future 5G networks, looking at the problem from different angles, including compliance with 5G system architecture, adoption of cloud computing and NFV paradigm, protection of the investment during network upgrade.

It is clear that in order for MEC platforms to be widely adopted by Mobile Network Operators as bridge to 5G, the MEC approach used needs to

1. create value for the customers with real business justifications
2. have minimal impact on existing 4G architecture and network processes
3. use standard 3GPP interfaces to the largest possible extent
4. provide a seamless software-only upgrade to 5G user plane functionality.

Whereas individual use cases that are deployed independently of mobile operators may allow deployment of any of the solutions described above, it appears that the family of solutions that push EPC functionality to the edge and are fully softwarized (i.e., cloud-ready) provide the most effective bridge to 5G.

List of abbreviations

3GPP	3rd Generation Partnership Project
4G, 5G	4th, 5th generation of mobile networks
AF	Application Function
API	Application Programming Interface
APN	Access Point Name
AR	Augmented Reality
BYOD	Bring Your Own Device
CDN	Content Delivery Network
CRAN	Cloud RAN
CUPS	Control/User Plane Separation
DMTF	Distributed Management Task Force
DNS	Domain Name System
DSCP	Differentiated Service Code Point
eNB	Evolved Node B
ETSI	European Telecommunications Standards Institute
EPC	Evolved Packet Core
GTP, GTP-U	GPRS Tunnelling Protocol, GTP-User plane
HSS	Home Subscriber Server
IaaS	Infrastructure as a Service
IMS	IP Multimedia Subsystem
IMSI	International Mobile Subscriber Identity
IoT	Internet of Things
IP	Internet Protocol
IPMI	Intelligent Platform Management Interface
IPsec	Internet Protocol Security
LAN	Local Area Network
LI	Lawful Interception
LIPA	Local IP Access
M2M	Machine to Machine
MANO	Management and Orchestration
MCPTT	Mission Critical Push to Talk
MEC	Multi-access Edge Computing
MEP	MEC Platform
MME	Mobility Management Entity
MNO	Mobile Network Operator
NFV	Network Functions Virtualisation
OAM	Operations, Administration and Management
ONAP	Open Network Automation Platform
PaaS	Platform as a Service
PCF	Policy Control Function
PDN	Packet Data Network
PGW, PGW-C	PDN Gateway, PGW Control plane
PoP	Point of Presence

QoS	Quality of Service
RD	Retained Data
REST	Representational State Transfer
RNIS	Radio Network Information Service
SaaS	Software as a Service
SBA	Service Based Architecture
SCEF	Service Capability Exposure Function
SGW, SGW-C	Serving Gateway, SGW Control plane
SGW-LBO	SGW with Local Breakout
SMF	Session Management Function
SNMP	Simple Network Management Protocol
SP	Service Provider
TAC	Tracking Area Code
TEID	Tunnel Endpoint Identifiers
UE	User Equipment
UL	Uplink
UPF	User Plane Function
URLLC	Ultra-Reliable Low Latency Communications
V2X	Vehicle to Everything
VoLTE	Voice over LTE
VIM	Virtualised Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Functions
VR	Virtual Reality
YANG	Yet Another Next Generation

References

[1] Several authors, “Mobile-Edge Computing – Introductory Technical White Paper,” Sept., 2014. https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge_computing_-_introductory_technical_white_paper_v1%2018-09-14.pdf

[2] ETSI GS MEC 011 V1.1.1, “Mobile Edge Computing (MEC); Mobile Edge Platform Application Enablement” (2017-07). http://www.etsi.org/deliver/etsi_gs/MEC/001_099/011/01.01.01_60/gs_mec011v010101p.pdf

[3] ETSI GS MEC 003 V1.1.1, “Mobile Edge Computing (MEC); Framework and Reference Architecture” (2016-03). http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_mec003v010101p.pdf

[4] ETSI White Paper No. 23, “Cloud RAN and MEC: A Perfect Pairing”, First Edition, February 2018, http://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp23_MEC_and_CRAN_ed1_FINAL.pdf

[5] ETSI TS 123 401, “LTE; General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access,” 2008-2017. http://www.etsi.org/deliver/etsi_ts/123400_123499/123401/

[6] 3GPP TS 23.501 V15.0.0, “3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Release 15)” (2017-12) http://www.3gpp.org/ftp//Specs/archive/23_series/23.501/23501-f00.zip

[7] ETSI GS MEC 014 V1.1.1, “Mobile Edge Computing (MEC); UE Identity API” (2018-02) http://www.etsi.org/deliver/etsi_gs/MEC/001_099/014/01.01.01_60/gs_mec014v010101p.pdf

[8] ETSI GR MEC 017 V1.1.1, “Mobile Edge Computing (MEC); Deployment of Mobile Edge Computing in an NFV environment” (2018-02) http://www.etsi.org/deliver/etsi_gr/MEC/001_099/017/01.01.01_60/gr_MEC017v010101p.pdf

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

ETSI White Paper No. 28

MEC in 5G networks

First edition – June 2018

ISBN No. 979-10-92620-22-1

Authors:

Sami Kekki, Walter Featherstone, Yonggang Fang, Pekka Kuure, Alice Li, Anurag Ranjan, Debashish Purkayastha, Feng Jiangping, Danny Frydman, Gianluca Verin, Kuo-Wei Wen, Kwihoon Kim, Rohit Arora, Andy Odgers, Luis M. Contreras, Salvatore Scarpina

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

About the authors

Sami Kekki
Huawei – Editor

Rohit Arora
Hewlett Packard Enterprise

Luis M. Contreras
Telefonica

Yonggang Fang
ZTE

Walter Featherstone
Viavi Solutions

Danny Frydman
Saguna

Feng Jiangping
Huawei

Kwihoon Kim
ETRI

Pekka Kuure
Nokia

Alice Li
Vodafone

Andy Odgers
Quortus

Debashish Purkayastha
Interdigital

Anurag Ranjan
Intel

Salvatore Scarpina
TIM

Gianluca Verin
Athonet

Kuo-Wei Wen
ITRI

Contents

Introduction

Edge computing as an evolution of cloud computing brings application hosting from centralized data centres down to the network edge, closer to consumers and the data generated by applications. Edge computing is acknowledged as one of the key pillars for meeting the demanding Key Performance Indicators (KPIs) of 5G, especially as far as low latency and bandwidth efficiency are concerned. However, not only is edge computing in telecommunications networks a technical enabler for the demanding KPIs, it also plays an essential role in the transformation of the telecommunications business, where telecommunications networks are turning into versatile service platforms for industry and other specific customer segments. This transformation is supported by edge computing, as it opens the network edge for applications and services, including those from third parties.

ETSI ISG MEC (Industry Specification Group for Multi-access Edge Computing) is the home of technical standards for edge computing. The group has already published a set of specifications (Phase 1) focusing on management and orchestration (MANO) of MEC applications [2, 3], application enablement API [4], service Application Programming Interfaces (APIs) [5, 6, 7, 8] and the User Equipment (UE) application API [9]. The MANO and application enablement functions contribute to enabling service environments in edge data centres, while the service APIs enable the exposure of underlying network information and capabilities to applications. One of the key value-adding features of the MEC specification is this ability for applications to gain contextual information and real-time awareness of their local environment through these standardized APIs. This local services environment is a flexible and extendable framework, as new services can be introduced by following the API guidelines in [10], when creating new service APIs. And finally, the UE application API lets the client application in the UE interact with the MEC system for application lifecycle management.

5G networks based on the 3GPP 5G specifications [11] are a key future target environment for MEC deployments. The 5G system specification and its Service Based Architecture (SBA) leverage the service-based interactions between different network functions, aligning system operations with the network virtualization and Software Defined Networking paradigms. These very same characteristics are shared by MEC specifications. In addition, 3GPP 5G system specifications define the enablers for edge computing, allowing a MEC system and a 5G system to collaboratively interact in traffic routing and policy control related operations. MEC features together with these complementary technical enablers of the 5G system allow integration of these systems to create of a powerful environment for edge computing.

In the following sections of the white paper, we illustrate and explain ways to deploy and integrate MEC in the 5G system. The emphasis of the document is on the opportunities for MEC to benefit from the edge computing enablers of the 5G system specification, and for 3GPP ecosystem to benefit from the MEC system and its APIs as a set of complementary capabilities to enable applications and services environments in the very edge of mobile networks.

Support for Edge Computing in 3GPP

In the 5G system specifications there is a set of new functionalities that serves as enablers for edge computing. These enablers are essential for integrated MEC deployments in 5G networks. This white paper is focused primarily on utilizing these edge computing enablers. The full list of enablers with brief explanations can be found in clause 5.13 of [10].

1. Local Routing and Traffic Steering: the 5G Core Network provides the means to select traffic to be routed to the applications in the local data network. A PDU Session may have multiple N6 interfaces towards the data network. The UPFs that terminate these interfaces are said to support PDU Session Anchor functionality. Traffic steering by the UPF is supported by Uplink Classifiers that operate on a set of traffic filters matching the steered traffic, see clause 5.6.4.2 of [10] or alternatively by IPv6 multi-homing, where multiple IPv6 prefixes have been associated with the PDU session in question, see clause 5.6.4.3 of [10].
2. The ability of an Application Function to influence UPF (re)selection and traffic routing directly via the Policy Control Function (PCF) or indirectly via the Network Exposure Function (NEF), depending on the operator's policies, see clause 5.6.7 of [10].
3. The Session and Service Continuity (SSC) modes for different UE and application mobility scenarios. Description of the SSC modes 1, 2 and 3 is found in clause 5.6.9 of [10]
4. Support of Local Area Data Network (LADN) by the 5G Core Network by providing support to connect to the LADN in a certain area where the applications are deployed. The access to a LADN is only available in a specific LADN service area, defined as a set of Tracking Areas in the serving PLMN of the UE. LADN is a service provided by the serving PLMN of the UE, see section 5.6.5 of [10].

Deployment of MEC in 5G

MEC as it is deployed currently in the 4th generation LTE networks, is connected to the user plane via one of the options described in the ETSI White Paper *MEC deployments in 4G and evolution towards 5G* [11]. With LTE networks already having been deployed for a number of years, it was necessary to design the MEC solution as an add-on to a 4G network in order to offer services in the edge. Consequently, the MEC system as defined in [1] and in the related interface specifications, is to a large extent self-contained, covering everything from management and orchestration down to interactions with the data plane for steering specific traffic flows.

With 5G, the starting point is different, as edge computing is identified as one of the key technologies required to support low latency together with mission critical and future IoT services. This was considered in the initial requirements. The system was designed from the beginning to provide efficient and flexible support for edge computing to enable superior performance and quality of experience.

The design approach taken by 3GPP allows the mapping of MEC onto Application Functions (AF) that can use the services and information offered by other 3GPP network functions based on the configured policies. In addition, a number of enabling functionalities were defined to provide flexible support for different deployments of MEC and to support MEC in case of user mobility events. The new 5G architecture is described and explained in more detail in the next clause.

5G System architecture & MEC

The 5G system architecture specified by 3GPP and described in [10] has been designed to cater for a wide set of use cases ranging from a massive amount of simple IoT devices to the other extreme of high bit rate, high reliability mission critical services. Supporting all the use cases with the same and common architecture has required significant changes in design philosophies both for the RAN and the core network.

One significant architectural change was made to the communications between the core network functions that until now have relied on a point-to-point paradigm. In the 5G system specification there are two options available for the architecture; one with the traditional reference point and interface approach and the other where the core network functions interact with each other using a Service Based Architecture (SBA). In this white paper the emphasis is on the SBA option of the 5G system architecture.

With the SBA, there are functions that consume services and those that produce services. Any network function can offer one or more services. The framework provides the necessary functionality to authenticate the consumer and to authorize its service requests. The framework supports flexible procedures to efficiently expose and consume services. For simple service or information requests, a request-response model can be used. For any long-lived processes, the framework also supports a subscribe-notify model. The API framework defined by ETSI ISG MEC is aligned with these principles and in fact does exactly the same for MEC applications as the SBA does for network functions and their services. The functionality needed for efficient use of the services includes registration, service discovery, availability notifications, de-registration and authentication and authorization. All this functionality is the same in both the SBA and the MEC API frameworks.

In the figure below the 3GPP 5G system with its SBA is shown on the left, while the MEC system architecture is on the right. In the remainder of this white paper the focus is on describing how to deploy the MEC system in a 5G network environment in an integrated manner where some of the functional

entities of MEC (blue boxes in MEC system part) interact with the network functions of the 5G core network.

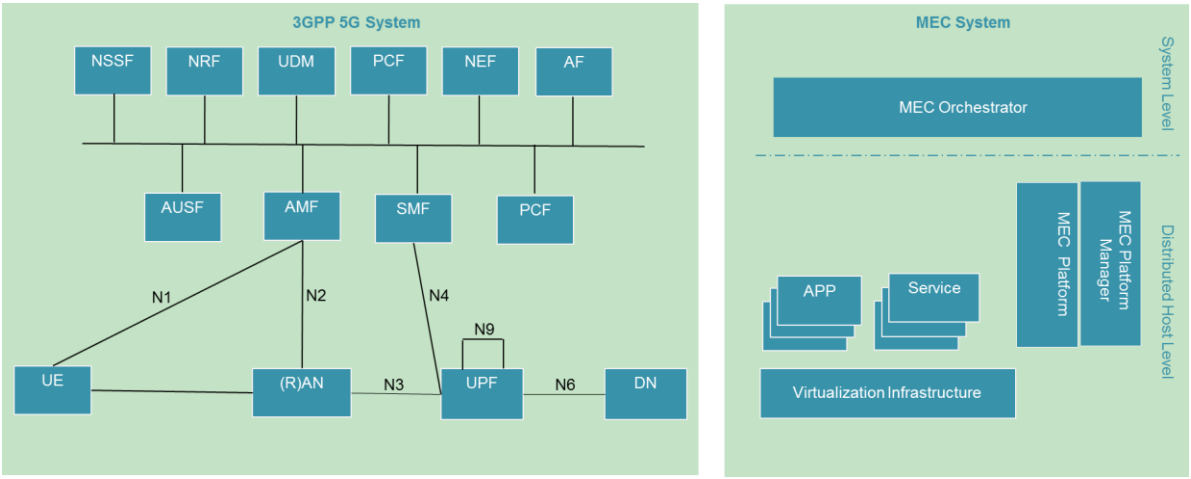


Figure 1. 5G Service-Based Architecture and a generic MEC system architecture

The network functions and the services they produce are registered in a Network Resource Function (NRF) while in MEC the services produced by the MEC applications are registered in the service registry of the MEC platform. Service registration is part of the Application Enablement functionality [4]. To use the service, if authorized, a network function can directly interact with the network function that produces the service. The list of available services can be discovered from the NRF. Some of the services are accessible only via the NEF, which is also available to untrusted entities that are external to the domain, to access the service. In other words, the NEF acts as a centralized point for service exposure and also has a key role in authorizing all access requests originating from outside of the system.

In addition to AF, NEF and NRF, there are a number of other functions that are worth introducing. The procedures related to authentication are served by the Authentication Server Function (AUSF).

One of the key concepts in 5G is Network Slicing that allows the allocation of the required features and resources from the available network functions to different services or to tenants that are using the services. The Network Slice Selection Function (NSSF) is the function that assists in the selection of suitable network slice instances for users, and in the allocation of the necessary Access Management Functions (AMF). A MEC application, i.e. an application hosted in the distributed cloud of a MEC system can belong to one or more network slices that have been configured in the 5G core network.

Policies and rules in the 5G system are handled by the PCF. The PCF is also the function whose services an AF, such as a MEC platform, requests in order to impact the traffic steering rules. The PCF can be accessed either directly, or via the NEF, depending whether the AF is considered trusted or not, and in the case of traffic steering, whether the corresponding PDU session is known at the time of the request.

The Unified Data Management (UDM) function is responsible for many services related to users and subscriptions. It generates the 3GPP AKA authentication credentials, handles user identification related information, manages access authorization (e.g. roaming restrictions), registers the user serving NFs (serving AMF, Session Management Function (SMF)), supports service continuity by keeping record of SMF/Data Network Name (DNN) assignments, supports Lawful Interception (LI) procedures in outbound roaming by acting as a contact point and performs subscription management procedures.

The User Plane Function (UPF) has a key role in an integrated MEC deployment in a 5G network. UPFs can be seen as a distributed and configurable data plane from the MEC system perspective. The control of that data plane, i.e. the traffic rules configuration, now follows the NEF-PCF-SMF route. Consequently, in some specific deployments the local UPF may even be part of the MEC implementation.

The following figure shows how the MEC system is deployed in an integrated manner in 5G network.

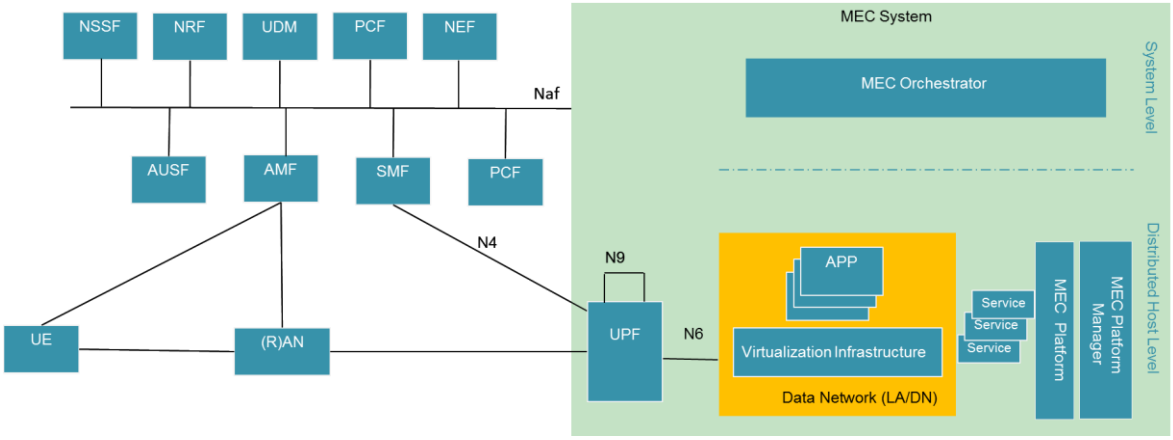


Figure 2. Integrated MEC deployment in 5G network

In the MEC system on the right-hand side of Figure 2 the MEC orchestrator is a MEC system level functional entity that, acting as an AF, can interact with the Network Exposure Function (NEF), or in some scenarios directly with the target 5G NFs. On the MEC host level it is the MEC platform that can interact with these 5G NFs, again in the role of an AF. The MEC host, i.e. the host level functional entities, are most often deployed in a data network in the 5G system. While the NEF as a Core Network function is a system level entity deployed centrally together with similar NFs, an instance of NEF can also be deployed in the edge to allow low latency, high throughput service access from a MEC host.

In this white paper it is assumed that MEC is deployed on the N6 reference point, i.e. in a data network external to the 5G system. This is enabled by flexibility in locating the UPF. The distributed MEC host can accommodate, apart from MEC apps, a message broker as a MEC platform service, and another MEC platform service to steer traffic to local accelerators. The choice to run a service as a MEC app or as a platform service is likely to be an implementation choice and should factor in the level of sharing and authentication needed to access the service. A MEC service such as a message broker could be initially deployed as a MEC app to gain time-to-market advantage, and then become available as a MEC platform service as the technology and the business model matures.

Managing user mobility is a central function in a mobile communications system. In a 5G system it is the Access and Mobility Management Function (AMF) that handles mobility related procedures. In addition, the AMF is responsible for the termination of RAN control plane and Non-Access Stratum (NAS) procedures, protecting the integrity of signalling, management of registrations, connections and reachability, interfacing with the lawful interception function for access and mobility events, providing authentication and authorization for the access layer, and hosting the Security Anchor Functionality (SEAF). With the SBA, the AMF provides communication and reachability services for other NFs and it also allows subscriptions to receive notifications regarding mobility events.

Similarly to the AMF, the Session Management Function (SMF) is in a key position with its large number of responsibilities. Some of the functionality provided by the SMF includes session management, IP address allocation and management, DHCP services, selection/re-selection and control of the UPF, configuring the traffic rules for the UPF, lawful interception for session management events, charging and support for roaming. As MEC services may be offered in both centralized and edge clouds, the SMF plays a critical role due to its role in selecting and controlling the UPF and configuring its rules for traffic steering. The SMF exposes service operations to allow MEC as a 5G AF to manage the PDU sessions, control the policy settings and traffic rules as well as to subscribe to notifications on session management events.

Until now the SBA has been discussed with its network functions and their roles in the 5G system. While they play an essential role in enabling the flexible integration of MEC in the next generation system, there are a few additional high-level concepts worth listing that are essential in providing high performance MEC services with an unparalleled quality of experience.

- Concurrent access to local and central Data Networks (DN) in a single PDU session
- Selection of the User Plane Function for a PDU session close to the UE's point of attachment
- Selection/establishment of a new UPF based on UE mobility and connectivity related events received from the SMF, see the "UE and application mobility" section
- Network Capability Exposure to allow MEC (AF) to request information about UE(s) or request actions towards UE(s), see the "Capabilities exposure" section
- Possibility for MEC (AF) to influence traffic steering for a single UE or a group of UEs, see the "Traffic steering" section
- Support for LI and Charging for MEC in the edge cloud, see the "Regulatory requirements" and "Charging" sections
- Indication about LADN availability for UEs (Local Access Data Network) for specific and local MEC services

MEC deployment scenarios

Logically MEC hosts are deployed in the edge or central data network and it is the User Plane Function (UPF) that takes care of steering the user plane traffic towards the targeted MEC applications in the data network. The locations of the data networks and the UPF are a choice of the network operator and the network operator may choose to place the physical computing resources based on technical and business parameters such as available site facilities, supported applications and their requirements, measured or estimated user load etc. The MEC management system, orchestrating the operation of MEC hosts and applications, may decide dynamically where to deploy the MEC applications.

In terms of physical deployment of MEC hosts, there are multiple options available based on various operational, performance or security related requirements. The following figure gives an outline of some of the feasible options for the physical location of MEC.

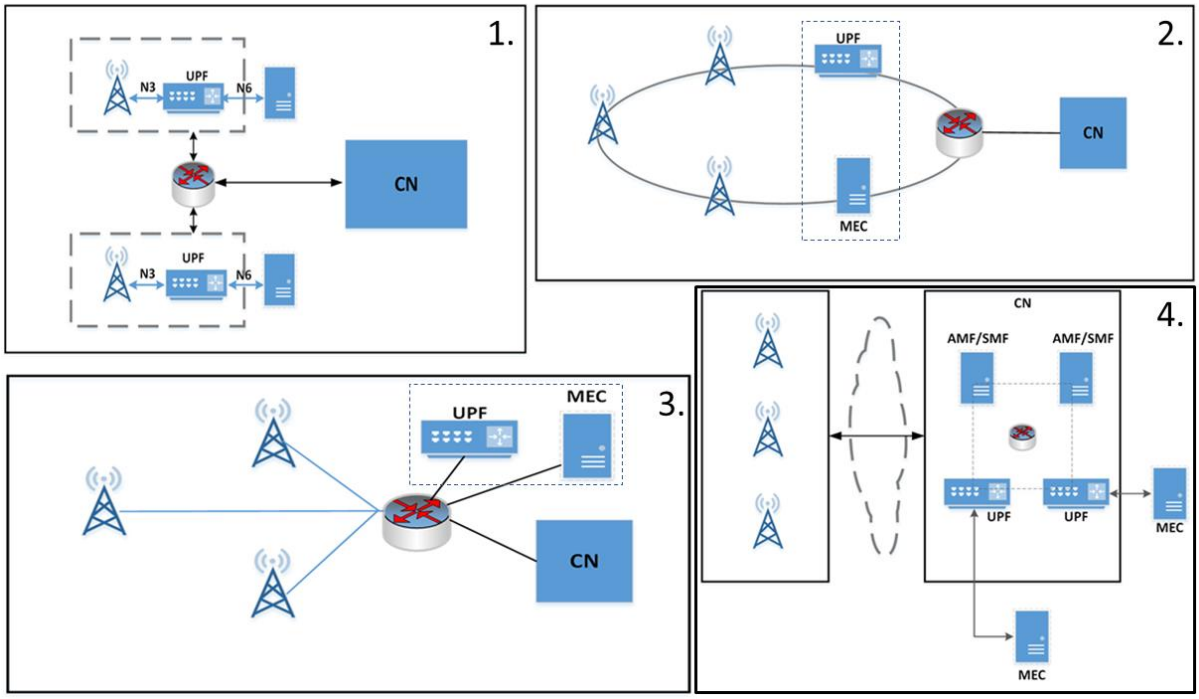


Figure 3. Examples of the physical deployment of MEC.

1. MEC and the local UPF collocated with the Base Station.
2. MEC collocated with a transmission node, possibly with a local UPF
3. MEC and the local UPF collocated with a network aggregation point
4. MEC collocated with the Core Network functions (i.e. in the same data centre)

The options presented above show that MEC can be flexibly deployed in different locations from near the Base Station to the central Data Network. Common for all deployments is the UPF that is deployed and used to steer the traffic towards the targeted MEC applications and towards the network.

Traffic steering

Traffic steering in MEC refers to the capability of the MEC system to route traffic to the targeted applications in a distributed cloud. Whilst in a generic MEC architecture as defined in [1], traffic steering is controlled by the MEC platform through configuring the data plane via the Mp2 reference point. In a 5G integrated deployment the role of the data plane is delegated to the User Plane Function (UPF). A UPF plays the central role in routing the traffic to desired applications and network functions. In addition to the UPF, there are a few related procedures specified by 3GPP that are used to enable flexible and efficient routing of the traffic to applications. One such procedure is the Application Function (AF) influence on traffic routing as described in clause 5.6.7 of [10]. It allows an AF to influence the selection and re-selection of a local UPF as well as request services to configure the rules to allow the traffic steering to a data network.

The toolset offered by a 5G network can be used by the AF, which, in the case of MEC, maps to Functional Entities (FE) of the MEC system. When a MEC application is instantiated, no traffic is routed to the application until the application is ready to receive traffic and the underlying data plane is configured to

route the traffic towards it. This configuration is done by the MEC platform. When deployed in a 5G network, a MEC FE such as a MEC platform, is in the role of a 5G AF towards the 5G core network. It interacts with the PCF to request traffic steering by sending information that identifies the traffic to be steered. The PCF will transform the request into policies that apply to targeted PDU session(s) and provides the routing rules to the appropriate Session Management Function (SMF). Based on the received information, the SMF identifies the target UPF, if it exists, and initiates configuration of the traffic rules there. If no applicable UPF exists, the SMF can insert one or more UPFs in the data path of the PDU session.

In the integrated deployment as described above the data plane functionality of the (generic) MEC architecture is now the responsibility of the UPF. This UPF is influenced by MEC through control plane interactions with 5G core network functions, rather than via a specific reference point that is termed Mp2 in the MEC architecture.

The SMF can also configure the UPF with different options for traffic steering. In the case of IPv4, IPv6, IPv4v6 or Ethernet, the SMF may insert an Uplink Classifier function (UL CL) in the data path. The UL CL can be configured with the traffic rules to forward the uplink traffic towards different targeted applications and network functions, and in the downlink direction it will merge the traffic destined to the UE(s). Alternatively, for PDU Sessions using IPv6 or IPv4v6, and if supported by the UE, the SMF may use the Multi-homing concept for traffic steering. In such a case, the SMF would insert a Branching Point function in the target UPF and configure it to split the UL traffic to a local application instance and the services in the central cloud based on the Source Prefixes of the IP data packets.

The 3GPP 5G system offers a flexible framework for AFs by enabling traffic steering based on a wide set of different parameters. This allows generic traffic rule setting or specific rule setting for certain specific UE(s). The parameters that can be used in traffic steering requests may contain, for instance, information to identify the traffic (DNN, S-NSSAI, AF-Service-Identifier, 5 tuple etc.), a reference ID to preconfigured routing information, a list of DNAIs, information about target UE(s), indication about application relocation possibilities, temporal validity condition (timeframe when routing condition is valid), spatial validity condition (location of UE e.g. geographic area), notification type for user plane management notifications and AF transaction ID (enables modifications to the routing rule).

In addition to selecting the UPF and configuring the traffic steering rules, the 5G system also provides efficient tools for MEC Functional Entities, e.g. tools for a MEC platform, or a MEC orchestrator, to monitor the mobility events that relate to users of MEC application instances in local clouds. MEC FEs can subscribe to user plane path management event notifications from SMF(s), in which case it would receive notifications about path changes, e.g. when the Data Network Access Identifier (DNAI) for a particular PDU Session changes. The MEC management functions can use these notifications to trigger traffic routing configuration or application relocation procedures.

The discussion above assumes that the MEC system, with the relevant Functional Entities supported there, are trusted by the 3GPP network and that policies allow direct access from AFs to the 5G Core Network Functions. There are cases however, where a MEC FE needs to request services from the Network Exposure Function (NEF), for example when MEC is not considered trusted and the policy does not allow direct interaction with the 5G Core NFs. Also, whenever the request targets, or may target multiple PCFs, it would be required to go via the NEF.

UE and application mobility

The MEC system combines the environments of networking and computing at the edge of the network to optimize the performance for ultra-low latency and high bandwidth services. One direct consequence of hosting the applications at the edge, possibly even very close to the radio nodes, is the exposure of those applications to UE mobility. The UEs, whether traditional handheld devices or vehicles equipped with V2X systems, are expected to be mobile, and their movements may render the location of the currently used edge application host non-optimal in the long run, even though the underlying network maintained the service continuity between the endpoints. For the MEC system to maintain the application requirements in a mobile environment, application mobility is required. In practice, this means that the application instance that is serving the user is changed to a new location. And consequently, for stateful applications, the user context also needs to be transferred. In wide area MEC deployments it can be anticipated that the MEC hosts in the system are provisioned and configured with the application supported across the system, thus reducing the likelihood that an application needs to be relocated from one host to another. This, however, does not yet remove the need for a user context transfer between the source and the target MEC host for stateful application services.

An application service may be categorized as either a stateful or a stateless service. Application mobility for a stateful service requires transferring and synchronizing the service state between the original and relocated application instance in order to provide service continuity. Service state synchronization highly depends on the implementation of the application itself, thus requiring support from the application developer. In other words, the application must be designed in such a way that multiple instances of the application can run concurrently, and state (context) of the application instance can be captured in the source instance and copied to another instance independently from the operation of the instance itself. Then the service produced by the relocated application instance in the target MEC host can continue in a seamless manner from the state of the application instance in the source MEC host at the time of UE disconnection from that. The support of application mobility for a stateless service, on the other hand, is relatively simple as most likely it will not need service state (application context) transfer and synchronization between the original instance in the source and the instance in the target.

Application mobility is a unique feature of the MEC system. It is necessary to be able to relocate a user's context and/or application instance from one MEC host to another to continue offering an optimized service experience for the user. Application mobility is a part of service continuity support, in which the service to the UE will resume once the user's context and/or application instance has been relocated to another MEC host. The following figure illustrates the basic scenario for application mobility in an integrated MEC deployment in a 5G network.

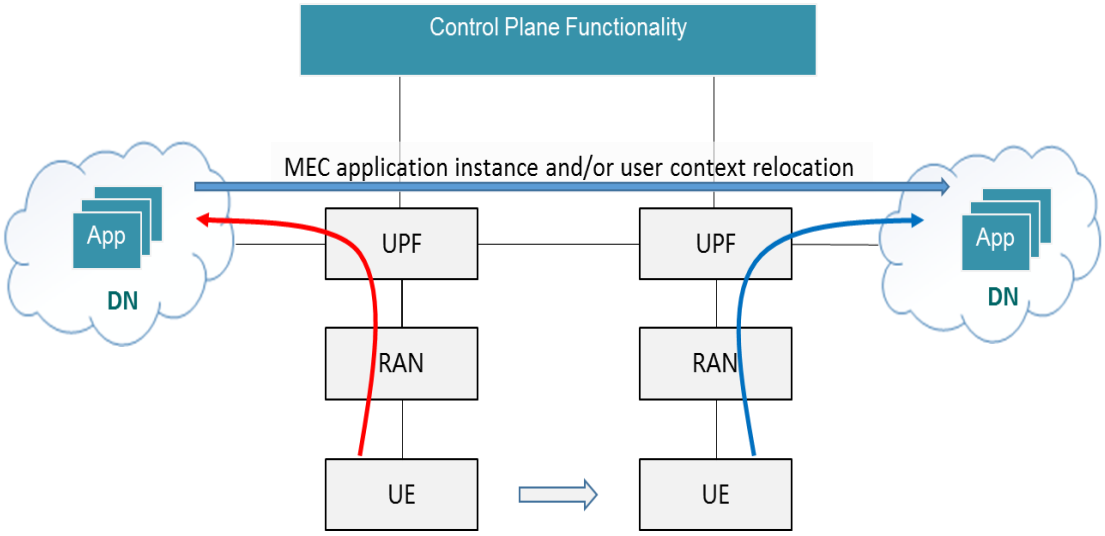


Figure 4. The principle of MEC application mobility.

The MEC application mobility feature is a work in progress in ETSI ISG MEC. The current definitions of the feature are grouped into procedures that may or may not apply depending on the characteristics of the application, environment and capabilities of the MEC host, MEC orchestrator and also of the MEC application itself. The procedures that are currently being developed are application mobility enablement, detection of UE movement, validation of application mobility, user context transfer and/or application instance relocation, and post-processing of application relocation. Ways in which a mobile UE application client may contribute to the enablement of the application mobility service to achieve the service continuity of the application are also being considered.

The detection of UE movement to a new serving cell is one of the triggers for application mobility, which may rely on the 5G Network Exposure Function (NEF) and the ability of the MEC functional entities to subscribe to relevant event notifications available there. The MEC platform may also subscribe to the Radio Network Information produced by the RNIS [5]. Through Radio Network Information the platform can identify UEs experiencing cell change and determine whether they are about to move out of the serving area of the current MEC host.

Applications running in the MEC system may produce a wide range of services from multi-media and gaming to machine type services like V2X. This variety creates significant complexity for application mobility support. The application/service providers should consider all aspects of the application lifecycle in the mobile environment, including the application mobility, when planning their application and its roll-out in the network edge.

Capabilities exposure

As previously highlighted, there is a specific function, namely the Network Exposure Function (NEF), to expose capability information and services of the 5G CN Network Functions to external entities. Such entities could include Application Functions (AF) such as MEC system functional entities. While 5G Service Based Architecture (SBA) also enables direct access to a Network Function for an authorized AF, there are many cases when services and capabilities are exposed over NEF. Those include the following:

- Monitoring: Allows an external entity to request or subscribe to UE related events of interest. The monitored events include a UE's roaming status, UE loss of connectivity, UE reachability and location related events (e.g. location of a specific UE, or identification of UEs within a geographical area). The AMF & UDM are the key entities in providing access to such event information.
- Provisioning: Allows an external entity to provision expected UE behaviour to the 5G system, for instance predicted UE movement, or communication characteristics.
- Policy and Charging: Handles QoS and charging policy for UE based requests made by an external party, which also facilitates sponsored data services. The PCF is the key entity with regard to Policy and Charging Control (PCC), although most NFs are involved to some degree in supporting the PCC framework.

Figure 5 illustrates an example of 5G capability exposure to the MEC system. In this case the MEC orchestrator (MEC system level management) appears as a 5G AF, providing centralized functions for managing the computing resources and operation of the MEC hosts. In addition, it offers orchestration of MEC applications running on MEC hosts. The MEC orchestrator as a 5G AF interacts with NEF and with other relevant NFs with regards to overall Monitoring, Provisioning, Policy and Charging capabilities. The MEC host, on the other hand, might be deployed at the edge of 5G RAN to leverage the advantages of MEC for optimizing the performance of applications and improving users' Quality of Experience. Therefore, it is possible that the MEC platform may need direct exposure to the Centralized Units (CUs) of the 5G RAN and potentially even the Distributed Units (DUs). For example, services offered by a MEC host such as the Radio Network Information Service (RNIS) [5] rely on exposure of the RAN capabilities, especially for the up to date radio information related to UEs. Such information could be used to help MEC applications running on the MEC host to optimize the services offered to those UEs. Directly exposing the radio information, such as received signal received power / quality, to the MEC platform also avoids unnecessary transmission latency and bandwidth consumption of routing messages to its consumers (i.e. MEC applications) via the Core Network. The exposure of local network information is a task of a local NEF instance deployed in the edge.

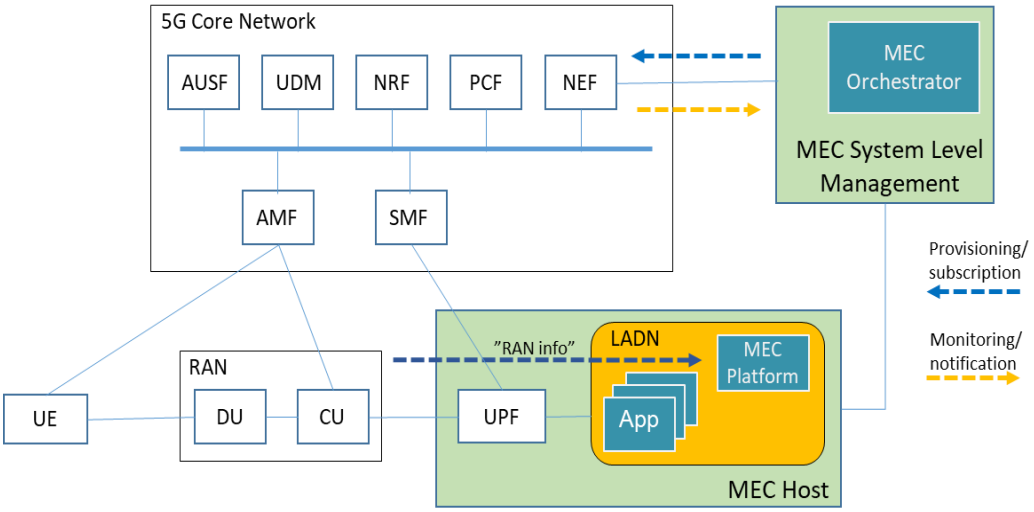


Figure 5. Capabilities exposure (MEC deployed in Local DN).

The MEC system is also able to leverage network capability information exposed by the 5G RAN to provide add-on services to MEC applications. For example, with access to real-time radio signal measurements, the MEC platform, or even a MEC service producing application, may calculate the precise position of a UE and expose it to MEC service consuming application via the MEC Location Service [6]. Such location information could even be provided back to the 5G network, for instance in relation to the NEF offered provisioning capability with regards to UE location predictions. Information can be used by the 5G system to optimize the service to the end user or be used as part of an overall Location-Based Service (LBS) offering, e.g. UE location-based marketing. This latter example illustrates the versatility of the SBA, where also the MEC functions/applications can produce services for the 5G system.

Charging

The integrated deployment of MEC in a 5G system relies on the UPF as the PDU session anchor and gateway to data networks where the MEC environment is deployed. Consequently, the same charging mechanisms and capabilities apply as apply to non-MEC applications.

The transformation of the telco networks into 3rd party service hosting environments where the application cloud becomes an integral part of the telco network calls for new approaches in charging principles and capabilities. Along with the tight integration between 5G NFs such as UPF and SMF and MEC, it is expected that the support of both online charging and offline charging will be relatively straightforward, which would allow 3GPP compliant MEC deployments with charging natively supported for MEC applications.

However more investigation is needed and ETSI ISG MEC looks at the 3GPP charging experts as the source of any new capabilities.

Regulatory requirements

5G Lawful Interception (LI) and Retained Data (RD) continue to play a crucial role in helping Law Enforcement Agencies (LEAs) to combat terrorism and serious criminal activity. LI enables a LEA to perform electronic surveillance on an individual (a target) as authorized by a judicial or administrative order. To facilitate the lawful intercept process, certain legislation and regulations require service providers and Internet service providers to implement their networks to explicitly support authorized electronic surveillance. Actions taken by the service providers include: provisioning the target identity in the network to enable isolation of target communications (separating it from other users' communications), duplicating the communications for the purpose of sending the copy to the LEA, and delivering the Interception Product to the LEA.

In the context of MEC, it is recommended to utilise the LI&RD at SMF and UPF based on the CUPS LI model (Control and User Plane Separation of EPC nodes) as specified in 3GPP Rel-14. The figure below shows an example 5G LI model that is currently under discussion in 3GPP. As explained in the previous sections, the UPF is effectively the data plane for the MEC hosts, thus it is an integral part of the MEC deployment in a 5G network. Consequently, the 3GPP LI&RD is natively supported for MEC application traffic as it is for any application traffic passing the UPF.

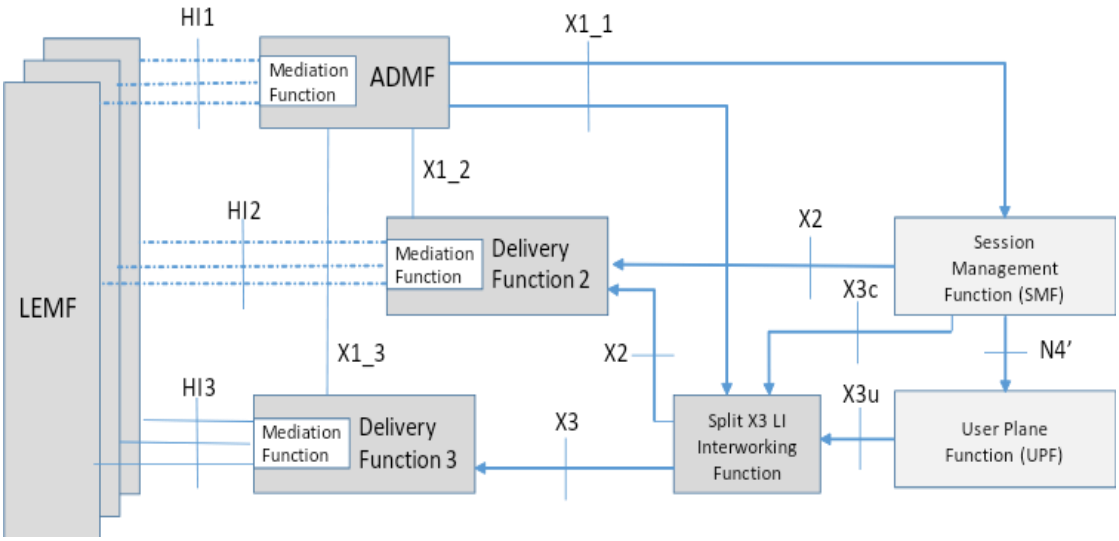


Figure 6. 5G Lawful Interception architecture (3GPP proposal).

UE application interface

In the MEC reference architecture there is one additional reference point of interest that has not been addressed in the previous sections of this white paper. The Mx2 reference point is between the device application and the User Application Lifecycle Management proxy. ETSI ISG MEC has defined a UE Application API over Mx2 reference point in [9]. This API allows the device application to request certain application lifecycle management actions in the MEC system, such as requesting a list of available MEC applications, and instantiation and termination of a new MEC application. Similarly, the API allows the device application to receive notifications of the change of the MEC application's IP address. The MEC applications that have been instantiated in a MEC host in response to a request of a user via a device application are referred to as user applications. In the context of this white paper, the assumption is that the deployment of the user application lifecycle management proxy does not directly impact the rest of the MEC system integration into the 5G system architecture, and any procedures resulting from requests over UE Application API get routed by the Operations Support System (OSS) towards the MEC system level management.

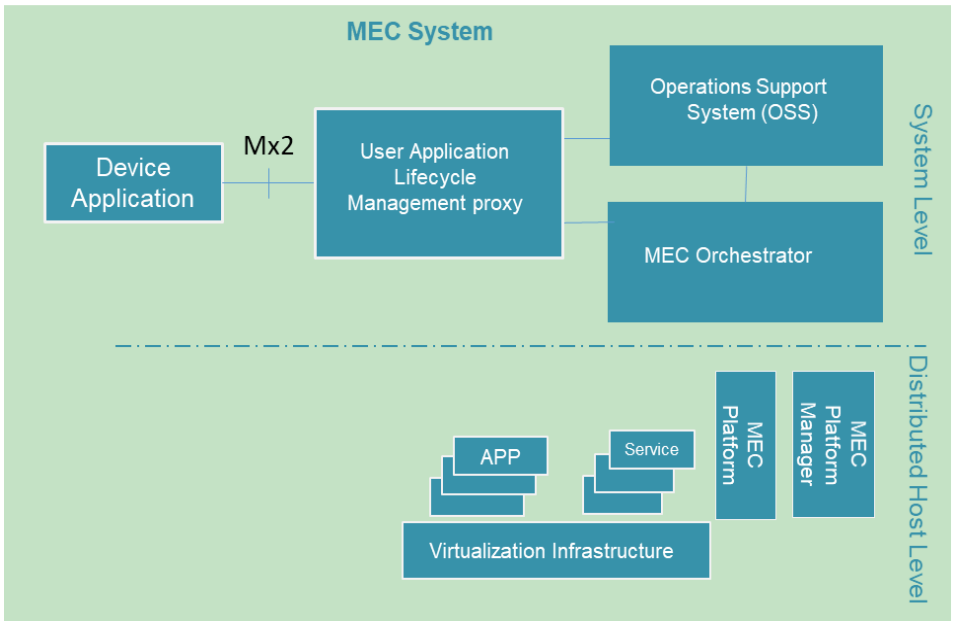


Figure 7. UE application API over Mx2 reference point

For MEC application mobility the UE Application API may be used to assist the MEC system in application/context relocation, as well as in application relocation between different MEC systems or between the MEC system and another cloud system. All these considerations are work in progress in ETSI ISG MEC.

MEC Use Case Examples

This section illustrates the benefits of deploying a MEC system in conjunction with a SBA based 5G network for a selected set of example use cases. In addition, the reader is recommended to consider the 5GAA™ white paper Toward fully connected vehicles: Edge computing for advanced automotive communications [12], which offers insight into how MEC can benefit Cellular V2X.

MEC for third-party cloud service providers

MEC services are typically envisaged as being offered and supplied by Mobile Network Operators. However, a MEC service can also be offered by the third parties. For instance, third party cloud service providers are entities offering MEC application hosting services and resources, while not being traditional network operators. Examples of such third-party providers willing to deploy edge cloud resources include: venue and facility owners or management companies, cell tower owners and neutral host vendors and vehicle fleet management companies (railway, automotive, etc.). Due to operational complexity, costs, or simply difficulty in deploying MEC in hard to reach areas (for example, due to real estate scarcity), MNOs may buy edge cloud services from such third-party providers to supplement their networks. Typically, MNOs are not involved in day to day management and operation of these third-party edge clouds.

The following diagram shows an example of how a third-party cloud service provider could leverage 5G network functions in harmony with the MEC architecture to provide an edge computing service.

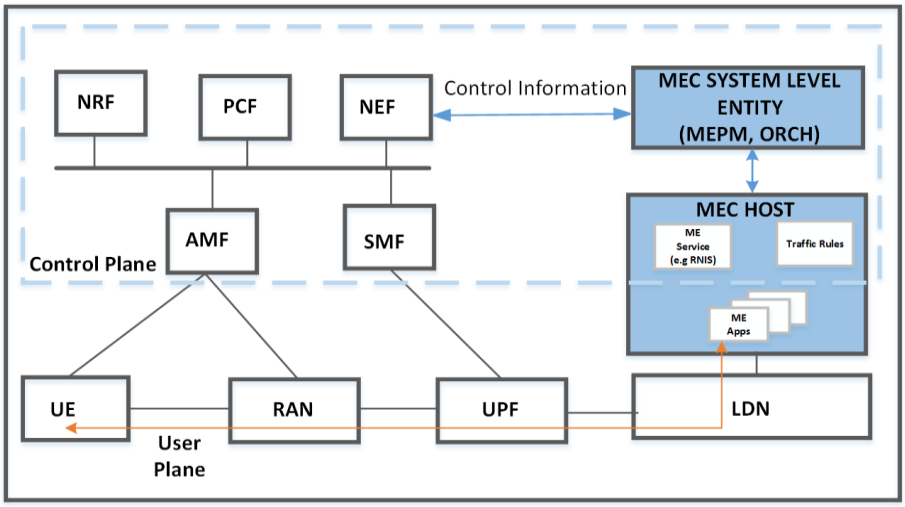


Figure 8. Third-party cloud for MEC in 5G network environment.

The 5G network provides a clear path to enable third party cloud service providers. The Network Exposure Function (NEF) may be used as the entry point in the 5G network for authorized third parties. Using this function, they may configure how appropriate application traffic in the user plane is directed towards MEC applications in Local Data Network (LDN). Also, the NEF may be used for exposing network information such as mobility, radio resource information, etc. to the MEC system. To summarize, the NEF can handle control plane functions for third-party service providers to manage MEC operation. Network information exposed by the NEF is consumed by MEC services and can be exposed to MEC applications. This allows a clear separation between the MNO and the third-party service provider.

User plane traffic is directed towards MEC applications by proper placement and configuration of UPF functions. Authorized third-party cloud service providers can influence the placement and configuration of UPF by using the control interface exposed through NEF.

MEC for Serverless Computing and Cloud Integration for Massive IoT Devices

MEC will enable serverless computing for a key 5G use case, namely. massive IoT devices, by hosting Function as a Service (FaaS) in the edge and supporting integration with the cloud service provider as shown below. In this implementation model FaaS could be implemented through cloud wrapper MEC application(s) running on one or more MEC hosts and provisioned with the required resources, managed locally via an MEC service application. The initial traffic steering from the MEC AF will set up the routes such that the packets from a range of IoT devices are directed to the correct cloud wrapper MEC applications.

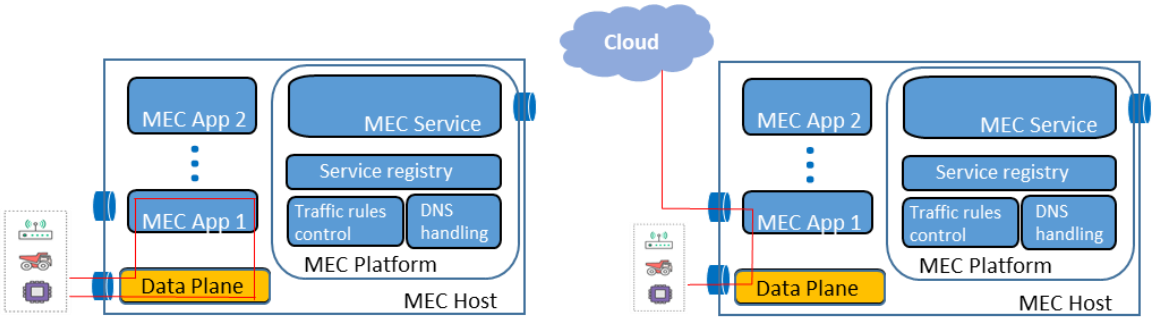


Figure 9. MEC host with and without break-out to cloud.

The egress traffic could be sent back to the IoT device, breakout to a cloud service provider from the cloud wrapper MEC application or transferred to another MEC application with the correct resources. The MEC application and/or MEC service may indicate to the AF to initiate traffic steering to an alternate MEC host with the required resources, if needed. Such indication can be made via traffic rule activation over the application enablement API.

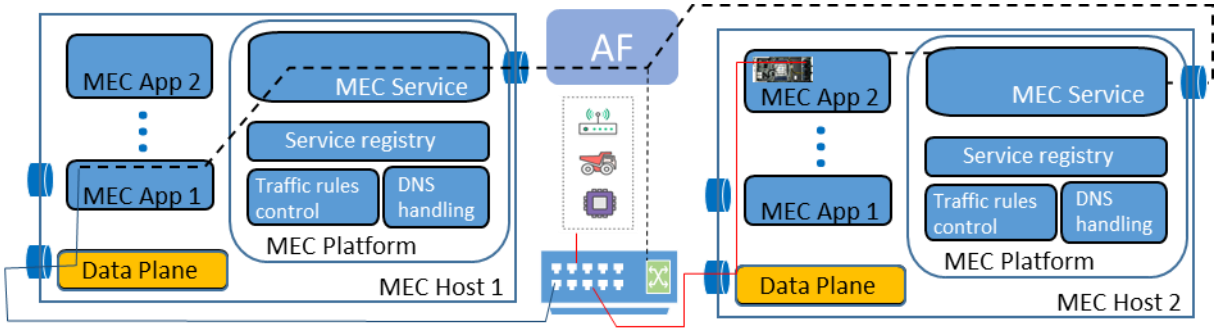


Figure 10. Traffic steering to alternative MEC host.

MEC for Enterprise Users

MEC is envisaged to play an important role in the way 5G penetrates enterprise connectivity and is used to support enterprise applications. Industry sectors such as healthcare, government institutions and fleet management are expected to benefit from MEC-based 5G applications.

However, in order to realize these benefits, the complexity and diversity of requirements associated with various types of enterprise use cases need to be addressed. Here, 5G capabilities, such as network slicing and flexible deployment of UPF, help enable flexible MEC-based deployment that is needed to host such different enterprise applications.

Deployment of MEC for enterprise applications in a corporate office environment could be applied as follows:

A corporate enterprise might require employees to access enterprise apps within the physical location of the corporation. The app can be used for tracking employees, sending training videos, security updates, etc. As explained in previous sections, MEC can exist as an AF within the 5G architecture. Within the 5G architecture, network slicing is very critical and will play an important role in the enterprise setting as well. This can be achieved through the NSSF function in 5G architecture. Within the corporate environment the requirements could be that the VPN connection to the corporate intranet should drop when the employee is out of the physical location. This can be provided by the UPF function in the 5G architecture. As the corporate location represents a limited physical area, the UPF serving that area can be collocated within the MEC site. Specific rules need to be configured in the PCF to provide traffic steering to the dedicated UPF when the enterprise users are trying to access the enterprise applications.

MEC for "Industrial IoT"

IoT has been one of the key drivers for the 3GPP 5G architecture and its requirements have been considered from the outset. There are multiple enablers specifically designed to serve a great variety of IoT use cases. For mission critical IoT services, there is the concept of URLLC (Ultra Reliable Low Latency Communications) that can be enabled by local processing in the Edge Cloud supported by the 5G architecture. The Edge Cloud is also a key component for Massive IoT, where huge amounts of data are processed near the source, where the data may originate from a massive number of sensors. Another key component that is extremely useful for IoT is network slicing, which allows offering dedicated resources for service tenants specifically tailored to their needs. The following presents a few Industrial IoT use cases that may well be deployed with MEC in 5G networks.

1. Security, safety, data analytics for Industrial IoT

This use case groups a number of innovative services for the operator or third party vendors based on the gathering of huge amounts of data (video, sensor information, etc.) from devices, analysed through a certain amount of processing to extract meaningful information before being sent towards central servers. Applications might run in a single location (i.e. on a single host) or be spread over a given area (e.g. campus coverage) or even in the whole network. In order to support the constraints of the operator or the third party requesting the service, the applications might have to be run on all requested locations (MEC hosts).

This use case describes an application running on a MEC host deployed close to the radio network that receives a very large amount of information from devices and sensors connected to the radio node associated with the MEC host. The application then processes the information and extracts the valuable

metadata, which it sends to a central server, likely deployed in a central cloud outside of the mobile network. A subset of the data might be stored locally for a certain period for a later cross-check verification.

A number of service scenarios can be enabled via this use case:

Security, safety: monitoring of an area for specific events, such as abandoned luggage, authorized access (e.g. with face recognition), car park car monitoring, etc.

Big data: massive sensor data pre-processing, smart factory, etc.

For any of these scenarios the local information can be complemented for example with device location tracking.

2. Active device location tracking

This use case enables real-time, network measurement-based tracking of active terminal equipment (independent of GPS) using 'best-in-class' geo-location algorithms.

The deployment of this use case in a MEC system provides an efficient and scalable solution with local measurement processing. It enables location-based services for enterprises and consumers (e.g. on opt-in basis), for example in venues, retail locations and traditional coverage areas where GPS coverage is not available.

3. Application computation off-loading

In the application computation off-loading use-case, an application in the MEC host executes compute-intensive functionalities with high performance on behalf of mobile devices. By providing rich computation resources on a MEC host, application computation can be off-loaded there, to be accelerated even if a user uses relatively low performance devices, satisfying the user experience regardless of the type of UE.

This use case is effectively used for especially computation-hungry applications such as graphical rendering (high-speed browser, Augmented Reality (AR), Virtual Reality (VR) and 3D gaming, etc.), pre-processing of data (sensor data cleansing, video analytics, etc.), and value-added services (language translation, log analytics, etc.). One example of application computation offloading is the Edge Accelerated Browser (EAB). Most parts of the browsing functions, such as Web content evaluation, rendering and optimized transmission, are off-loaded to the MEC application, while the UE just renders reconstituted browser graphics on its display. Again, by transferring any compute-intensive process from a UE to a MEC host to accelerate an application, a rich application experience is made available on various types of mobile devices.

Conclusions

ETSI has published the baseline MEC standards to allow a standards-based environment for cloud applications in the network edge. Application enablement API and the MEC service APIs are the essential components in the MEC specification for a unified, standards-based environment for context-aware cloud applications. With this set of APIs an application can produce its intended function and it can also produce services for other applications, and discover and consume services produced by other applications and the MEC platform. The APIs for application lifecycle management and for the UE application interface are the other essential part of the published MEC specification. These two APIs are for application lifecycle management. The APIs for MEC application mobility are an ongoing effort in ETSI ISG MEC. New MEC service APIs are also being developed for specific industry applications such as V2X to allow MEC better serve and add value to these applications.

The 3GPP 5G system specification of Rel-15 includes native enablers for edge computing. This white paper has illustrated the potential of these enablers for an integrated MEC deployment in 5G networks. The key components of this integration are the ability of MEC, as a 5G AF, to interact with the 5G system to influence the routing of the edge applications' traffic and the ability to receive notifications of relevant events, such as mobility events, in the 5G system for improved MEC deployment efficiency and end user experience. Moreover, the versatility of the 3GPP service exposure and API frameworks in principle also allows MEC to provide services to the 5G system.

This white paper has demonstrated the benefits of deploying MEC on the N6 reference point of the 5G system. This deployment is enabled by flexibility in UPF location. It is the UPF that implements the data plane for MEC hosts. This data plane is controlled by the SMF of the 5G system and is influenced by MEC as a 5G AF. Whether the physical deployment of MEC is in the RAN or in the core network or somewhere in-between, the role of the UPF remains; it is the data plane for the MEC host and it integrates the MEC application traffic in the 5G bearer stratum, thus enabling charging and LI&RD for that traffic.

The value of ETSI MEC standards remains the same regardless of the way in which MEC is deployed in the 5G system. ETSI MEC standards support a unified, standards-based application environment where the applications are exposed to all application enablement and network information and capabilities through standardized MEC APIs. To sum up, while there are many options to deploy, there is only one application environment for applications to experience.

Standardization of the APIs to expose the required 3GPP system capabilities assumed in this white paper is a work in progress in 3GPP. Interested parties are invited to contribute to this work.

List of abbreviations

3GPP	3rd Generation Partnership Project
4G, 5G	4th, 5th generation of mobile networks
ADMF	Administration Function
AF	Application Function
AMF	Access and Mobility Management Function
API	Application Programming Interface
APN	Access Point Name
APP	Application
AR	Artificial Reality
AUSF	Authentication Server Function
CN	Core Network
CU	Centralized Unit (of RAN)
CUPS	Control/User Plane Separation
DNAI	Data Network Access Identifier
DNN	Data Network Name
DU	Distributed Unit (of RAN)
EAB	Edge Accelerated Browser
EPC	Evolved Packet Core network
ETSI	European Telecommunications Standards Institute
IoT	Internet of Things
IP	Internet Protocol
LEA	Law Enforcement Agency
LEMF	Law Enforcement Monitoring Facility
LI	Lawful Interception
MANO	Management and Orchestration
MEC	Multi-access Edge Computing
MEP	MEC Platform
MNO	Mobile Network Operator
NEF	Network Exposure Function
NFV	Network Functions Virtualization
NSSF	Network Slice Selection Function
NRF	Network Repository Function
PaaS	Platform as a Service
PCF	Policy Control Function
PDN	Packet Data Network
QoS	Quality of Service
RAN	Radio Access Network
RD	Retained Data
RNIS	Radio Network Information Service
SBA	Service Based Architecture
SMF	Session Management Function
S-NSSAI	Subscribed Network Slice Selection Assistance Information
UDM	Unified Data Management
UE	User Equipment
UL	Uplink

UL CL	Uplink Classifier
UPF	User Plane Function
V2X	Vehicle to Everything
VNF	Virtualized Network Function
VR	Virtual Reality

References

[1] ETSI GS MEC 003 V1.1.1, “Mobile Edge Computing (MEC); Framework and Reference Architecture” (2016-03)

[2] ETSI GS MEC 010-1 V1.1.1, “Mobile Edge Computing (MEC); Mobile Edge Management; Part 1: System host and platform management” (2017-10)

[3] ETSI GS MEC 010-2 V1.1.1, “Mobile Edge Computing (MEC); Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management” (2017-07)

[4] ETSI GS MEC 011 V1.1.1, “Mobile Edge Computing (MEC); Mobile Edge Platform Application Enablement” (2017-07) ETSI GS MEC 011

[5] ETSI GS MEC 012 V1.1.1, 2Mobile Edge Computing (MEC); Radio Network Information” (2017-07)

[6] ETSI GS MEC 013 V1.1.1, “Mobile Edge Computing (MEC); Location API” (2017-07)

[7] ETSI GS MEC 014 V1.1.1, “Mobile Edge Computing (MEC); UE Identity API” (2018-02)

[8] ETSI GS MEC 015 V1.1.1, “Mobile Edge Computing (MEC); Bandwidth Management API” (2017-10)

[9] ETSI GS MEC 016 V1.1.1, “Mobile Edge Computing (MEC); UE Application Interface” (2017-09)

[10] 3GPP TS 23.501 V15.1.0, “3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; System Architecture for the 5G System; Stage 2 (Release 15)” (2018-03)

[11] ETSI White Paper “MEC deployments in 4G and evolution towards 5G”, February 2018 (http://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp24_MEC_deployment_in_4G_5G_FINAL.pdf)

[12] 5GAA White Paper “Toward fully connected vehicles: Edge computing for advanced automotive communications”, December 2017 (http://5gaa.org/news/toward-fully-connected-vehicles-edge-computing-for-advanced-automotive-communications/)

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

ETSI White Paper No. 30

MEC in an Enterprise Setting: A Solution Outline

First edition – September 2018

ISBN No. 979-10-92620-25-2

Authors:

Alex Reznik, Editor (HPE), Anthony Sulistio (Bosch), Alexander Artemenko (Bosch), Yonggang Fang (ZTE), Danny Frydman (Saguna), Fabio Giust (Athonet), HuaZhang Lv (China Unicom), Saad Ullah Sheikh (STC), Yifan Yu (Intel), Zhou Zheng (Huawei)

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

Contents

Introduction

Multi-access Edge Computing (MEC) complements the corporate data centre by providing compute, storage, networking and data analytics at locations closer to the data source (e.g. Internet of Things (IoT) devices, , workers, operators, etc.) and points of consumption [1]. For example, MEC solutions enable enterprises to manage their security and compliance requirements effectively, since data analysis can be performed in a more deterministic manner and the data are kept within a specific premises or political region compared to the centralized data centre. Additionally, enterprise MEC solutions are closely "integrated" with access network(s) to provide a fully-converged enterprise access and compute environment. Such environments often include enterprise WiFi. However, they may also include mobile access, especially when large campus and outdoor coverage is required. An example is a "private LTE" network which can use licensed spectrum, unlicensed spectrum or new technologies such as Citizens Broadband Radio Service (CBRS).

Having a well-defined and structured set of functionalities, a MEC-enabled enterprise infrastructure supports both scaling options: (i) Small scale application scenarios in small enterprises benefit from easy deployment, setup and scalability advantages of edge computing solutions; (ii) middle to large scale scenarios will boost the enterprise size by using flexible and full-automated management and orchestration functions.

The environment of a MEC deployment can differ for each location depending on the use cases and the digital services that are offered to end users, which are sensitive to network latency and require a high level of performance. However, most of the existing solutions are done on a rather ad-hoc and proprietary basis for a specific environment, with minimal adherence to standards and/or interoperability. To avoid building a MEC solution from scratch for each location, the ETSI work on MEC aims to address this problem.

Due to historical reasons, many companies utilize a large set of heterogeneous technologies in different domains, including communication networks and data processing. By introducing MEC, the unification of the applied interfaces will enable a new level of interoperability for various components from different vendors and facilitate a natural reuse of underutilized capacity elements (like network, storage, processing, etc).

The purpose of this white paper is to give a solution overview of MEC deployments in the enterprise environment. Firstly, this paper presents several use cases and MEC deployment options. It then highlights key challenges when trying to deploy these use cases in an existing enterprise infrastructure. In addition, it demonstrates how the ETSI MEC APIs help to overcome these challenges.

Use Cases and Deployment Examples

Enterprise Solution Use Cases

Use Case A: Smart Enterprise Building

Enterprises already invest a lot in smart buildings since they are increasingly valuable for a company's facility management and its employees. Some of the advantages gained from smart buildings are:

- Reduced energy and utility costs and thus lowering the carbon footprint,
- Improved building operations from sensors data collected for predictive maintenance,
- Increased occupant comfort and productivity by having personalization of the working environment, such as room ambience, temperature and lightning,
- More efficient use of space by providing live data of available desks and meeting rooms,
- New level of working experience and collaboration opportunities.

A smart building contains many IoT devices and sensors, such as motion, light, temperature, humidity, infrared, video, etc. A typical application is to continuously monitor energy consumption, room occupancy, parking spaces, temperature, coffee machines, etc. The information is shown in a central dashboard for facility management. The building's occupants can access relevant information via a smartphone app for personalized workspaces, finding empty desks, booking a meeting room at a short notice, etc.

As these sensors generate gigabytes of data (depending on the size of the building). The facility management deals with an operational challenge in capturing, processing and storing data (also for historical data). A local processing in the edge (e.g. per floor) rather than a centralized data centre reduces network latency and analyses the data more quickly. Thus, MEC provides a better user experience.

Another challenge is that the IoT devices and sensors are heterogeneous, coming from various manufacturers. The devices need to communicate through standardized protocols and the smartphone app needs to interact with the local edge or IoT gateway via open Application Programming Interfaces (APIs). The ETSI work on MEC aims to address this challenge.

Use Case B: Data Analysis and Security

In regulated industries, such as finance and healthcare, data need to be stored and analysed on-premises in order to comply with local regulations. For example, financial institution branches can find non-compliant transactions in real-time and stop them more quickly, compared to sending the data to a central data centre [2]. In addition, the local branches can provide online digital services to their customers by transforming ATMs with a video capability into interactive tellers [3], [4]. Using MEC, enterprises with sensitive digital assets can protect the security and integrity of their data by providing real-time security monitoring for traffic anomalies [5].

Use Case C: Augmented Reality Conferencing

Telephone conferences represent a vital communication means in every enterprise giving a high level of flexibility and location independence to workers. A transition from pure voice to video plus voice conferences increases a feeling of close human presence which remains an important factor of successful

collaboration within working teams. The next level can be reached using augmented reality (AR). With AR, an immersive user experience helps to reach a feeling of real presence. However, higher requirements on latency and image quality present a showstopper for today's mobile devices. Using offloading of image processing into the edge, almost unlimited possibilities arise.

Use Case D: Location-restricted BYOD access

A large enterprise with a large mixed indoor/outdoor campus (e.g. an automobile test and assembly facility) is providing Bring Your Own Device (BYOD) access to numerous enterprise applications to its employees. The access is available only when employees are on-site, but in such cases it is automatic. The employees' devices and mobile identities are mapped to appropriate enterprise identities allowing for proper application of enterprise access policies. This happens over an indoor WiFi network as well as an outdoor LTE-based access. Enterprise traffic does not leave the enterprise premises. Non-employees are able to use LTE-based data access, however this happens over a separate "network slice" – none of the enterprise network assets are visible to non-employees.

Use Case E: Streaming media and entertainment in Enterprise

An end-to-end streaming media solution suite is deployed locally within each enterprise location, which could mean a building or a portion of a building. This solution supports 4K/8K video playback, in-campus video conferencing, and a number of applications which include AR/VR. Minimizing the amount of streaming media that has to leave the boundaries of the location improves the overall experience, while reducing the traffic on the enterprise WAN – thus reducing both costs to the enterprise and to the carrier.

An Example of an Enterprise MEC Deployment

Enterprise ICT services target a clear and usually well-controlled set of subscribers, i.e., staff members and other authorized collaborators. In addition, they are inherently localized within the premises of the enterprise and/or where the core operations are carried out, and are designed for very specific purposes, sometimes implemented by tailor made solutions. For these reasons, MEC appears as a natural partner technology to provide edge computing and communication infrastructure to enterprises.

In this section, we highlight what such an implementation may look like using an example that captures much of the complexity associated with an enterprise deployment, keeping in mind that many enterprise deployments may actually be simpler than this example. Consider the overall system shown in Figure 1 and let's take what happens at the AR/VR terminal as an example. The VR terminal uses the 5G NR air interface (gNB) to access local application content on the MEC edge business platform. Video transcoding processing and cloud game graphics calculation and rendering are all performed at the edge site, avoiding the need to upload the business flow to the centralized cloud in the Internet. Because the MEC edge business platform is an extension of the cloud platform in the internet, it does not require customized development of apps, but rather can run application components of well-designed apps "as-is.[1]" This enables rapid deployment and iteration of applications.

The MEC management platform is deployed in a local or regional data centre, which enables the coordination and management of MEC business platforms across the enterprise.

[1] Please see [6] for some discussion of what a well-designed application is

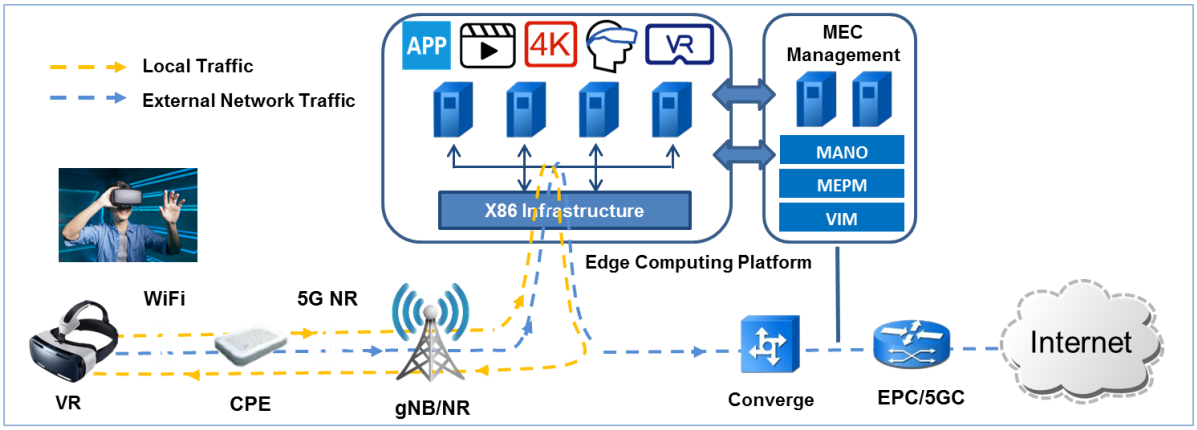


Figure 1: Illustration of an in-building streaming media system

Focusing now on each local MEC site, we note that it is a miniaturization of a full data centre; a detailed diagram is shown in Figure 2. A typical scale involves 10 to 20 enterprise-grade x86-based compute nodes with built-in storage. These are used for general computing, as well as network functions (thus distinguishing them from a traditional enterprise-owned cloud). A dedicated cluster, e.g. for storage, video transcoding or AI can be added within the framework as necessary.

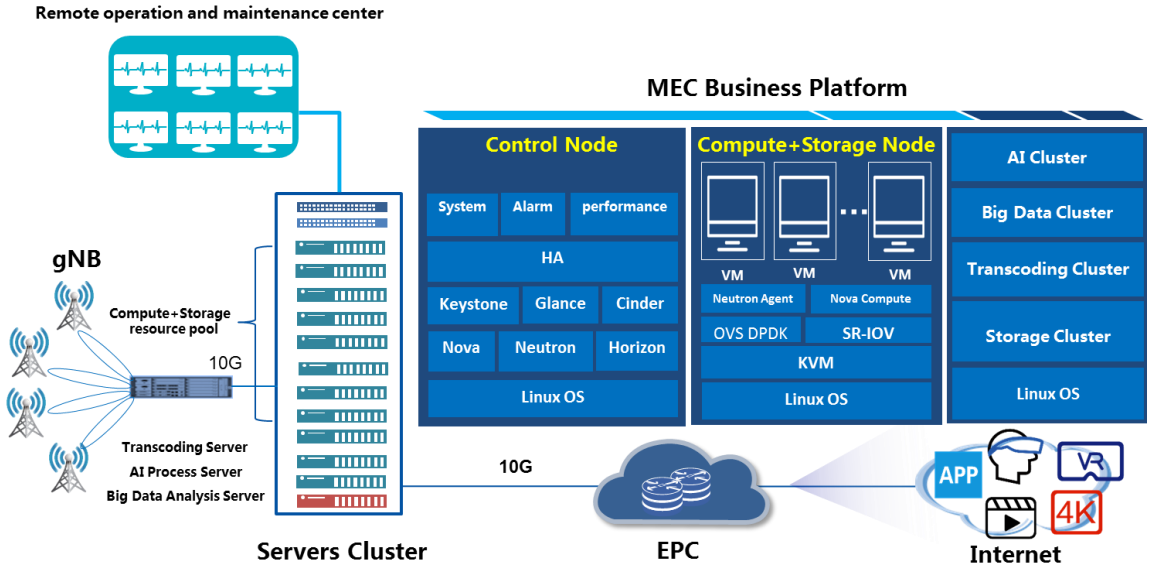


Figure 2: Detailed diagram of a MEC site

Service Enablement Challenges in MEC

It is widely accepted that the enterprise market, as enabled by MEC, is going to be a major component of the emerging 5G mobile ecosystem. Moreover, many in our industry hold the opinion that enterprise will be the lead market for 5G. However, integrating mobile networks and enterprise networks continues to involve a number of challenges. Some of the key ones, especially as related to MEC, are listed below.

Supporting Enterprise-Grade MEC Applications

Enterprise IT application development has increasingly been shifting towards a micro-services based application architecture that heavily utilizes containers. Such an approach holds additional benefits for MEC applications, as pointed out in our earlier white paper on software development for MEC [6]. However, existing telecommunication network architectures, including MEC specifications, often assume (implicitly or explicitly) that a virtual machine-based NFV management environment is in place. For true enterprise applications, container-based services must be supported in MEC.

Unity of experience across all networks (Fixed /3G /4G /5G / Wi-Fi)

MEC will deliver services to many enterprises that require applications to be delivered in a manner agnostic of network access type. Having said this, each wireless network type has its unique delay/latency characteristics. One of the main challenges for enterprises will be to ensure the same experience across different access technologies.

A related challenge concerns enterprises that operate a large number of remote locations and/or mobile employees. Here, a unity of experience must also be provided over a large number of distinct types of public internet accesses, e.g. public mobile access, public WiFi, etc.

Integration of Access Control

In order to properly support enterprise applications, operator-offered MEC solutions must be able to "understand" the world of enterprise identity and access management. Unfortunately, enterprise identity and access management is based on completely different technologies to those used in mobile networks. Mobile networks utilize 3GPP-defined SIM-based approaches, while enterprise networks are built around systems such as LDAP, IdAM, etc.

A key challenge will be to devise techniques for identity and access management that are able to "connect" an operator's, in particular a mobile operator's subscriber management systems and an enterprise access and identity management system in a way that is acceptable to both – i.e. taking into account that the MEC system (i.e. the MEC platform and applications running on it) may not be considered a trusted entity by either party.

Furthermore, this creates an opportunity for operators to offer common authentication /identification in both mobile and enterprise segments "as-a-service" providing additional added value to enterprises.

High Bandwidth Content Optimization and QOE enablement

Many of the enterprise use cases will rely on video and similar high bandwidth content. In the case of mid to large size organizations, the cost associated with sufficient throughput to support such applications is a significant challenge. Translating this into an operator-provided MEC system supporting multiple applications means solving this issue potentially for multiple enterprises at a time – while keeping the

traffic of each enterprise fully differentiated and separate from each other and from public network traffic.

Enterprise Operations & Maintenance requirements

Every enterprise requires the ability to monitor and control its assets. The same will apply to enterprise applications running in MEC clouds. This means that MEC clouds will require O&M tools and solutions like those in use today for other public clouds – but adapted to the unique nature of MEC as a highly distributed collection of smaller mini and micro clouds. This issue is particularly acute in those cases where enterprise locations include hard to reach areas. For example, oil/gas pumping sites are often remote, poorly connected and many – an important and highly challenging case of highly distributed infrastructure where effective O&M is critical.

Addressing the MEC Enterprise Challenges

Supporting Enterprise-Grade MEC Applications

Enterprise deployments of MEC are expected to require co-location of operator-managed network functions and enterprise-managed IT applications on a shared infrastructure. Such deployments need to satisfy several requirements.

Co-existence with NFV management framework, such as ETSI NFV

Virtualizing network functions is subject to very different performance and management requirements than virtualization of traditional IT applications, see [8] for a detailed list of NFV requirements. The upshot of this different set of requirements is the development of an understanding that virtualizing network functions represents a different type of virtual application and that the infrastructure for enabling such functions must be different as well. This different approach to virtualization is now called Network Functions Virtualization (NFV).

However, this does mean that co-located NFV and enterprise applications on the same infrastructure means that the infrastructure management framework must be able to deal with an additional layer of complexity. Specifically:

- Because network functions and enterprise applications belong to two different domains of trust (the carrier's domain and the enterprise domain), they must be located in well-separated tenant spaces (and, if possible on separate physical resources). Communication between these domains must be enabled using separate LANs (virtual and physical) with appropriate security infrastructure deployed (firewalls, policy-based routing, etc.) For management of virtual network functions, a useful resource is ETSI MEC's report on integration with the NFV management framework [9].
- Notwithstanding the above requirement, we must recognize that the shared physical infrastructure ultimately has a single owner (enterprise or carrier) and that this owner must be able to manage the infrastructure as a whole – preferably in a fairly dynamic fashion so as to be able to realize the benefits resulting from the flexibility of virtualization.

This leads to a layered approach towards management of enterprise-based MEC deployments. Each entity maintains its own management framework which has control over one or several tenant spaces allocated to it. The enterprise management framework can be based on traditional enterprise tools, while the carrier management framework can be based on traditional NFV management tools. However, in both cases, the management framework must limit its "scope" to management of virtual infrastructure (vCPUs, volumes, vLANs, etc.) assigned to it. In addition, the owner of this physical infrastructure maintains a third management framework for the physical infrastructure. Its scope must be limited to management of physical infrastructure and allocation of virtual resources to each of the tenants using the physical infrastructure. A well-defined simple API framework is required for this infrastructure and a good practice is never to integrate it with one of the virtual management frameworks – even when the two happen to be operated by the same entity (e.g. both are operated by the carrier).

Support of multiple approaches to virtualization

As we all know, modern approaches to virtualization have evolved from a single, Virtual Machine-based approach, to several, notably including containerization and serverless compute. It is widely believed that both the NFV and MEC management frameworks are mostly agnostic to the virtualization type. However,

it is likely that each could benefit from certain optimizations that are specific to a virtualization approach and both ETSI MEC and ETSI NFV are in the process of studying this topic.

The ongoing study in ETSI MEC includes the gaps in currently defined MEC functionalities when running MEC applications as containers. There are several use cases considered in the study, which includes containerized MEC application packaging, on-boarding, instantiating etc. The MEC study is expected to take into account the requirements of application developers and identify gaps in existing MEC specifications. The study is expected to be published in the first quarter of 2019.

Unity of experience across all networks (Fixed /3G /4G /5G / Wi-Fi)

An enterprise network is a private network. It could be large and geographically distributed across multiple cities/countries in the world, or it could be as small and localized as a single office. Each enterprise may choose a different approach to build up its networks based on its size and business requirements.

Figure 3 illustrates an example of enterprise network deployment, consisting of several “zones”:

- The headquarters, where the core business services are located.
- Satellite offices, with local enterprise networks being inter-connected with the headquarters cloud through secured backhaul networks provide accessibility for enterprise employees to access the enterprise services. An enterprise network may use 4G/5G small cells for the outdoor coverage, Wi-Fi networks for the indoor coverage and fixed access for static devices.
- Remote employees, which access enterprise services using VPN over the public Wi-Fi or cellular networks.

A MEC host provides a computing environment with networking interfaces for running applications. In order to meet its service requirements, an enterprise may distribute services from its central cloud to MEC hosts deployed at the edges of networks so that the timing critical or bandwidth consuming applications can be run very close to the device’s location. Some applications may only need to run over local area networks. This may require the MEC system to support all the access network connection types.

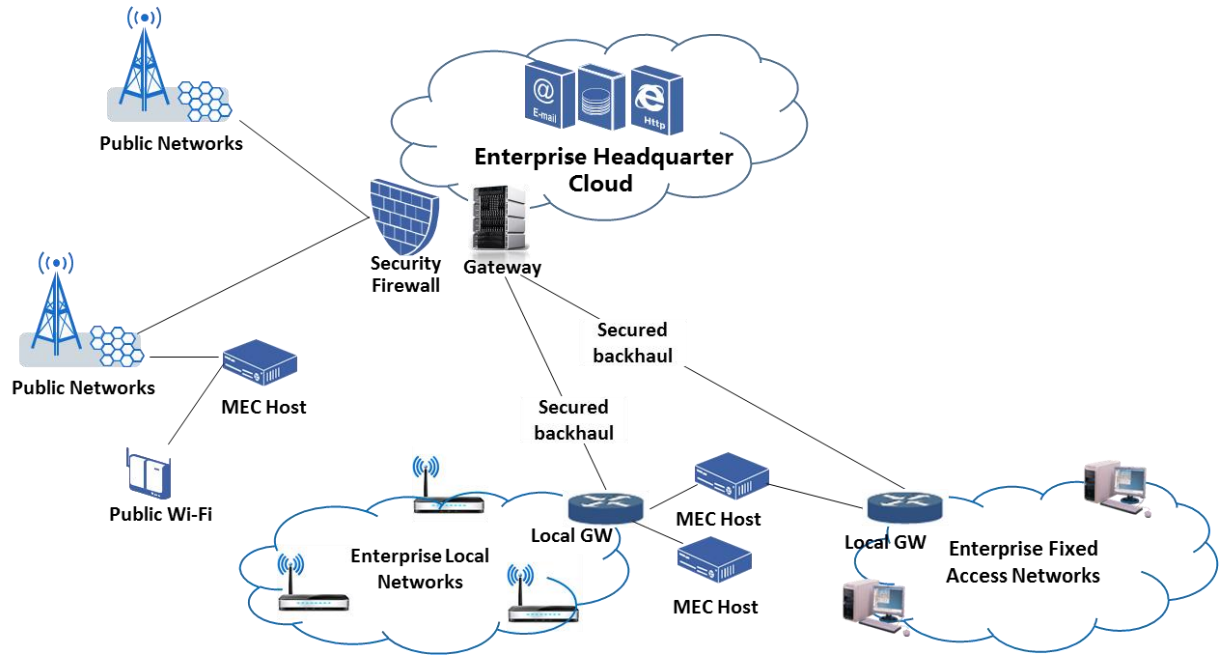


Figure 3: MEC deployment across different enterprise networks

Orchestration and optimization of enterprise applications

Different enterprise applications may have different characteristics and requirements. For example, augmented reality conferencing may be latency critical and bandwidth consuming, while location-restricted BYOD may require a seamless service when employees move in the enterprise campus across different access networks. Multiple types of enterprise networks provide an opportunity for MEC to orchestrate and optimize the performance of enterprise applications. In order for the MEC system to deliver a unity of experience over a large number of distinct types of access networks, it may require information on enterprise application requirements and information on access networks.

Table 1 shows an example of characteristics of different types of access network. Table 2 provides an example of characteristics for enterprise use cases described in above and possible access networks for running applications.

Table 1: A example of characteristics of access networks for enterprise

	4G LTE	5G NR	Wi-Fi	Fixed Access
Frequency band	Licensed	Licensed	License exempt	NA
Max data rate*	>100Mbps (DL) >50Mbps (UL)	10Gbps	6.9 Gbps (802.11ac) 9.6 Gbps (802.11ax)	Variable
Min latency*	10ms (air interface)	1ms (air interface)	Varies as the access loading	Variable
Session continuity	It supports “make before break”	It supports “make before break” inter-cell handover.	It supports the “break before	NA

	inter-cell handover.		make" for ESS inter-AP handover.	
Note 1: the maximum data rate depends on the bandwidth of the operational channel. Note 2: the minimum latency refers to one-way physical layer latency.				

Table 2: An example of enterprise application characteristics

Enterprise Usecases	Characteristics		
	Data rate	Latency	Possible access
Smart enterprise building	Variable	Variable	Wi-Fi or Fixed
Data analysis and security	> 20Mbps	Variable	Wi-Fi or Fixed
Augmented reality conferencing	100Mbps – 9.4Gbps	< 5ms	Wi-Fi or Fixed
Location-restricted BYOD access	Variable	Variable	Wi-Fi, Small Cell, Fixed
Video streaming	> 25Mbps	Variable	Wi-Fi or Fixed

Based on enterprise application requirements and access network characteristics, the MEC management could optimally orchestrate and schedule enterprise applications running on a MEC host close to devices' locations over one or multiple appropriate access network connections. For enterprise applications like augmented reality conferencing, MEC management may choose a MEC host with wide bandwidth WiFi or fixed access to instantiate the application for delivery of the service to enterprise users. For the location-restricted BYOD access, enterprise applications may only be on-boarded to the MEC host at a specified location. Therefore only on-site employees can receive the services produced from those enterprise applications over local enterprise networks.

Unity of MEC APIs across enterprise networks

ETSI ISG MEC is developing a series of API specifications for different access networks:

- GS MEC 012 [14] specifies the APIs for radio network information service (RNIS). This specification defines an API that provides access to a large amount of network information for a 3GPP-defined network.
- GS MEC 028 is a specification under development that will specify the APIs for WLAN information service (WIS) and which will serve a purpose for WiFi networks that is similar to that of [14] for 3GPP-based access.

- GS MEC 029 is a specification under development that will specify the APIs for Fixed Access Information Service (FAIS) and which will serve a purpose for Fixed-Access networks that is similar to that of [14] for 3GPP-based access.

These APIs provide facilitate the unity of service interfaces by providing a common standardized service access to enterprise applications. Moreover, by providing information on the status of the access network, these APIs assist applications and service orchestrators in properly configuring and mapping applications across available access networks.

Integration of Access Control

A fundamental MEC operation is the ability to forward traffic between the access network and an application instance on a MEC host. ETSI MEC specifications enable this operation by specifying a traffic filtering service that a MEC Platform (MEP) must provide. The service is specified in ETSI GS MEC 011 [10]. The most common approach to indicating which traffic to forward is to use the IP 5-tuple: the transport protocol (TCP/UDP/etc.) and the source and destination IP addresses and port numbers. However, ETSI MEC recognizes that a number of other means of traffic filtering may be of use. In particular, when the access network is a 3GPP mobile access network and the traffic is encapsulated in GTP tunnels, filtering by GTP tunnel parameters may be of use as well. The TrafficFilter data type ([10], clause 6.5.6) supports all these capabilities. Additionally, [10] effectively enables filtering by web names (Fully Qualified Domain Name - FQDN) by defining a DNS service which returns a set of IP address for an FQDN and these IP addresses can then be used to define a traffic rule.

However, in the case of enterprise services, an additional filtering capability is required – filtering by an "enterprise user" – i.e. a service where all traffic associated with a particular enterprise user is directed between the access network and the enterprise application. A simple solution would be to use some service to look up all the IP flows associated with a particular user and set up IP-based traffic filters for all such flows. Indeed, in the mobile network such a service is readily available – e.g. the MME in 4G networks. Unfortunately, the association of IP flows is made to the mobile identity (e.g. IMSI) and not the enterprise identity – which highlights the need (as noted previously) for a way to associate such identities.

A naïve solution would be to create a table mapping mobile identities and enterprise identities. If a mobile number (MS-ISDN) is sufficient and the enterprise is willing to maintain a mapping of user identities and their MS-ISDNs then this is a sufficient approach. However, in some instances the MS-ISDN cannot be used and a mapping to IMSI (the actual mobile network identity) is required. This creates a problem: IMSI is a critical "identity asset" within the mobile network, much as an enterprise user identity (e.g. an LDAP identity) is within an enterprise network. Neither can be expected to share its identity with the other – doing so would be a major violation of standard security practices and expose both the mobile network and the enterprise to significant potential security risks. Unfortunately, this means that a naïve direct mapping is not a feasible solution.

The problem we are describing can be viewed as a special case of a well-known single-sign-on (SSO) problem and SSO techniques can be used to solve this problem. Essentially, a mutually trusted entity generates a stand-in value – a *token*, which is used for the following purposes:

- The MEC System is able to associate the *token* to an access network identity; however the *token* does not reveal the access network identity to any entity that is not "trusted" by the access network operator

- The enterprise application is able to associate the *token* to an enterprise identity; however the *token* does not reveal the enterprise identity to any entity that is not "trusted" by the enterprise.

Assuming that such a *token* can be defined, ETSI MEC has specified the means to use it for traffic filtering as follows:

- ETSI MEC 014 [11] defines an API by which such a *token* can be made available to the application.
- ETSI MEC 011 [10] supports traffic filtering by *token* (see clause 6.5.6).

What remains, therefore is how to define such a token. While the specific approach is left up to the design of each system, several well-known and standardized means are available. Below are a few examples:

- **Using MS-ISDN.** As noted above, the public phone number (or, more broadly, MS-ISDN) is one means to identify a user. Although not fully secure – as it is not secret – it may be sufficient for some applications. In this case, the MS-ISDN becomes the *token*; the enterprise maintains the mapping between the MS-ISDN and enterprise identities and the MEP is able to associate MS-ISDN and IP flows (typically by invoking services provided by mobile network entities such as the MME).
- **Using SIM-based authentication, such as EAP-SIM.** EAP-SIM [12] is a well-known protocol developed for the purposes of integrating authentication and access control mechanism between WiFi and 3GPP systems. The protocol is in use in HotSpot 2.0 systems (marketed under the Passpoint® brand) as defined by the WiFi Alliance. It uses a mobile device, which includes both a UE functionality with a SIM module for 3GPP access and WiFi functionality which relies on the AAA server and EAP protocol for access. As part of the EAP-SIM access procedure, a SIM-based "key" is generated to be used as the "master key" in the WiFi keying system. Because it is SIM-based, it can be generated by the mobile network and made available to MEP. Thus, it can be used as a *token*. The enterprise application is provided this token by an appropriate enterprise agent on the client device using secure means that are enterprise specific. Note that for the purposes of MEC *token* generation, the procedure can be ran without any WiFi based interaction – it is just being re-used for a different purpose.
- **Using 3GPP's Generic Authentication Architecture / Generic Bootstrapping Architecture (GAA/GBA).** GAA/GBA are 3GPP defined mechanisms allowing a mobile operator to become a provider of SIM-based SSO services. Should an enterprise take advantage of these services, any of the GAA/GBA generated identities/keys available to the enterprise can be used as a *token*.
- **3rd party SSO Providers, such as OAuth.** Authentication by-products of 3rd party SSO providers can be used as *tokens* – provided that both the enterprise and the access network operator agree to use the same SSO provider. As with GAA/GBA, any of a number of by-products of the SSO access procedure known to both entities can be used as tokens.

High Bandwidth Content Optimization and QOE enablement

High-bandwidth applications, such as video conferencing and video streaming, continue to suffer from the mismatch between the network design and the application demand. For example, it well known that traditional TCP congestion control is designed with a view towards wired networks and highly heterogeneous traffic. While cross-layer optimization across architectural boundaries remains the wrong approach for broad-use public applications and for consumer products, in the context of an enterprise network it is perfectly acceptable and can bring about significant efficiencies in network performance, resulting in both user QoE improvement and IT cost reductions. The challenge is then, to achieve such

vertical cross-layer optimization given a system designed from general purpose consumer components (HW and SW) – and to do it in the proper location within the system.

MEC can facilitate this by helping an application to identify the type of service and user profile in accordance with user data packets, and then by defining the QoS information for appropriate information flows that can be propagated into the access network (e.g. an LTE eNB). By integrating OTT service information and radio access network information, the MEC platform can use the advantages of intelligent channels to guarantee QoS of key users and services. Figure 4 shows an example of this.

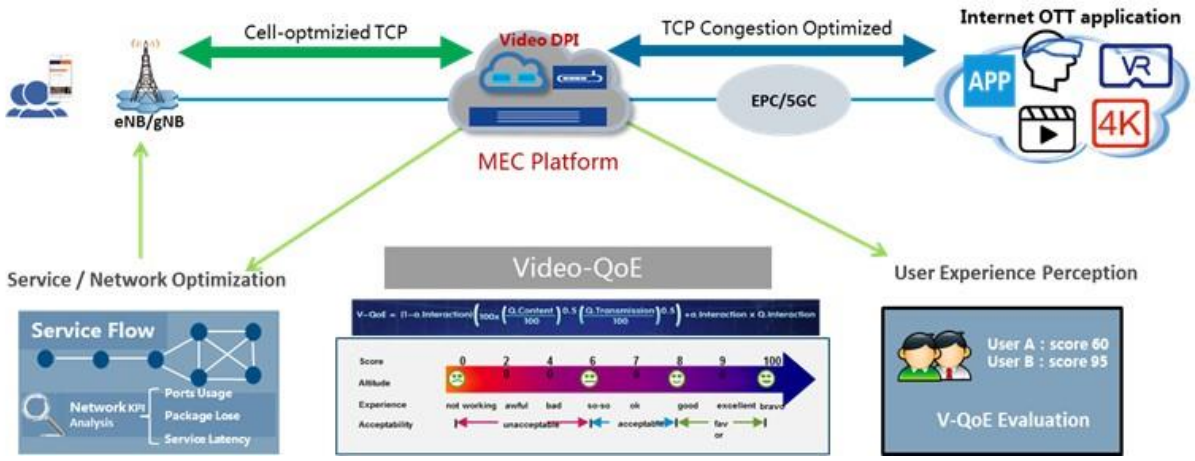


Figure 4: Example of a MEC-based Video-QoE optimizing application

How does ETSI MEC help achieve this? First the "Video DPI" component of the MEC service requires information about the state of the access network, which it can obtain using RNIS [14] for a 3GPP-based network (as shown) or using the upcoming WiFi and Fixed-Access information service APIs. Furthermore, user information in a particular edge site can be obtained using the Location APIs [15] and mapped to enterprise identities as discussed above. These can then be used to filter by specific user traffic, which allows such traffic to be operated on. Finally, the BW Management set of APIs, as defined in ETSI GS MEC 015 [20] enables the definition of QoS parameters to the various traffic flows and thus achieving the necessary goals as defined by the "computation" block in Figure 4.

Enterprise O&M requirements

As noted above, a key concern with operation and management of edge clouds is the highly distributed nature of these clouds and the fact that the communication links on which the O&M operations rely may be unreliable, or latency and throughput constrained. This issue is widely recognized, see e.g. [19]. This means that a successful O&M approach requires the following:

- A partitioning between a centralized system-wide orchestration and a localized on-edge-site management entity for implementation of decision.
- A well-defined secure set of APIs between these entities that is designed to be robust to communication links that may be unreliable, or latency and throughput constrained.

ETSI MEC facilitates the implementation of such systems, firstly by defining a reference architecture [16] that defines an on-the-host MEP Management Entity (MEPM) and a centralized, system-wide orchestration function (MEO). Additionally, REST-ful APIs for the management of the MEC platform [17] and the applications running at a MEC site [18] are defined with the requirement of robustness to imperfect communication links (none of the APIs are latency sensitive nor require significant throughput).

Summary and Conclusions

Enterprise is a key focus area for edge computing and represents most of the early deployments of edge computing. However, integration of edge computing in an access network presents a number of challenges that go beyond the typical issues that enterprises deal with. In this paper we have highlighted some of the key such challenges and outlined approaches to solutions. Clearly, it is not possible to provide detailed solutions in a short white paper, moreover a good solution should always take the specific needs and characteristics of each enterprise into account. However we do hope that this paper helps its readers in designing an appropriate solution. Additionally, we hope that by illustrating how ETSI MEC specifications enable such solutions in a simple, industry-standard interoperable way, we can encourage enterprises, operators and vendors to think of enterprise edge as a highly scalable market where much can be reused and duplicated despite the need to design to the specifics of each customer.

References

All ETSI MEC Specifications listed below can be accessed via: https://www.etsi.org/technologies-clusters/technologies/multi-access-edge-computing

[1] "Edge Computing Will Decentralize but Complement Traditional and Cloud-Based Data Center Architectures," [Online]. Available: https://itcblogs.currentanalysis.com/2018/04/09/edge-computing-will-decentralize-but-complement-traditional-and-cloud-based-data-center-architectures/ . [Accessed April 2018].

[2] "Why Edge Computing Is Here to Stay: Five Use Cases," [Online]. Available: https://www.rtinsights.com/why-edge-computing-is-here-to-stay-five-use-cases/ . [Accessed April 2018].

[3] "A Local Edge Lifecycle to Create Competitive Differentiation," [Online]. Available via: https://blog.schneider-electric.com/datacenter/2018/04/04/local-edge-lifecycle-competitive-differentiation/

[4] "An A.T.M., With a Real Teller on the Screen," [Online]. Available via: https://bucks.blogs.nytimes.com/2013/04/04/an-a-t-m-with-a-real-teller-on-the-screen/

[5] "Which Data Center Use Cases Are Best For a Value Added Reseller Business Plan," [Online]. Available via: http://www.ingrammicroadvisor.com/data-center/which-data-center-use-cases-are-best-for-a-value-added-reseller-business-plan

[6] ETSI, "Developing Software for Multi-Access Edge Computing," [Online]. Available via: http://www.etsi.org/images/files/ETSIWhitePapers/etsi_wp20_MEC_SoftwareDevelopment_FINAL.pdf

[7] "The Edge Is the Greenest, Most Intelligent Building in the World," [Online]. Available via: https://www.bloomberg.com/features/2015-the-edge-the-worlds-greenest-building/

[8] ETSI GS NFV-IFA 010, “Network Functions Virtualisation (NFV) Release 2; Management and Orchestration; Functional requirements specification,” v. 2.4.1, 02/2018.

[9] ETSI GR MEC 017, “Mobile Edge Computing (MEC); Deployment of Mobile Edge Computing in an NFV environment,” v. 1.1.1, 02/2018.

[10] ETSI GS MEC 011, “Mobile Edge Computing(MEC); Mobile Edge Platform Application Enablement,” v. 1.1.1, 07/2017.

[11] ETSI GS MEC 014, “Mobile Edge Computing (MEC); UE Identity API,” v. 1.1.1, 02/2018.

[12] IETF RFC 4186, “EAP-SIM Authentication,” 01/2006. Available via: https://tools.ietf.org/html/rfc4186

[13] Wi-Fi Alliance, “Hotspot 2.0 (Release 2) Technical Specification,” v. 1.2, 2016. Available via https://www.wi-fi.org/discover-wi-fi/passpoint

[14] ETSI GS MEC 012, “Mobile Edge Computing (MEC); Radio Network Information API,” v. 1.1.1. 07/2017.

[15] ETSI GS MEC 013, “Mobile Edge Computing (MEC); Location API,” v. 1.1.1, 07/2017.

[16] ETSI GS MEC 003, “Mobile Edge Computing; Framework and Reference Architecture,” v. 1.1.1, 03/2016.

[17] ETSI GS MEC 010-1, “Mobile Edge Computing (MEC); Mobile Edge Management; Part 1: System, host and platform management,” v. 1.1.1, 10/2017.

[18] ETSI GS MEC 010-2, “Mobile Edge Computing (MEC); Mobile Edge Management; Part 2: Application lifecycle, rules and requirements management,” v. 1.1.1, 07/2017.

[19] OpenStack, “Cloud Edge Computing: Beyond the Data Center,” Available via https://www.openstack.org/assets/edge/OpenStack-EdgeWhitepaper-v3-online.pdf

[20] ETSI GS MEC 015, “Mobile Edge Computing (MEC); Bandwidth Management API,” v. 1.1.1, 10/2017.

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org