Edge Computing Architectures – A Survey on Convergence of Solutions

Eugen Borcoci, Serban Obreja
University POLITEHNICA of Bucharest - UPB
Bucharest, Romania
Emails: eugen.borcoci@elcom.pub.ro, serban@radio.pub.ro

***Abstract* —Edge computing architectures and technologies, recently proposed, are complementary to centralized Cloud Computing, given that edge computing can offer faster response, higher context and location awareness, better mobility features, minimization of the data transfer to the centralized data centers, flexibility and so on, for a large scale of applications, including Internet of Things. Several approaches have been developed in parallel by different entities, like research groups, industry, operators, and standardization organizations. Fog computing, Multi-access (Mobile) Edge Computing, cloudlets, etc., are relevant examples, included in the large class of Edge computing. Their architectures and technologies have many essential common characteristics, but also differences in approach. A natural question is raised - if any significant convergence (which is not yet seen) will emerge in the near future. This paper is not intended to be a complete survey, but it attempts to identify some convergence directions and issues related to the Edge computing technologies.**

Keywords — Edge computing; Mobile (multi-access) Edge Computing; Fog computing; Software Defined Networking; Network Function Virtualization.

I. INTRODUCTION

The significant growth of the Internet based applications and services contributed to a steep rise in data storage and processing requirements. Cloud Computing (CC) is a powerful solution in this context, by integrating the advancements in computing and network technologies. The Cloud Computing paradigm is mainly based on the data centers which are capable of handling storage and processing of large amount of data. The data centers can be interconnected over optical networks to form data center networks (DCNs), seen by the end user as a unique powerful resource.

Internet of Things (IoT), mobile applications and services, smart cities applications, etc., recently emerging, pose novel challenging requirements the CC based solutions. Cloud computing centralization (in terms of processing and storage) in traditional data centers have inherent limitations, being non-appropriate for the above mentioned specific classes of applications. The IoT, mobile applications, vehicular networks et al., require real or near real-time response/latency, high bandwidth, location and context awareness, reduction in amount of data transferred to CC and back, more flexibility in functional distribution, resource usage optimization and others.

One solution for the above problems is Edge-oriented computing (EC) [1][2], where the main idea is to create additional CC capabilities at the network edge, close to, or even installed in the “terminal” data sources (e.g., vehicles). EC is not seen as a replacement of the CC but it is complementary to it. The CC and EC can be used independently or in cooperation. Several proposals currently exist for EC, like: *Fog Computing* (FC), *Multi-access (Mobile) Edge Computing*, *Cloudlets*, etc.

A current open research issue is to investigate if any convergence could be established in terms of concepts, architecture and implementation of such solutions.

Edge computing is of high interest not only in research communities; there are several entities involved and active in EC, in industry, research and standardization organizations. Different sets of specifications have been elaborated by independent entities.

Fog Computing (FC) is a term coined by CISCO (2012) [3][4] primarily to serve IoT needs.

Later, the *OpenFog Consortium* organization (November 2015) [5][6] has been created, having as founders: Cisco, ARM, Dell, Intel, Microsoft, Princeton University Edge Laboratory and comprises more than 60 members today. They defined an FC and *Open Reference Architecture* [5] and are creating standards to enable interoperability in IoT, 5G, artificial intelligence, tactile Internet, Virtual Reality (VR) and other complex data and network intensive applications.

The European Telecommunications Standardization Institute (ETSI) established in 2014 the *Mobile Edge Computing Industry Specification Group* [7][8]; in March 2017, the name has been re-defined as: *Multi-access Edge Computing ISG* [9] to include also non-cellular operators requirements. Cooperation started recently between Open Fog Consortium and ETSI MEC ISG to give the industry a cohesive set of standards around Fog Computing in mobile environments, while eliminating redundancy.

Another entity is *Edge Computing Consortium (*Dec. 2016) was formed, having as initial founders Huawei, Intel, ARM, and now comprising several members. The *US National Institute for Science and Technology (NIST) elaborated a* definition of FC (draft closed in September 2017) [10]. Also IETF recently started to contribute to the development of these technologies.

At Carnegie Mellon University, they developed the *Cloudlet* [11][12]; a cloudlet is middle tier of a 3-tier hierarchy: “mobile device – cloudlet – cloud”. It can be seen as a "data center in a box" whose goal is to "bring the cloud closer" to the data sources. Cloudlets are mobility-enhanced *micro data centers* located at the edge of a network and serve the mobile or smart device portion of the network. They are designed to handle resource-intensive mobile applications and take the load off both the network and the centralized data center and keep computing close to the point of origin of information.

Microsoft Research proposed in 2015 [13] the *Micro data center*, as an extension of today's hyperscale cloud data centers (e.g., Microsoft's Azure). The goal is to meet new requirements, e.g., lower latency and new demands related to devices (e.g., lower battery consumption).

The EC can generally represent any set of computing and network resources along the path between data sources and cloud data centers. However, *there is not yet a unique vision on "edge" semantics*, except the common attribute of proximity of the EC capabilities to the data sources. An important overlap exists between particular EC architectures; so, convergence is predicted in the near future.

The EC deployment can be strongly helped by novel software technologies. In the architectural management and control planes, *Software-defined networking* (SDN) [14] and *Network Function Virtualization* (NFV) [15][16] are seen as a strong support, given their features like flexibility, programmability, abstraction via virtualization, dynamicity, etc.

This paper is organized as follows. Section II summarizes the use cases and applications for which Edge computing is an attractive technology. Section III is a short overview of Edge computing architectures. Section IV identifies some common characteristics in different EC approaches, while emphasizing some points of convergence. Section VI contains conclusions and possible future work.

II. EDGE COMPUTING BASED APPLICATIONS AND USE CASES

This section gives examples of domains where EC (Fog, MEC, Cloudlets, etc.) could be effective to support the applications [1-4][7][9][12][17-19]. There is a need of open architectures based on EC, to enable interoperability in various domain of applications like in IoT, Artificial Intelligence, novel generation of networking 5G, tactile Internet, vehicular networks and Internet of Vehicles (IoV), smart cities applications, services virtual reality, and other complex data and network intensive applications. In particular, IoT applications generate high amounts of data that can be useful in many ways. Therefore, EC nodes can be used to carry out data mining and data analysis on a large volume of multi-modal and heterogeneous data from various sensor devices and other IoT devices to achieve real time and fast processing for decision making.

The Fog architecture can be hierarchically organized. This can be useful big data analysis in smart cities. Experimental results demonstrated the feasibility of the system's city-wide implementation in future smart cities scenario. Fog computing can support new services for mobile networks requiring high data rates and low latency (e.g., virtual reality).

Internet of vehicles [18] may benefit from EC (e.g., Fog, MEC) capabilities to develop a large range of applications like: safety and management-oriented (traffic safety, traffic and navigation management, remote telematics); business-oriented (infotainment, insurance car sharing, etc.). The vehicle itself can be equipped as to become a fog node, to attain optimum utilization and benefit from vehicular communications and computational resources. The mobile fog nodes can inter-communicate and provide services including infotainment, advanced driver assistance systems, autonomous driving, collision avoidance, and navigation.

Other area of applications is oriented to emergency, health care services. The latency-sensitive and security-privacy-sensitive services also can benefit from fog/edge nodes capabilities. Experimental results validated that EC supporting cyber-physical systems can improve the cost efficiency significantly based on by jointly considering base station association, task distribution.

MEC technology is a particular EC case where the MEC resources (i.e., MEC servers) are placed at the network edge (e.g., in *Radio Access Network* (RAN), i.e., Base Stations, or in aggregation points, etc.). It offers low latency, proximity, high bandwidth, and real-time insight into radio network information and location awareness. Therefore MEC can support many applications and services for multiple sectors, (enterprise, consumer, health, vehicular, etc.) [7].

In *RAN-aware Content Optimization*, an application running in the MEC server, can expose accurate cell and subscriber radio interface information (cell load, link quality) to a content optimizer, enabling dynamic content optimization, improving Quality of Experience (QoE) perceived by the users and improve network efficiency. Dynamic content optimization enhances video delivery through reduced stalling, reduced time-to-start and 'best' video quality. Among smart cities applications, a video streaming service can benefit from MEC approach. Video streams from monitoring devices can be locally processed and analyzed at the MEC server to extract meaningful data from video streams. The valuable data can be transmitted to the application server to reduce core network traffic. Other mobile application where MEC could be a subsystem is *Augmented Reality* (AR); this requires low latency and a high rate of data processing in order to provide the correct information depending on the location of the user. Collaboration is required for data collection in the uplink, computing at the edge, and data delivery in the downlink. The data are actually processed in a local MEC server rather to improve the user experience.

MEC is also useful in IoT, given that IoT devices often have low capabilities (processing, storage capacity). There is a need to aggregate various IoT messages connected through the mobile network close to the devices. Gateways (collocated with MEC servers) aggregate the messages and ensure security and low latency. To achieve an efficient service, grouping of sensors and devices is accomplished.

 ISBN: 978-1-61208-608-8

This approach also provides an analytics processing capability and a low latency response time.

III. EDGE COMPUTING ARCHITECTURES

A summary of some EC relevant architectures is presented in this section. Note that this paper investigates a possible convergence between EC architectures and technologies. In this respect, is to be noted that, currently, *there is no unique vision neither on the terminology, nor on architectural definition for EC.*

We selected and give below a summary of the *Fog computing* and *Mobile Edge Computing* and architectures, which seem to be of strong interest for industry, operators, and standardization organizations.

Fog computing (FC) is an important EC technology complementary to CC (i.e., cooperation CC/FC is envisaged). The FC distributed platform brings computation close to its data sources, to reduce the latency and cost of delivering data to a remote cloud. FC has been proposed originally to support the IoT, introduced by Cisco (Bonomi [3]]).

The *OpenFog Consortium* (2015) [5] defines FC as a system-level horizontal architecture that distributes resources and services of computing, storage, control and networking *anywhere along the continuum* from a cloud data center down to things. Therefore, FC extends the traditional CC model; implementations of the architecture can reside in multiple layers of a network's topology. The CC benefits are extended to FC (containerization, virtualization, orchestration, manageability, and efficiency) and FC can cooperate with CC. OpenFog reference architecture includes security, scalability, openness, autonomy, RAS (reliability, availability and serviceability), agility, hierarchy, and programmability. The FC focuses the processing efforts outside the cloud data center, i.e., in the fog area. Data are gathered, processed, and stored within the network, by way of an IoT gateway (GW) or a FC node (FN). Information is transmitted to this GW from various sources and it is processed in FN; then, relevant data (plus additional command - if necessary), are transmitted back, towards the devices. A FN can process data received from multiple end-points and send information exactly where it is needed.

Note that OpenFog Consortium sees the EC differently from FC, in the sense that FC works with the cloud, whereas EC is defined by the exclusion of cloud. FC is hierarchical, where EC tends to be limited to a small number of layers. In addition to computation, FC also addresses networking, storage, control and acceleration.

This vision is not agreed by all documents related to Edge- oriented computing.

On the other part, MEC, originally targets only the very edge part of the network (e.g., RAN). FC can support multiple industry verticals and application domains delivering intelligence and services to users and business. FC capability is spanning across multiple protocol layers and is not dependent on specific access systems.

OpenFog Consortium defined a *flexible deployment* hierarchical model for FC, IoT-oriented [5]. The model is mapped on a layered architecture consisting of the following layers: 1. *Sensors and actuators* (bottom layer); 2. *Monitoring and control*; 3. *Operational support*; 4. *Business support;* 5. *Enterprise systems* (highest layer).

The flexibility of the model consists in the fact that, depending on the nature and requirements of the target application class and, depending on the CC availability, cost, etc., a cooperating (for a combination CC/FC) vertical chain can be defined for layers, 2, 3, 4, e.g.:

L2.FC, L3.FC, L4.FC, or
L2.FC, L3.FC, L4.CC, or
L2.FC, L3.CC, L4.CC, or
L2.CC, L3.CC, L4.CC.

Fog computing targets quite a large range of applications. Therefore, no unique "universal" architecture exists. In terms of Fog architecture, many studies split this problem in two classes (for detailed discussion, see the survey by Mouradian, et. al. [20]):

End-User Application agnostic architectures (comprising End-User Application provisioning, Resource management, Communication functions, Cloud and federation). This class envisages some general aspects, not specific to a given application.

Application specific architectures (Smart living and Smart Cities, Connected vehicles and IoV, Healthcare and other applications).

In [20], also some criteria for evaluation of the architectures are proposed:

Heterogeneity (C1): it should be considered when deciding which application component(s) should be deployed and where;

QoS Management (C2): there are necessary architectural modules for QoS management (e.g., to assure latency) such as migration engine;

Scalability (C3): modules are needed to assure horizontal scalability (e.g., elasticity engine);

Mobility (C4): a mobility engine is necessary to ensure the continuity of a service for the end-user;

Federation (C5): cooperation is needed between different providers in order to ensure the proper coordination of the necessary interactions between application components;

Interoperability (C6): there is a need for appropriate signaling and control interfaces, and appropriate data interfaces to enable interoperability.

In [4], a general hierarchical FC-CC layered architecture is defined (application agnostic). Note that this architecture includes also the CC. Three macro-layers are defined.

Terminal layer is closest to the end user and physical environment. It consists of IoT devices (e.g., sensors, mobile phones, smart vehicles, smart cards, readers, and so on) widely geographically distributed. Note that some devices like mobile phones and smart vehicles having sufficient computing power, can be included in the next Fog layer. They are responsible for sensing the feature data of physical objects or events and transmitting these sensed data to upper layer for processing and storage.

Fog layer is located outside the CC (i.e., in the network) and it is composed of a large number of fog nodes (FN) (routers, gateways, switchers, access points, base stations, specific fog servers, etc.). The FNs can be static, or mobile on a moving carrier and are widely distributed between the end devices and cloud (e.g., cafes, shopping centers, bus terminals, streets, parks, etc.) The end devices can connect with FNs to obtain services. The FNs can compute, transmit and temporarily store the received sensed data. The Fog layer can perform real-time analysis and latency sensitive applications. The FNs are also connected with CC data center by IP core network, and responsible for interaction and cooperation with cloud to obtain more powerful computing and storage capabilities.

Cloud layer includes multiple high performance servers and storage devices, to support for extensive computation analysis and permanently storage of a huge amount of data. It provides various application services, such as smart home, smart transportation, smart factory, etc. However, different from traditional CC architecture, not all computing and storage tasks go through the cloud. According to the demand-load, the cloud core modules are efficiently managed and scheduled by some control strategies to improve utilization of the cloud resources.

In [21], a six-layer FC architecture is presented, comprising the layers described below.

Physical and virtualization layer involves different types of nodes (physical, virtual nodes and virtual sensor networks) distributed geographically. These nodes are managed and maintained according to their types and service demands. The sensors are sensing the surroundings and send the collected data to upper layers via gateways for further processing and filtering.

Monitoring layer supervises the resource utilization, availability of sensors, FNs and network elements. All tasks performed by nodes are monitored in this layer (which node, which task, at what time, what is its output, etc.).

Pre-processing layer performs data management tasks. Collected data are analyzed filtered and trimmed, in order to extract meaningful information. The pre-processed data are then stored temporarily in the temporary storage layer.

Security layer performs the encryption/decryption of data. Additionally, integrity measures may be applied to the data to protect them from tampering.

Transport layer uploads the pre-processed data to the cloud to allow the cloud to extract and create more useful services For efficient power utilization, only a portion of collected data is uploaded to the cloud.

Note that the above architecture does not include the CC itself.

NIST [10] defines the Fog computing as a horizontal, physical or virtual resource paradigm that resides between smart end-devices and traditional cloud or data centers. FC supports vertically-isolated, latency-sensitive applications by providing ubiquitous, scalable, layered, federated, and distributed computing, storage, and network connectivity. FC has as main characteristics: contextual location awareness, and low latency, geographical distribution with predominance of wireless access, large-scale sensor networks, very large number of nodes, support for mobility, real-time interactions, heterogeneity, interoperability and federation, support for real-time analytics and interplay with the cloud. NIST has defined [10] a three-layer architecture composed of *Smart end-devices* layer, *Fog* layer and *Cloud* layer. The bottom layer of the Fog is named “mist” and comprises an infrastructure close to the end-devices. The so-called “edge” is seen as a part of the Cloud layer.

Despite still different visions on EC semantics, the most agreed vision on EC and FC is that FC is actually a superset of the EC, i.e., FC would include EC.

From the industry world, the Industry 4.0 vision on Fog and IoT emerged [22]. Industrial IoT and Industry 4.0 need for extensive adoption of advanced IT features across multiple Industry verticals. IT and Operational Technology (OT) convergence is aimed. The important step to this aim is the deployment of *Cloud-like resources at the edge and within the Industrial Operational domain.*

FC merges CC features with real-time and safety OT features (efficiency, flexibility and resource management) It applies resource virtualization, real-time and no real-time computing, modern application management, data interoperability middleware, storage, analytics, advanced networking and security. Complementary technologies are *Time-Triggered Technologies* which refers to precise time distribution, time-sensitive networking and computing resource allocation (standardized as *IEEE Time Sensitive Networking TSN);* TSN is a key element of Industry 4.0 and a necessary component of FC in industrial environment. It enables the convergence of Industrial wired protocols towards a unified standard.

Recently, Bacarelli et. al. [23]) extended the FC scope, by defining *Fog of Everything (FoE)* to serve *Internet of Everything (IoE).* The FNs are usually virtualized networked data centers, which run on top of (typically, wireless) *Access Points* (APs), at the edge of the access network, resulting in a *three-tier IoE-Fog-Cloud* hierarchy. In this context, a “thing” (fixed, nomadic or mobile) is a resource-limited user device that needs resource augmentation in order to execute its workload. The work [23] proposes a hierarchical general architecture for a FoE virtualized platform, integrating the building blocks:

- *IoE layer*, where a number of (possibly, heterogeneous) things operate over multiple spatial clusters;
- *Wireless access network (fixed/mobile)*, to supports *Fog-to-Thing* (F2T) and *Thing-to-Fog* (T2F) communication through TCP/IP connections running atop, e.g., *IEEE802.11/15* single-hop links;
- A set of *inter-connected FNs*, that act as virtualized cluster headers;
- *Inter-Fog backbone* (wireline/wireless) providing inter-Fog connectivity and making feasible inter-Fog resource pooling;
- *Virtualization layer*, allowing things to augment their limited resources by exploiting the computing

capability of a corresponding virtual clone. This last runs atop a physical server of the FN that currently serves the cloned thing;

- the resulting *overlay inter-clone virtual network*, that allows P2P inter-clone communication by relying on TCP/IP E2E connections.

The corresponding protocol stack [23] comprises four layers:

IoE layer provides services like: *(a)* T2F access through a reservation-based collision free access protocol for the things served by a same FN; *(b)* F2T broadcast services.

Fog layer performs: *(a)* energy-efficient management of the networking and computing physical resources equipping each FN, and *(b)* energy-efficient management of the inter-Fog traffic conveyed by the wireless backbone.

Overlay layer supports the overlay inter-clone P2P network by: (a) inter-Fog Clone migration; it can be supported by the implementation of the so-called *Follow-Me-Cloud* framework (e.g., Taleb et al., [18]), to solve "live" inter-Fog clone migration, in response to the thing mobility; (b) dynamic management of the required migration bandwidth, to minimize the energy consumed by clone migrations.

Cloud layer orchestrates the overall Cloud-Fog-IoE platform on the basis of the specific features and Quality of Service (QoS) requirements of the running applications. The solutions must be tailored on the expected attributes of the supported applications.

The MEC architecture is another important EC approach. It has been promoted mainly by ETSI [7-9] and offers low latency/response time, high bandwidth, location and context awareness, reduction in amount of data transferred from/to a terminal device to a centralized cloud data center, etc. The ETSI MEC Industry Specification Group (2014) provided first specifications. In 2017, the MEC name (and scope) has been extended to *Multi-access Edge Computing* [9], to include non-cellular and fixed access cases. MEC supports multi-services and multi-tenancy and is usually developed in the operators' networks. However, authorized external third-parties may also make use of the MEC storage and processing capabilities.

The MEC resources are placed at the radio network edge (e.g., in *Radio Access Network* – RAN, i.e., Base Stations, or in aggregation points, etc.). The key element is *MEC application server*, integrated in RAN and providing computing resources, storage capacity, connectivity, and access to user traffic and radio and network information.

The MEC reference architecture is presented in Figure 1 (details, in [8]). The mobile edge host level is the main MEC sub-system, composed of: the *Mobile Edge Host* (MEH) and its *management*. The MEH includes a virtualization infrastructure (based on *Network Function Virtualisation Infrastructure* –NFVI- coming from ETSI NFV framework [15][16]) and the *Mobile Edge Platform* (MEP), supporting the execution of mobile edge applications. The MEC server can be installed in various places at the network edge: at the 4G/LTE macro base station (eNB); at the multi-technology (3G/LTE) cell aggregation site; at the Radio Network Controller (RNC) site, for 3G.

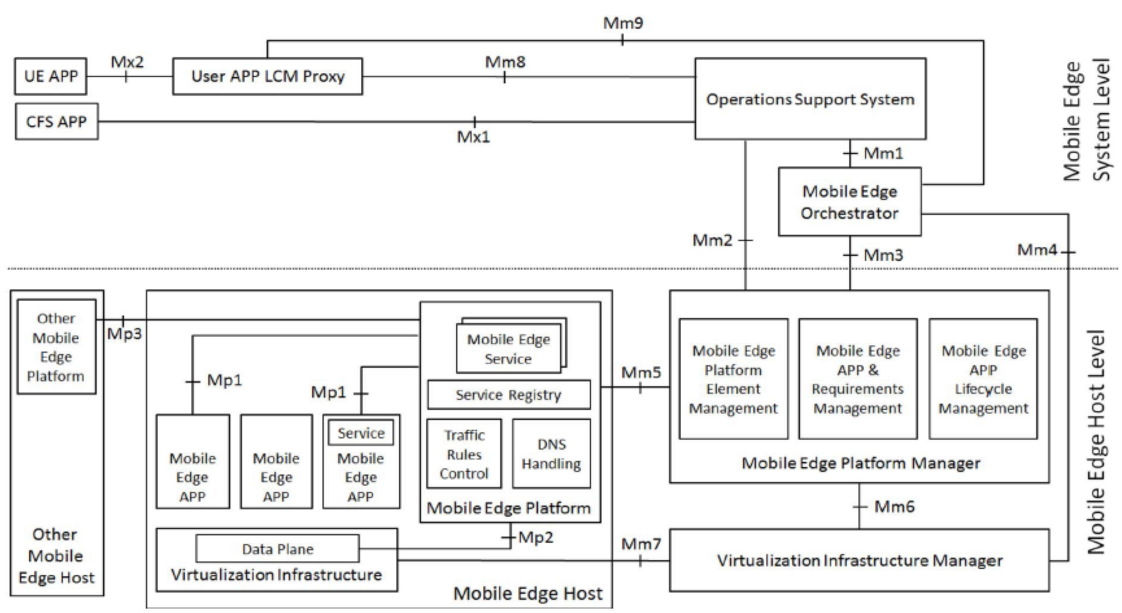

Figure 1. MEC reference architecture (ETSI) [8]

 ISBN: 978-1-61208-608-8

IV. EDGE COMPUTING POSSIBLE CONVERGENCE

The previous section outlined the general characteristics of the EC and summarized two main approaches FC and MEC. This section analyzes some common features and also differences in approaches, to evaluate chances of convergence.

MEC/FC/Cloudlets have quite a lot of common characteristics like: low latency; support for real time interactions, location awareness and mobility and large number of server nodes; geographical distribution proximity to the end devices (single network hop or few hops); service location at the edge of the local network; various working environment outdoor (streets, base stations, etc.) or indoor (houses, cafes, etc.); wireless communication access: WLAN, WiFi, 3G, 4G, ZigBee, etc., or wired communication (part of the IP networks); weak dependence on the quality of core network; low bandwidth costs energy consumption. However, the nodes in FC, MEC, Cloudlets have weak computation and storage capabilities, which raises a need for them to cooperate with CC. All three approaches can benefit from technologies like SDN and NFV in different architectures.

Considering MEC and FC, there are differences between them from several points of view (see [1][6]), as summarised in Table I.

TABLE I. MEC VERSUS FOG DIFFERENCES

Criterion	MEC	Fog computing
Placement of node devices	Servers running in Base stations Network Controller/Macro Base Station	Anywhere -between end devices and cloud data center: Routers, Switches, Access Points, Gateways
Compute Distribution and Load Balancing	Employ a strategy of placing servers, apps or small clouds at the edge	Broader architecture and tools for distributing, orchestrating, managing and securing resources and services across networks.
Software Architecture	Mobile Orchestrator based (strongly specified)	Fog abstraction layer based (only partially specified)
Standardization/ specifications	ETSI/	/OpenFog Consortium
Context awareness	High	Medium
Proximity	One hop	One or multiple hops
Access Mechanisms	Mobile networks: 3G/4G/5G	Wi-Fi, Mobile networks, etc.
Virtualization and management mechanisms	Strongly specified by ETSI (NFV framework)	Larger view of virtualization. In progress at OpenFog Consortium
Hierarchical structure of the overall system	Possible	Yes: multiple levels of cooperating nodes, supporting distributed applications.
Horizontal scalability	Medium	High
Internode Communication	Possible	Supported
Communication with CC	Possible	Required
Modular architecture with multiple access modes	Edge deployments are typically based on gateways with fixed functionality. However they can be made more flexible and dynamic by using NFV	Highly modular HW&SW architecture; every FN is equipped with exactly the resources its applications need; itcan be dynamically configured
Topology of server nodes	Less flexible (limited by RAN spread)	Very flexible

The above table shows several differences between MEC and FC approaches. Note that Table I does not suppose some application specific architectures, but general ones. From this, it is apparent that there is not yet defined a common EC architecture. The MEC/FC/Cloudlets paradigm can offer more or less appropriate support for a large variety applications and use-case scenarios and heterogeneous end devices. On the other side different use cases and applications might have their own set of requirements and trade-offs which can determine which solution is the appropriate choice.

Actually, for a given set of use cases, the selection of an appropriate EC approach is a multi-criteria problem. Among the parameters/criteria for selection those presented in Table I could be considered, if appropriate weights are assigned to them.

However, recently, a strong effort for cooperation started, between different organizations, towards a convergence of vision in the domain of edge computing (including MEC, Fog, Cloudlets, etc.).

Open Edge Computing (OEC) [24][25] is a novel general approach of EC, towards convergence, consisting in small data centers at the network edge, offering computing and storage resources next to the user. Carnegie Mellon University (CMU) performed an early work on Cloudlets at the edge. Given the interests in EC, in 2015 a few parties joined research efforts under the open source banner of Open Edge Computing (OEC). Currently, OEC ecosystem includes CMU, Intel, Huawei, Vodafone, and so on.

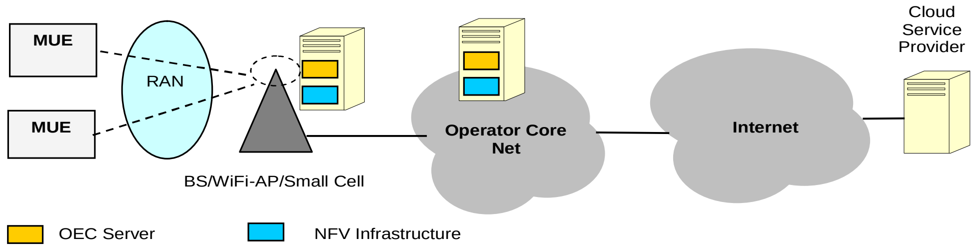

Figure 2. OEC general architecture.
MUE- Mobile User Equipment; BS- Base Station; NFV – Network Function Virtualization

The main OEC goals are: to promote Cloudlets as enabling technology; to drive the necessary technology for various use cases (low latency and computation at the edge) (e.g., extensions to OpenStack, KVM, QEMU); to prototype applications that leverage EC pushing the boundaries and demonstrating benefits; to drive the eco system development for OEC and use current IT solutions.

OEC is engaging with target service industries/sectors through demonstrators and joint projects; with developer communities, seeking feedback and driving EC acceptance. OEC is synchronizing its work with other efforts including ETSI ISG MEC and OPNFV.

The OEC servers can be located close/associated to Base Stations, Access Points, Small Cells, or even in the Operator Core Network (Figure 2). Edge Computing will utilize the Network Function Virtualization (NFV) infrastructure wherever possible. This will reduce significantly the deployment cost of EC.

V. CONCLUSION AND FUTURE WORK

This paper presented a preliminary comparative view of some Edge Computing approaches (Fog computing, Mobile Edge Computing, Cloudlets) in order to identify their common and different characteristics and possible chances to have in the near future an EC unified architecture. For the time being, no strong convergence exists, given the large area of use cases and applications targeted to be supported and the pragmatic attempts to tailor the specific architecture to the desired class of applications.

However, recently, a strong effort for cooperation started, between different organizations, towards a convergence of vision in the domain of edge computing (including MEC, Fog, Cloudlets, etc.).

Future work should be done, to investigate more deeply the sets of architectural layers and mechanisms to identify where a common approach can be applied, in order to reduce the development effort and reuse some already developed functional modules.

REFERENCES

[1] K. Dolui and S.K. Datta, "Comparison of Edge Computing Implementations: Fog Computing, Cloudlet and Mobile EdgeComputing", Global Internet of Things Summit (GIoTS), 2017, https://ieeexplore.ieee.org/document/8016213/, [Retrieved: May 2018].

[2] Y. Ai, M. Peng and K. Zhang, "Edge computing technologies for Internet of Things: a primer", Digital Communications and Networks 4 (2018), pp. 77–86, www.keaipublishing.com/en/journals/digital-communications-and-networks/ , [Retrieved: May 2018]

[3] F .Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog Computing and Its Role in the Internet of Things", August 2012, http://conferences.sigcomm.org/sigcomm/2012/paper/mcc/p13.pdf, [Retrieved: January, 2017].

[4] P. Hu, S. Dhelim, H. Ning and T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues", Journal of Network and Computer Applications 98 (2017) pp. 27–42, www.elsevier.com/locate/jnca, [Retrieved: May 2018].

[5] OpenFog Consortium (2015), OpenFog Reference Architecture for Fog Computing, http://www.openfogconsortium.org/resources/#definition-of-fog-computing, [Retrieved: April 2018].

[6] OpenFog Consortium, "10 ways fog computing extends the edge", August 21, 2017, https://www.rtinsights.com/10-ways-fog-computing-extends-the-edge/, [Retrieved: May, 2017]

[7] ETSI GS MEC-IEG 004 V1.1.1 (2015-11), "Mobile-Edge Computing (MEC); Service Scenarios", http://www.etsi.org/deliver/etsi_gs/MEC-IEG/001_099/004/01.01.01_60/gs_MEC-IEG004v010101p.pdf , [Retrieved: January 2018].

[8] ETSI GS MEC 003 V1.1.1 (2016-03), "Mobile edge computing (MEC); Framework and reference architecture," http://www.etsi.org/deliver/etsi_gs/MEC/001_099/003/01.01.01_60/gs_MEC003v010101p.pdf [Retrieved: January, 2018].

[9] T.Taleb, K.Samdanis, B. Mada, H. Flinck, S. Dutta, and D. Sabella, "On Multi-Access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration", IEEE Comm. Surveys &Tutorials, vol. 19, no.3, pp. 1657-1681, 2017.

[10] M. Iorga et. al., NIST Special Publication 800-191 (Draft) 1, "The NIST Definition of Fog Computing", 2017,

https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.500-325.pdf, [Retrieved: January , 2018].

[11] "Cloudlets: all about cloudlet-enabled mobile computing" http://elijah.cs.cmu.edu/, [Retrieved: January , 2018].

[12] Satyanarayan, et. al., "Cloudlets: at the Leading Edge of Mobile-Cloud Convergence", 2014, Int'l Conf.on Mobile Computing, Applications and Services (MobiCASE)http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=7026272, [Retrieved: January , 2018].

[13] V. Bahl, " Microsoft, interview about micro datacentres", Sept 2015, http://www.networkworld.com/article/2979570/cloud-computing/microsoft-researcher-why-micro-datacentersreally-matter-to-mobiles-future.html, [Retrieved: February , 2018].

[14] B. N. Astuto, M. Mendonca, X. N. Nguyen, K. Obraczka, and T. Turletti, "A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks", Communications Surveys and Tutorials, IEEE Communications Society, (IEEE), 16 (3), pp. 1617 – 1634, 2014.

[15] ETSI GS NFV 002: "Network Functions Virtualisation (NFV); Architectural Framework", http://www.etsi.org/deliver/etsi_gs/NFV/001_099/002/01.01.01_60/gs_NFV002v010101p.pdf, [Retrieved: January , 2018].

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

[17] W. Yu, et al., "A Survey on the Edge Computing for the Internet of Things", IEEE Access, 2017, DOI 10.1109/ACCESS.2017.2778504

[18] O. Kaiwartya, A.H. Abdullah, Y. Cao, A. Altameem, and M. Prasad, "Internet of Vehicles: Motivation, Layered Architecture, Network Model, Challenges, and Future Aspects" IEEE Access, Special Section on Future Networks, Architectures, Protocols and Applications, vol. 4, pp. 5536-5372, September 2016.

[19] K. Zhang, Y. Mao, S. Leng, Y. He, and Y. Zhang, "Mobile-Edge Computing for Vehicular Networks- A Promising Network Paradigm with Predictive Off-Loading", IEEE vehicular technology magazine, pp. 36-44, June 2017.

[20] C. Mouradian, et. al., "A Comprehensive Survey on Fog Computing: State-of-the-art and Research Challenges", IEEE Communications Surveys & Tutorials, 2017, https://arxiv.org/ftp/arxiv/papers/1710/1710.11001.pdf, [Retrieved: April 2018].

[21] H.F. Atlam, R. J. Walters and G.B. Wills, "Fog Computing and the Internet of Things: A Review", Big Data Cogn. Comput. 2018, pp.2- 18; doi:10.3390/bdcc2020010, https://www.mdpi.com/2504-2289/2/2/10/pdf, [Retrieved: April 2018]

[22] "Fog Computing: Keystone of Industrial IoT and Industry 4.0", https://www.fogworldcongress.com/conference/agenda/fog-computing-keystone-industry-40, [Retrieved: April 2018]

[23] E.Bacarelli, P.G.Vinueza Naranjo, M.Scarpinti, M.Shojafar, and J.H.Abawaji,"Fog of Everything: Energy-Efficient Networked Computing Architectures, Research Challenges, and a Case Study", IEEE Access, vol. 5, 2017, pp. 9882-9910.

[24] http://openedgecomputing.org, [Retrieved: April 2018]

[25] R. Schuster and P. Ramachandran, Open Edge Computing – From Vision to Reality - 2016 OPNFV Design Summit, Berlin, Germany, http://docplayer.net/32036666-Open-edge-computing-from-vision-to-reality.html, [Retrieved: April 2018].