Mobile Edge Computing with Network Resource Slicing for IoT

Conference Paper · February 2018

DOI: 10.1109/WF-IoT.2018.8355232

5 authors, including:

Syed Husain
Syracuse University
26 PUBLICATIONS **115** CITATIONS
SEE PROFILE

Andreas Kunz
Lenovo
35 PUBLICATIONS **512** CITATIONS
SEE PROFILE

Athul Prasad
Nokia
60 PUBLICATIONS **474** CITATIONS
SEE PROFILE

K. Samdanis
Nokia Bell Labs, Munich
74 PUBLICATIONS **1,492** CITATIONS
SEE PROFILE

Mobile Edge Computing with Network Resource Slicing for Internet-of-Things

Syed Husain
NTT DOCOMO
Tokyo, Japan
shusain2016@gmail.com

Andreas Kunz
Lenovo
Oberursel, Germany
akunz@lenovo.com

Athul Prasad
Nokia Bell Labs
Espoo, Finland
athul.prasad@nokia-bell-labs.com

Konstantinos Samdanis
Huawei Technologies
Munich, Germany
konstantinos.samdanis@huawei.com

JaeSeung Song*
Sejong University
Seoul, Republic of Korea
jssong@sejong.ac.kr

Abstract— Network slicing is an end-to-end concept that encompasses network functions, radio accesses, and clouds for enabling customized information-centric Internet-of-Things (IoT) services. The key idea is to virtualize all the resources from radio accesses to IoT service layers, so that IoT service providers can automate resource provisioning and management for users. This paper introduces the recent standards effort on network slicing for IoT in various standards bodies such as 3GPP, NGMN, IETF, ETSI and oneM2M. In particular, standards activities on ETSI Multi-Access Edge Computing (MEC), NGMN and 3GPP Network Slicing and Virtualization, IETF slicing on transport networks and oneM2M IoT service layer resource slicing are introduced. Finally, this paper proposes a novel Edge Computing architecture customizing required network resources at the edge cloud, as close to users as possible, to minimize network signaling overhead in providing optimal IoT services.

Index Terms—Edge Cloud, Network Slicing, IoT, MEC, oneM2M, Consistency, and 3GPP.

I. Introduction

The demand on the mobile network to deliver services to the end-users with low latency is ever increasing due to the fact that end-users do not see any difference in delivery of services whether in a fixed or mobile environment. In addition, the proliferation of machine type communication (MTC) services, such as smart home, smart city, smart cars, etc., which are all part of Internet-of-Things (IoT), require the mobile network to be agile, easily configurable, and highly efficient [1, 2]. These requirements are being addressed as part of the overall 5G standardization effort in 3GPP radio and network technical specification working groups. In future mobile networks (5G and beyond), Mobile Edge Computing (MEC) [3] will play a key role in addressing these requirements.

MEC provides operators the ability to host services at the edge of the network in close proximity to the end-user to fulfil low latency requirements. In addition, to allow services to be easily configurable based on customer needs, it is necessary that network resources are partitioned into manageable components that can be easily combined to deliver a particular service. This concept has been defined in ITU-T specifications [4], as logically isolated network partition (LINP), and is called network slicing in 3GPP. Each network slice is an aggregation of multiple physical and virtual network resources on a dynamic basis for delivery of a service, on demand or on a continuing basis, as deemed necessary by the operator. Therefore, network slicing and virtualization are key components of MEC.

In this paper, we look at the use of MEC for delivery of IoT services. The concept and standards of MEC, which are now being defined in ETSI are introduced. Since network slicing and virtualization are key components of MEC, we then describe the status of this work in 3GPP together with resource slicing technologies at the IoT service layer [5], which can also be hosted at the edge cloud. In addition, the architectural details of Mobile Edge IoT Cloud are described providing insight on how Edge Cloud is formed in delivery of low latency services.

The rest of the paper is organized as follows. Section II presents the introduction to the concept of Multi-Access Edge Computing and the recent status of MEC related standards activities. Network slicing technologies, which are used at the core network and being standardized in 3GPP, are explained in Section III followed by introducing transport network slicing in Section IV. Then Section V discusses about a high-level concept of slicing IoT service layer platform. Section VI introduces a novel architecture for integrating various network slicing technologies from lower layer (e.g., access network) to higher layer (e.g., service and application) as a future direction. Finally, we have concluded the paper in Section VII.

II. Multi-Access Edge Computing for IoT

Multi-access Edge Computing (MEC) [6] offers open cloud computing and IT capabilities at the network edge considering heterogeneous radio and fixed network environments. The use of MEC facilitates computational and storage resources in close proximity to end usage, assuring low latency and high data rates, providing also the potential for innovation at the edge offering a customized service experience. At the moment, a MEC

platform can also host a local IoT gateway functionality capable of performing data aggregation or data trimming as well as big data analytics for event reporting (e.g., temperature measurements) or alarm notifications.

In other words, MEC can assure scalability by processing or filtering data or analyzing and converting raw data into meta-data at the edge, avoiding in this way unnecessary signaling and traffic that would otherwise have overloaded the backhaul and core networks. MEC can also enable a number of location specific services including smart city services, such as including video analytics, transport, location services, intelligent public spaces, safety and emergency. These services have limited geographical scale and stringent latency requirements such as retail targeted video advertising. The use of MEC can complement slicing by enabling customization and the means of meeting latency Service Level Agreement (SLA) for particular applications, such as Tactile Internet.

Technical standards specifications for MEC are being developed by a special expert group, called MEC-ISG (Industry Specification Group), at the European Telecommunications Standards Institute (ETSI) [7]. The main purpose of MEC standards is to provide a new ecosystem for Multi-Access Edge Computing. Many operators are looking for opportunities to open their Radio Access Network (RAN) edge to third parties providing innovative IoT services. MEC-ISG intends to glue technologies from the telecommunications and cloud worlds in order to take the advantages of using cloud computing features in the RAN. For this purpose, the standards group defines essential elements enabling various IoT applications to be located in a multi-vendor mobile edge. As IoT applications and services are hosted on top of the network layer, users can receive various IoT services that require elasticity, flexibility, dynamicity and customization along with different levels of service. After delivering its first set of specifications successfully, the group is now standardizing more advanced MEC features such as supporting non-3GPP access technologies, enabling containerization of network resources, integration of MEC and NFV environment. In addition, the group plans to support interoperability via developing testing methodologies. This phase 2 activities will increase the level of virtualizations so that more advanced IoT applications can benefit from being in close proximity to the customers with full blown MEC features.

III. Network Slicing and Virtualization

Network slicing can be seen as one of the new key features of the 3GPP 5G architecture [8] that refers to logical self-contained networks on top of a common physical infrastructure incorporating physical and/or logical network and cloud resources into a programmable environment open for vertical and application providers. The concept and architecture of network slicing is introduced by NGMN [9] considering a 3-layer architecture including a service layer, network slice instance and the resource layer.

The notion of slicing was already introduced in 3GPP Release 13 for the Evolved Packet System (EPS) with the Dedicated Core Network (DECOR) feature, which was

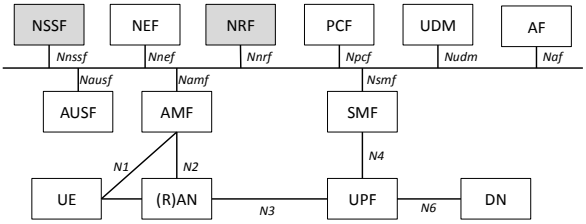

Fig. 1. 3GPP 5G Non-Roaming Architecture

intended for Machine Type Communication (MTC) devices. At time of attach to the network of such an MTC UE, the MME (Mobility Management Entity) would detect the MTC UE and would reroute it to a dedicated MME serving only those MTC devices. This dedicated MME could further select a dedicated Serving Gateway (SGW) and PDN Gateway (PGW) for the user plane (UP) traffic. This concept was improved in Release 14 with Enhanced DECOR to avoid the rerouting via RAN, thus the UE indicates directly in the request the Dedicated Core Network (DCN) ID so that the RAN can select the right MME from the beginning. Those features require static pre-configuration in the UE and do not offer much flexibility for the operator to apply slicing in a more dynamic manner to the network. This is taken care of with the completely new system architecture of the 5G system of the 3GPP Release 15, as shown in Fig. 1.

One main difference to the traditional system architectures of previous generations is the service based approach where all of the core network functions can communicate with each other. This enables also the virtualization of the network and facilitates the slicing with dedicated network functions. The functionality in MME of the Evolved Packet Core (EPC) got split into the Access and Mobility Management Function (AMF) for terminating the 5G-NAS (Non-Access Stratum) and the Session Management Function (SMF) for all user plane related signaling with the User Plane Function (UPF). Further new functions are the Authentication Server Function (AUSF), Network Exposure Function (NEF), NF Repository Function (NRF), Network Slice Selection Function (NSSF), Policy Control function (PCF), Unified Data Management (UDM and the Application Function (AF). In the following the use of NSSF and NRF is described later in their relevance to network slicing.

Network slices can have different features or may offer different network optimizations. Network operators are free in their number of supported slices and how to assign subscribers to one or more slices. The maximum limit of simultaneously served slices for one particular UE is 8. When the UE registers to the network for the first time, the AMF will select the Network Slice instances for the UE. The so-called S-NSSAI (Single Network Slice Selection Assistance information) is the identifier of a Network Slice and consists of two parameters:

- Slice/Service type (SST): describes the expected Network Slice behaviour in terms of features and services;

- Slice Differentiator (SD). optional information to differentiate amongst multiple Network Slices of the same Slice/Service type.

The UE sends the collection of S-NSSAIs to the network of which the UE would like to get service of those network slices. This collection of max 8 S-NSSAIs is called Network Slice Selection Assistance Information (NSSAI).

At the moment, there are three standardized SST values for establishing global interoperability for slicing so that PLMNs can support roaming users more efficiently. Those three services/features are eMBB (enhanced Mobile Broadband), URLLC (ultra- reliable low latency communications) and MIoT (massive IoT). The MIoT SST provides network optimizations for the support of a large number and high density of IoT devices.

The HPLMN can configure the UE with the Configured NSSAI per PLMN and the UE shall only use the S-NSSAIs of the Configured NSSAI of this PLMN. After registration in a PLMN, the UE may receive from the AMF an Allowed NASSI for this PLMN with one or more S-NSSAIs, which takes precedence over the Configured NSSAI in the UE. If the Allowed NSSAI consists of S-NSSAIs with non-standardized values, which are not part of the Configured NSSAI in the UE then the Allowed NSSAI contains mapping information for the UE.

When the AMF receives and initial registration request from the UE, it queries the UDM for the subscribed S-NSSAI and verifies whether the requested S-NSSAIs from the UE are permitted based on the subscribed S-NSSAIs.

If the AMF cannot serve all the requested S-NSSAIs, then it queries the NSSF, which selects the set of network slice instances for the UE and selects the target AMF Set to serve the UE. The AMF then queries the NRF with the target AMF Set and receives from the NRF a list of AMFs and may reroute the registration request from the UE to a target serving AMF. The detailed procedures are documented in [10].

As compared to core network slicing where computing infrastructure could be sliced depending on use cases and service-level agreements, radio access network slicing is more challenging due to the usage of physical infrastructure and related access / spectral resources. But, similar to the core network, network slicing is considered to be a key enabler for supporting the quality of service requirements for 5G use cases and services, while enforcing end-to-end service-level agreements. One key application for RAN slicing, especially in the context of IoT is the delivery of common content to a massive number of devices. Such content could include software updates or other signaling information that would be applicable to all the devices within the coverage area of the mobile network. The radio protocol architecture for RAN slicing, adapted from the figure presented in [11], taking into account the currently defined 5G/NR overall architecture presented in 3GPP TS 38.300 [12], is as shown in Fig. 2. Here the key consideration is the usage of service data adaptation protocol (SDAP) which handles the QoS flow mapping, to take into account the dynamic usage of resources.

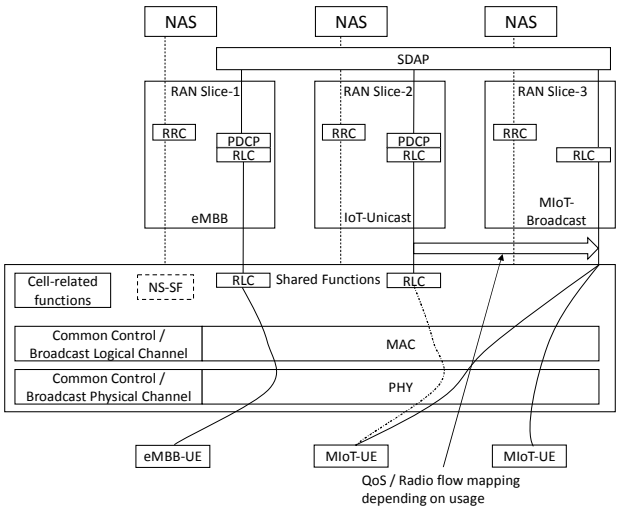

Fig. 2. Radio protocol architecture for RAN slicing [11, 12], with dynamic QoS/Radio flow mapping.

It has been shown that dynamic RAN slicing with shared spectral resources enables the most efficient of the overall available physical resources, due to the dynamic reallocation of available resources, depending on real-time radio network conditions. Yet another key application of RAN slicing, in the context of IoT, would be the mapping of content meant to be delivered to multiple devices to a broadcast slice. Since broadcast data in mobile network involves limited control functions (for e.g., without the usage of PDCP layer), enabling such slicing would require dynamic mapping of appropriate QoS/radio flows to broadcast physical resources for e.g., using the network slice – selection function (NS-SF). Here the assumption is that the SDAP could be made aware of such QoS flow to radio bearer mapping requirements, depending on the dynamic radio resource utilization within a subset of base stations / cells within the radio network. Thus, using dynamic radio network slicing, the common content flows for MIoT-UEs could be mapped from a unicast to broadcast IoT slice, if a large of MIoT devices are consuming such content. Currently, the support for multicast/broadcast content delivery is also not supported in 5G, and here it is assumed that such mapping functions could be enabled with the specification of such features. It is also assumed that the 5G end-to-end architecture shown in Fig. 2 would be adapted to support multicast/broadcast functions.

IV. Transport Network Slicing

Slicing in the transport network involves isolation and customization in the backhaul and front haul networks that provide connectivity for separate slices among the RAN and core networks. Currently, the legacy overlay networks, VLANs and VPNs [13] can be re-used for this purpose, however their capability is limited since they only provide software-based separation in the data plane.

In providing an enhanced solution, VPN+ [14] introduces separation the in the control plane allowing an independent control plane per slice. In addition, VPN+ relies on Segment

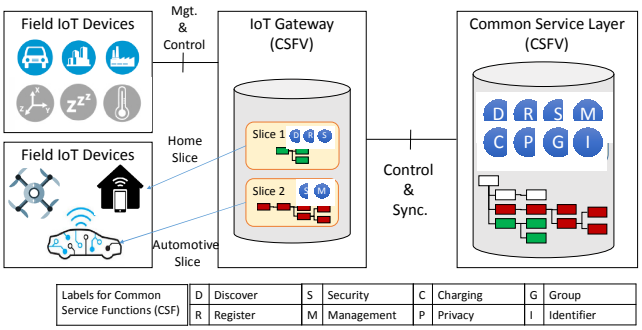

Fig. 3. Slicing for IoT Common Service Functions (CSF) using VMs

Routing to provide Flexible Service Chaining, by explicitly specifying in the routing header a partial path that contains, virtual network function and/or specific cloud platforms that should be involved in delivering a service. Segment Routing can also enable the selection of a particular physical or logical link even out of a bundle, and choose a specific queue for achieving a pre-determined service performance within a routing or switch. These characteristics are significant for IoT services since they can assist the creation of a service and the fulfillment of delay requirements.

Alternatively, DetNet [15] can be adopted as an option for the data plane for particular IoT slice in order to ensure a lower delay bound and ultra-reliability via the use of disjoint links, which are significant for critical communications. The use of deep programmability can be realized through network slicing offering Information Centric Networking untying data from a host, storage, application, and transport protocol, enabling replication within the network supporting a large-scale of heterogeneous IoT devices, enhancing network robustness, and scalability.

V. IoT Service Layer Slicing

There are many ongoing research works on network slicing for access and core networks. However, network functions for IoT service layers can also be virtualized. Several global standards (e.g., oneM2M) and proprietary (e.g., IBM Watson) IoT service layer platforms have been integrating Cloud and IoT to provide scalable IoT services. Most of these cloud-based IoT service are designed and implemented based on a centralized-cloud concept [2, 5]. This means that all common IoT service functions such as device management, group data handling, and discovery are deployed to a centralized cloud service. In order to receive any of these common services, a request or messages have to be delivered from a user device to the cloud server where all these common service functions are located in. These kinds of cloud-based IoT services are typically fine for traditional IoT services such as smart home. However, there exist much more different IoT use cases, which require different types of features and service functions. For example, industrial and mission critical IoT services require very fast response time (e.g., less than a few ms) and higher reliability than other normal services. If a common IoT service layer is located at the cloud where is far away from users, even accesses and core networks are virtualized and moved to a place close to users, it is not possible to deliver a service within a few ms as a request message has to be delivered to the cloud server. In order to cope with this, a concept of slicing IoT common service function is proposed in this paper and is explained in Fig. 3 in three steps.

In the current IoT service layer platforms, common IoT service functions are deployed to three types of IoT entities, i.e., field devices (e.g., sensor and actuator), gateway and cloud server. Similar to other network slicing technologies, to implement IoT service platform slices, all these common service functions have to be virtualized together with service dependent data. Basically, the main idea of Common Service Function Virtualization (CSFV) is to softwarize all Common Service Function (CSF) defined in IoT service layer platforms and place these v-CSF onto virtual machines deployed on a virtualized commercial server that can be installed on IoT gateways. Not just v-CSFs but also a set of resources associated with an IoT service using slicing technologies is retrieved from the server-side cloud and placed at IoT Gateways. This way, IoT Gateway works as edge cloud while IoT Infrastructure Server works as core service cloud. Interface between VMs located in edge gateway and core server clouds is defined to synchronize data and control v-CSF instances. Then, slices are created for each IoT service (i.e., unmanned vehicle slice, smart city massive IoT slice, mission-critical IoT slice, and so on).

VI. Architecture for Mobile Edge IoT Cloud

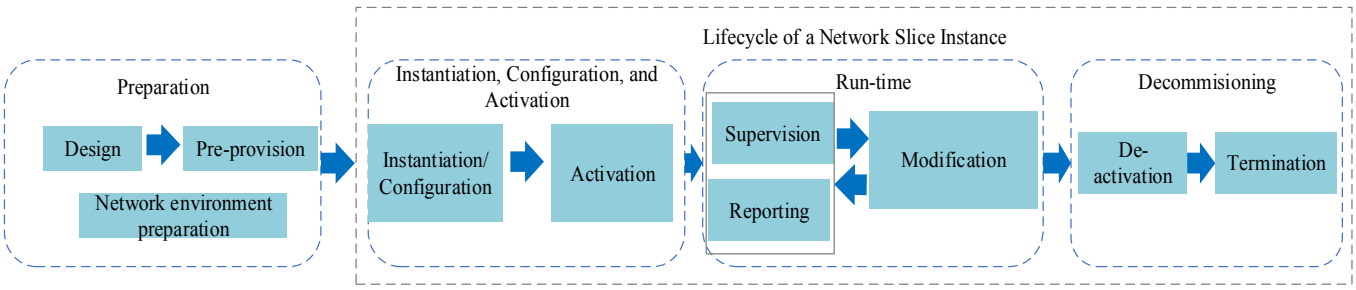

Fig.4. 3GPP Network Slice Life-cycle Management [16]

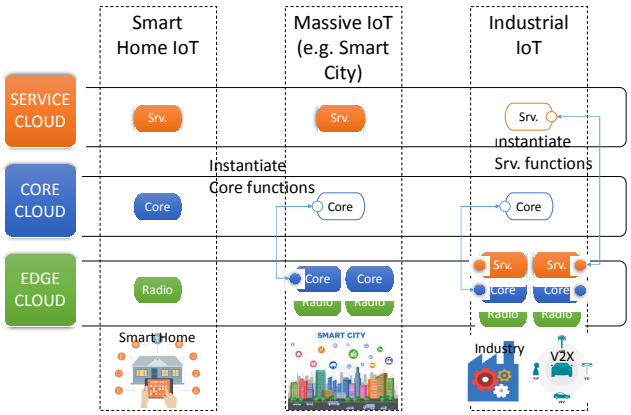

Fig. 5. Architecture for Mobile Edge IoT Cloud

In this section, we introduce a high-level architecture of potential Mobile Edge IoT Cloud integrating various network slicing technologies and its coordination & orchestration function.

A. Slicing Management & Orchestration

In 3GPP study on slice management and orchestration [16], the lifecycle management of a network slice is decoupled from the corresponding service instance that uses it, allowing scalability in service provision. In particular, the service management can be performed in e.g. the IoT provider domain, while the slice management in the network operator's domain. The inter-relation between the service and slice management relies on a business interface that may allow different levels of control such as monitoring, selection from catalogue or full programmability.

The life-cycle management of a network slice is illustrated in Fig.4, including the preparation phase and the lifecycle of network slice instance. The preparation phase concentrates on all activities related with the design of the network slice and creation of the appropriate network environment in order to install a network slice instance. The lifecycle management phase of a network slice includes the following phases: (i) instantiation, configuration and activation, (iii) run-time and (iii) decommissioning. The instantiation, configuration and activation includes the network slice resource configuration in where the necessary resources both shared and dedicated considering network function are configured but not yet used, while in the activation sub-phase, the slice becomes actively handling network traffic and user context. The run-time phase enables data traffic for a particular slice providing the essential guidance and reporting, i.e., feedback loop, to assure performance considering resource scaling and reconfiguration based-on evolving demands. Finally, the decommissioning phase includes the deactivation and termination of a network slice instance reclaiming the allocated resources.

B. Slicing Management & Orchestration

Based on the slicing management & orchestration feature introduced in the previous section, we derive a novel architecture for MEC integrating and enabling various network slicing technologies so that increase the level of automation, flexibility and programmability for IoT applications.

Fig. 5 shows how IoT applications in different service domain can be virtualized and instantiated in each slice. For example, each slice can be instantiated as follows:

- Normal IoT applications (e.g., smart IoT home): Existing IoT cloud services can be provided to users using existing network configurations.
- Massive IoT applications (e.g., IoT-enabled smart city): As there exist many different IoT services (e.g., smart building and smart grid) within a smart city, these services are better to be served with their specific needs. Network access and 5G IoT Core can be virtualized at Edge Cloud while the services are provided at Service Cloud.
- Industrial IoT applications (e.g. Mission critical IoT): Network access, 5G IoT Core and even associated IoT server platforms are all moved to Edge Cloud to minimize transmission delay and provide optimal services to users.

As explained, dedicated slices are created for IoT services based on different service requirements. Depending on needs virtualized network functions are being placed in different places in each slice (i.e., Edge, Core and Service Cloud). For 5G IoT Core, some network functions (e.g., charging, multicast) are necessary in one slice, but unnecessary in other slices. Similar to this, for IoT service platform, some common service functions (e.g., group management, security) are essential in one slice at Edge Cloud because of time constraints, while others are enough to be provided at Service Cloud. IoT service providers can customize network slices the way they want, probably in the most cost-effective way.

Drones could be one of use cases for enabling Mobile Edge IoT Cloud. Drones can fly dangerous areas where human being cannot explore easily. These unmanned IoT devices (i.e., Drones) should communicate and interact with their control center where is located at the centralized server. Mobile Edge IoT Cloud enables drones to analyze data and react based on the result in real-time. For example, a drone identifies a forest fire accident, the device can provide valuable information to its Mobile Edge IoT Cloud.

VII. Conclusions

In this paper, we described how MEC, network resource slicing, and cloud computing technologies residing in various layers (network, radio access, and service) can be combined for enabling customized IoT services. In addition, we described the standardization efforts underway of these technologies in various global standards bodies that will facilitate multi-vendor deployments of IoT services. The high-level architecture of Mobile Edge IoT Cloud is shown demonstrating how various network slicing technologies can be integrated and delivered in a coordinated way. Research and standardization efforts of these technologies will continue in the next phase of 5G in order to support new use cases and business models. For future works we intend to investigate how to synchronize information distributed at various layers and validate the consistency of a

set of distributed data. In addition we plan to design a real-time reliability testing framework for the Mobile Edge IoT Cloud systems. Furthermore, the proposed architecture can be compared with other open standard architectures such as the OpenFog Consortium.

Acknowledgment

Prof. Song was supported by Institute for Information & communications Technology Promotion (IITP) grant (No.B0184-15-1003) and the National Research Foundation of Kore (NRF) grant (No. NRF-2017R1D1A1B03036285).

References

[1] H. Shariatmadari *et al*., "Machine-type Communications: Current Status and Future Perspectives toward 5G Systems," in *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 10-17, September 2015.

[2] J. Song *et al*., "Connecting and Managing M2M Devices in the Future Internet," in *Mob. Netw. Appl.*, vol. 19, no. 1, pp. 4-17, February 2014

[3] ETSI, "Mobile-edge computing introductory technical white paper," White Paper, Mobile-edge Computing Industry Initiative. [Online]. Available: https://portal.etsi.org/portals/0/tbpages/mec/docs/mobile-edge_computing_-_introductory_technical_white_paper_v1

[4] V. P. Kafle, Y. Fukushima and H. Harai, "Internet of things standardization in ITU and prospective networking technologies," in *IEEE Commun. Mag.*, vol. 54, no. 9, pp. 43-49, September 2016.

[5] J. Swetina, G. Lu, P. Jacobs, F. Ennesser and J. Song, "Toward a Standardized Common M2M Service Layer Platform: Introduction to oneM2M," in *IEEE Wireless Commun.*, vol. 21, no. 3, pp. 20-26, June 2014.

[6] T. Taleb, *et.al.*, "Towards Multi-Access Edge Computing: A Survey of Emerging 5G Network Architecture & Orchestration", IEEE Comm. Surveys & Tutorials, Vol.19, No.3, 3rd Quarter 2017.

[7] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing—A key technology towards 5G," ETSI White Paper, vol. 11, 2015.

[8] M. Peng, Y. Li, Z. Zhao and C. Wang, "System architecture and key technologies for 5G heterogeneous cloud radio access networks," in *IEEE Network*, vol. 29, no. 2, pp. 6-14, March-April 2015.

[9] NGMN Alliance, "Description of Network Slicing Concept," Tech. Rep., 2016.

[10] 3GPP TS 23.502 "Procedures for the 5G System; Stage 2, (Release 15)", v0.6.0, Aug. 2017

[11] P. Rost, et al. "Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks," IEEE Communications Magazine, vol. 55, no. 5, pp. 72-79, May 2017.

[12] 3GPP TS 38.300, "NR; Overall description; Stage-2," ver. 1.0.1, Sept. 2017.

[13] BBF TR-221, Technical Specifications for MPLS in Mobile Backhaul Networks, 2011.

[14] S. Bryant, J. Dong, Enhanced Virtual Private Networks (VPN+), IETF Draft, Jul. 2017.

[15] N. Finn, P. Thubert, B. Varga, J. Farkas, Deterministic Network Architecture, IETF Draft, Aug. 2017.

[16] 3GPP TR 28.801, Study on management & orchestration of network slicing for next generation network", Rel.15, May 2017.