End-to-end Mobile Network Slicing

Ibrahim Afolabi

School of Electrical Engineering

Thesis submitted for examination for the degree of Master of Science in Technology.

Espoo 31.07.2017

Thesis supervisors:

Prof. Tarik Taleb

Thesis advisor:

PhD. Miloud Bagaa

Aalto University
School of Electrical
Engineering

AALTO UNIVERSITY
SCHOOL OF ELECTRICAL ENGINEERING

ABSTRACT OF THE
MASTER'S THESIS

Author: Ibrahim Afolabi

Title: End-to-end Mobile Network Slicing

Date: 31.07.2017 Language: English Number of pages: 8+56

Department of Communications and Networking

Professorship: Architectural enhancements to mobile core networks

Supervisor: Prof. Tarik Taleb

Advisor: PhD. Miloud Bagaa

Wireless networks have gone through several years of evolution until now and will continue to do so in order to cater for the varying needs of users. These demands are expected to grow in the future, both in size and variability. Hence, the 5G technology considers these variabilities in service demands and potential data explosion which could accompany users' demands at the core of its architecture. For 5G mobile network to handle these foreseen challenges, network slicing [12] is seen as a potential way forward as its standardization progresses. In light of the proposed 5G network architecture and to support an end-to-end mobile network slicing, we implemented radio access network (RAN) slicing over a virtualized evolved Node B (eNodeB) and ensured multiple core network slices could communicate through it successfully. Our results, challenges and further research path are presented in this thesis report.

Keywords: EPS, NFV, SDN, RAN slicing, EPC, eNodeB

Preface

In the Name of Allah the Most Beneficent, the Most Merciful.

I give absolute thanks and exaltations to Allah alone and I will forever remain grateful to Allah who has spared my life till date and given me the opportunity to complete my masters program.

I would like to acknowledge my parents Alhaji and Alhaja Afolabi for believing in me to pursue my dreams and always supporting me with regards to my academics and other important aspects of my life.

To the Minister for home affairs and general matters (My wife and Doctor) and her assistant (my princess) who have shown unconditional love and support. Their prayers and most importantly their endurance, braving through my late nights home coming and sometimes early morning departures.

To my Professor/supervisor Prof Tarik Taleb, for giving me the chance to work under his supervision and creating a warm and helpful environment to work in, for his guidance and unrelenting desire to make me succeed by all cost.

I will also like to acknowledge my senior colleague, advisor, friend and brother in Islam Dr. Miloud Bagaa for his unwavering support and always lending a helping hand when needed the most.

Finally, my appreciation goes to the entire members of the MOSA!C Lab in particular, and the COMNET researchers in general, including the IT support team for their understanding and support throughout my master's studies.

To all, I say Jazaakumullahu khayran! (May the Almighty God reward you all in abundance) .

Otaniemi, 31.07.2017

List of Papers and Deliverables contributed to by the author

0.1 Papers

I. Afolabi, A. Ksentini, M. Bagaa, T. Taleb, M. Corici and A. Nakao,“Towards 5G Network Slicing over Multiple-Domains", IEICE Transactions on Communications, 2017.

I. Afolabi, M. Bagaa, T. Taleb, H. Flinck, “End-to-End Network Slicing Enabled Through Network Function Virtualization", To appear in IEEE CSCN 2017, Helsinki, Finland, September 2017.

0.2 Deliverables

Architecture description of 5G NFV core network with SDN, MEC and Wi-Fi access and the Proof-of-Concepts, Release 2

5G!Pagoda D3.1–Slice Components Design–ver.1

5G!Pagoda D2.3: Initial report on the overall system architecture definition

5G!Pagoda D2.1–UseCase Scenarios, and Technical System Requirements Definition –ver.1.1

Contents

List of Figures

Abbreviations

AKA	Authentication and Key Agreement
DC	Data Center
DPI	Deep Packet Inspection
E2E	End-to-End
EMM	EPS Mobility Management
eNodeB	E-UTRAN NodeB
EPC	Evolved Packet Core
EPS	Evolved Packet System
E-RAB	E-UTRAN Radio Access Bearer
ESM	EPS Session Session Management
ETSI	European Telecommunications Standard Institute
E-UTRAN	Evolved UTRAN
GRE	Generic Routing Encapsulation
HSS	Home Subscriber Server
IaaS	Infrastructure as a Service
LTE	Long Term Evolution
MME	Mobility Management Entity
NFV	Network Function Virtualization
NFVO	Network Function Virtualization Orchestrator
NFV-MANO	Network Function Virtualization Management and Orchestrator
PCRF	Policy Charging Rule Function
P-GW	Packet Data Network Gateway
RAT	Radio Access Technology
RAN	Radio Access Network
SGSN	Serving GPRS Support Node
S-GW	Serving Gateway
SDN	Software Defined Networking
UE	User Equipment
VM	Virtual Machine
VNF	Virtual Network Functions
VM	Virtual Machine Monitor

1 Introduction

Several studies have revealed the numerous benefits of network resources sharing both on the capital and operational expenditures levels. In particular, Radio Access Network (RAN) slicing which is an active form of network sharing is rapidly gaining momentum within telecommunications standardization bodies, so much so that, it is being touted that it would be an intrinsic part of the next fifth generation (5G) system architecture. For this to be possible, there are a number of essential conditions and requirements to be carefully considered and catered for. These requirements as mentioned in [1] are Slice Isolation, Customization and Resources Utilization. To fulfill these requirements, wireless virtualization concepts such as Wireless Network Virtualization (WNV), Software Defined Networking (SDN) and Network Function Virtualization (NFV) are seen as possible candidates and key enablers to actualize this standard [2]. But before we proceed, it is crucial to have a good understanding of what slicing means in the context of network resources virtualization. Slicing is a multidimensional concept with varying definitions depending on the application approach. But in a more general term, and from a flow point of view, a slice could be regarded as a stream of IP packets belonging to different end users sharing a wireless service such that while flows of multiple users are supported, users can also obtain services from multiple slices [2]. It is a mechanism that allows the sharing of a single network infrastructure between multiple network operators, whereby each operator provides its own unique functionalities and services to fulfill its users needs [3]. Indeed, a complete Evolved Packet System (EPS) architecture on a high level consists of three major parts, namely the UE, Radio Access Network (RAN) consisting of the Evolved Node B(s) (eNodeBs) and the core network part also called the Evolved Packet Core (EPC). In this context, we imagine an E2E network slicing as the slicing of all these network parts.

In order to actualize network slicing as a potential solution for the 5G system characterized with varying user demands, we certainly know that a number of architectural challenges have to be considered. Especially, when considering the fact that a mobile network is usually identified by its core network. This means that for multiple mobile network operators to share a commodity network infrastructure inform of network slices, each of the sharing tenants has to run on different virtual platforms hosted on the same hardware infrastructure while sharing the access network infrastructure. These virtual platforms will basically house the respective EPCs of the sharing partners. Bearing in mind that the virtual EPCs do not necessarily have to be orchestrated from exactly the same set of network resources, it is important to note that the resources to be allocated to each of the EPCs may be determined by the service demands each of them has to fulfill. Also, their service demands will determine the amount of access network resources in form of physical resource blocks to be allocated. In particular, this thesis focuses on determining the flavour of virtual machine and the right amount of resource blocks which is most suitable to host a virtual EPC considering the business requirements the virtual EPC has to accomplish, and develop an architecture for the orchestration of these resources.

In developing the system architecture, technologies such as WNV, SDN and NFV, would be leveraged for designing and automating of the framework. The proposed system architecture aims to reduce Operational Expenditures (OPEX) while ensuring the Service Level Agreement (SLA). Indeed, the different VNFs would be placed in appropriate locations (cloud network or edge cloud) and the appropriate resources (CPU and memory) would be used to ensure the proper functionality of each E2E core network slices while reducing the costs. Moreover, an auto-scaling (scaling up/down and scaling in/out) mechanism is adopted for reducing the cost further. To the best of our knowledge, we are the first to practically enable the E2E mobile network slicing especially in terms of slicing a virtual RAN. For enabling RAN slicing, we have updated the OAI RAN software to allow the connection of a single eNodeB to multiple core networks. Two core networks VNFs were deployed in our experimental setup: *i*) OAI core network; *ii*) Aalto core network. Slicing the RAN entails sharing the radio resources, ensuring that the traffic of the tenant EPCs are isolated from each other both on the control and data planes and manipulating the MAC scheduler of the RAN so that it allocates the right amount of physical resource blocks to the users of the networks belonging to the tenants.

The proposed framework, herein, mainly consists of EPCaaS orchestrator, SDN controllers (Opendaylight and/or ONOS) and NFV orchestrators. For each EPCaaS slice creation request, the EPCaaS orchestrator instructs the NFV orchestrators to create different VNFs' instances in different network cloud, as well as it instructs the SDN controllers to interconnect those instances. Besides the aforementioned contributions, we have also performed a limited set of benchmarking for matching between the features of E2E mobile network slices and their required resources in terms of CPU, memory, bandwidth and latency. We plan to extend these experiments later by taking into account multiple eNodeBs and EPCs in the networks. The obtained results augmented with some mathematical tools, such as machine learning, the EPCaaS orchestrator will be able later to make the right decisions in placing different VNFs in the right locations with the right resources.

2 Background on virtualization and Network slicing

In the context of virtualization, network softwerization, based on Software Defined Networking (SDN) and Network Function Virtualization (NFV), represents the major enabling technologies towards 5G; allowing the creation of virtual network flavours customized towards the service requirements. In this light, an efficient integration of SDN and NFV within cloud computing ensures multiple advantages in terms of network configuration, flexibility, scalability, and elasticity, which are highly needed to build the dedicated slices concept next to the usage of the same physical resources for the multiple dedicated networks. Generally speaking, a mobile network slice is composed of a number of Virtual Network Functions (VNF) chained together and connected to at least one Radio Access Technology (RAT) to deliver a complete mobile network functionality, customized to suit the particular requirements of a service. The VNFs cover both the Control plane and User plane functions of the Core Network components (e.g. Mobility Management, Authentication, Forwarding, etc.) and the Radio Access Network (RAN) (e.g. Optimal splitting, PDCP-C, PDCP-U, etc.) and may include application specific enablers. Moreover, the VNFs may also include authorized legal Deep Packet Inspection (DPI), Firewall and caching functions and storage resources as well as application specific enablers up to even applications.

The concept of Network Slicing is not new, and found its foundation in the Infrastructure as a Service (IaaS) cloud computing model and overlay networks. In the IaaS model, the aim behind network slicing is to share a computing, networking and storage infrastructure between different tenants, in order to build a self-contained virtual network infrastructure with a required level of isolation. A Network slice is composed of different virtual machines (VMs) connected together on a Virtual network enabled by using Virtual LAN (VLAN) technology in a system setup where the VMs are in the same Data Center (DC); or VLAN and tunneling protocol, like Generic Routing Encapsulation (GRE) and VxLAN (x: extended) in a system setup where the VMs are hosted on different DCs.

2.1 Network Function Virtualization Technology

Network Function Virtualization promotes the idea of removing network functions from dedicated physical network hardware equipments to run on any virtualized server environment deployed on any location on the network. This could make it possible to decouple network functions running on proprietary network devices to run on centralized and virtualized network platform which could be deployed at anytime on the network with respect to the network needs and service requirements [49].

Using a standard architecture such as the European Telecommunications Standard Institute's (ETSI) framework, network functions which are needed to enable and instantiate network slices can be orchestrated from virtual network resources as virtual network functions (VNFs). This network functions are originally deployable on dedicated hardwares and run as proprietary functions which are tightly coupled with the hardwares they are running on. Thanks to the NFV technology architecture,

these network components can now run on commodity off the shelf servers. The NFV architecture is briefly discussed in the following subsection.

2.1.1 The ETSI Framework Architecture for Network Function Virtualization

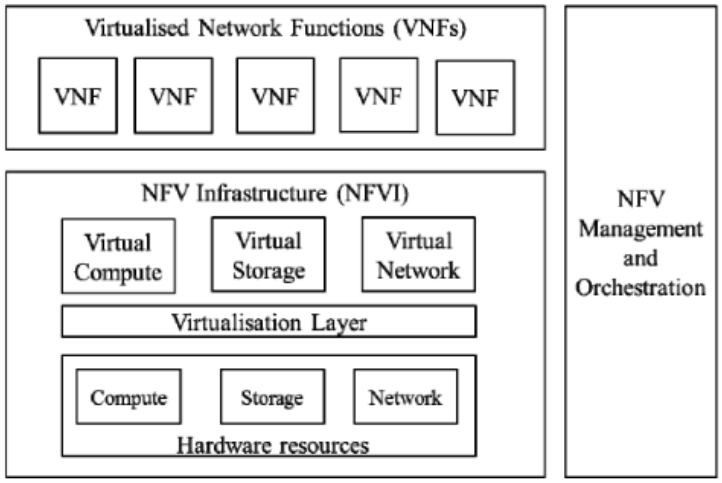


Figure 1: The ETSI Architectural Framework [17].

The European Telecommunications Standard Institute's (ETSI) framework architecture for network function virtualization presented in figure 1 illustrates the architecture in the simplest form it can be. The framework consists of three major building blocks, the NFV Management and Orchestration (NFV MANO) building block, the Virtualized Network Functions (VNFs) building component and the NFV Infrastructure (NFVI) fundamental element.

The NFV MANO is the brain and intelligence of the system. It is equipped with all the necessary management and control logic of the system. It is the master piece that dictates instructions about network resource requests to the NFVI. It jobs encompasses the orchestration of network resources both physical and software needed to support the infrastructure virtualization of VNFs and the management of the lifecycle of the orchestrated resources and the VNFs using that are using them.

The VNFs are network functions which originally are designed strictly for certain hardware equipment and optimized to perform certain network tasks adequately in those hardware boxes. The Network functions could for instance be a simple firewall, network router, DHCP server or even a combination of all which are running in custom-made equipment and are physically installed on the customer's premises. By adopting the ETSI framework, these network functions can now be deployed as virtual functions running in virtual environments which are orchestrated from general purpose infrastructure such as servers running on commercial or private datacenters.

The NFVI is the element which houses both the physical infrastructure and the virtual resources which are abstracted from it. The virtual resources which are the

virtual compute, virtual storage and virtual network resources are abstractions of the real underlying physical infrastructure which are separated via a virtualization layer.

The virtualization layer sits above the real physical infrastructure, abstracts the hardware resources, controls and manage them in such a fashion that the virtual resources which are provided by the hardware resources abstraction, appear to the different VNFs has a complete isolated set of resources. This emulation of virtual system environment which has all the resources such as the compute, network, etc. which a normal physical computer or hardware infrastructure needs in order to support any set of functions which may be deployed on it is known as the Virtual Machine (VM). And the virtualization layer is known as the Hypervisor or the Virtual Machine Monitor (VMM).

2.2 Software Defined Networking Technology

was also described as an enabler in that is tries to separate the control plan from the data plane of network devices. The separation between the control plan functionalities from the data forwarding functionalities brings about the much needed flexibility needed to achieve an almost perfect RAN slicing implementation. If SDN is carefully deployed to manage wireless network slices, it could turn out to be the necessary tool needed to ease the complexity that could accompany the management and programmability of wireless network slices [49].

The flexibility that the SDN technology offers in data plane programmability and traffic engineering can not be overemphasized. The fact that it decouples the control plane functionalities from the plain data plane packet forwarding functionalities allows network operators the privilege to control and deployed new network functions and control policies in the data plane of the network. This would enable a faster role out and enforce of newer rule which would enhance the timely performance of the network. The SDN technology architecture is introduce in the following subsection.

2.2.1 The SDN Technology Architecture

The SDN architecture illustrated in figure 2 is divided across three planes, the Application, Control and data planes. In the middle of the architecture is the control plane which divides the entire framework into two equal layers, the north and the south. Sitting in the middle is a centralized controller known as the SDN controller. The SDN controller connects with the north layer using its northbound interface known as the SDN Northbound Interface (NBI) and connects with the south layer on the southbound interface known as the SDN Control-Data-Plane Interface (CDPI).

Centrally placing the SDN controller gives it a global view of the entire architecture and enables it to issue control instructions to the network elements sitting on the data plane. The network elements are programmable general purpose forwarding and processing function modules which connects with the centrally positioned command issuing master controller on the its CDPI driver. Using the CDPI driver, the SDN controller connects with the network elements on the CDPI agent in order to enforce network behavioural rules, give low level controls, acquire statistics about the network

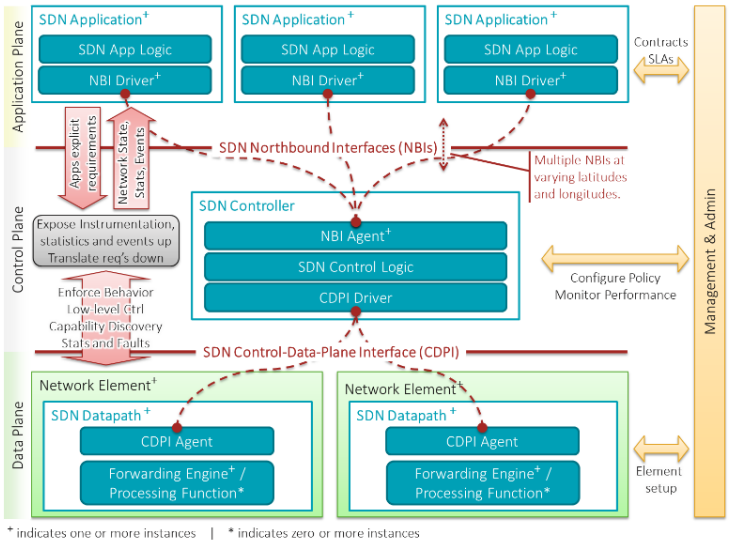


Figure 2: The SDN Architecture Framework [18].

and discover other network capabilities.

Similarly, the SDN controller has an NBI agent through which it communicates with the Application plane on the NBI driver of the SDN applications. Necessary and timely network instructions, control information and applications explicit requirements are passed on to the SDN controller from the SDN applications sitting on the application plane on the north layer of the SDN controller.

There is also a management and administrative component which is sitting on the right and spanning across all three planes, where all network management and administrative related functionalities are carried out. The network policies are configured and network performance are monitored from this component. Also SLAs contracts are reflected to the SDN applications as well as network element setup are performed for the network elements through this component.

Following this fundamental framework, the SDN technology is able to simplify the functionality of traditional proprietary hardware network equipment by detaching the control logic from the forwarding logic which normally deals only with routing user packets in the network. This give more flexibility and dynamicity to the overall functionalities and design of the network, thereby making it more robust, proactive and resilient in dealing with possible network failures and behavioural changes.

2.3 Towards enabling 5G networks requirements

In addition to cloud computing, the concept of slice in networking was also used in the overlay network research efforts, such as PlanetLab [33], where a network slice has been defined as an isolated set of network resources such as bandwidth, computational functions, storage capacity allocated for a group of users that "program" network functions and services over their overlay network overlaid across "the planet". The concept follows the Slice Federation Architecture [28] used in the large scale

GENI federation between the US research institutions. Since then, various network virtualization testbed efforts such as PlanetLab EU, PlanetLab JP, VNode, FLARE [30], Fed4Fire, have inherited the concept of slices as a basis of the infrastructures, as a set of programmable resources to create new network services and protocols. Network slicing in 5G shares some concepts with the cloud computing and overlay network, but it requires more management and orchestration procedures as well as adaptation to the mobile network characteristics, such as mobility, wireless changing resources, fronthaul capacity e.t.c.

5G systems are expected to build a mobile network architecture that supports not only classic mobile broadband applications but also specific vertical industry services, such as those of automotive systems, e-health, public safety, and smart grid which previously were supported through private dedicated networks [4]. Vertical services require different and incompatible performance parameter levels, which are difficult to achieve using the same physical infrastructure. For instance, automotive systems require high reliability added to low latency access to remote servers, public safety services need ultra-reliable and highly available system, while enhanced broadband access services require high bandwidth covering a dense area and massive IoT requires the cost efficient connection of a huge number of devices. Consequently, the envisioned 5G systems would need to re-architect the current uniform mobile architecture to allow multiple, logical, self-contained networks on a common physical infrastructure platform enabling a flexible stakeholder ecosystem that allows technical and business innovation integrating network and cloud resources into a programmable, software-oriented network environment.

Meanwhile, 5G systems should support a flexible and on-demand provisioning of network resources, network functions and applications, using a virtual resources layer spanning on top of physical resources of multiple domains. This will enable value added creation for vertical segments that would receive a wide area network support while being cost effective through the transition from dedicated networks to common cloud resources that can be used in an isolated, disjunctive or shared manner allowing customizable network operation. Finally, 5G is anticipated to shift the conventional networking paradigm away from the 4G mobile broadband ideal, wherein a single architecture fits all services, towards sliced network instances using as much as possible the same software components, however tailored to address particular service needs, maintaining in this way a truly differentiated service provisioning.

The network slices may sometimes require conflicting performance requirements. For example, a slice may require low latency, high bandwidth and high mobility, but would not care about reliability, such as slices for enhanced Mobile Broadband services (eMBB). Another slice may require low latency, high reliability and high traffic density but would not care about the bandwidth such are for Critical Communications (CriC) or others may require efficient communication and high traffic density but would neither care about reliability nor bandwidth such as the massive IoT [29]. These slice flavours may have to be deployed across a multi-domain environment consisting of a mixture of dedicated and shared slices, for this reason, advanced orchestration mechanisms should be explored to enable an elastic allocation of resources in an efficient manner over multiple domains.

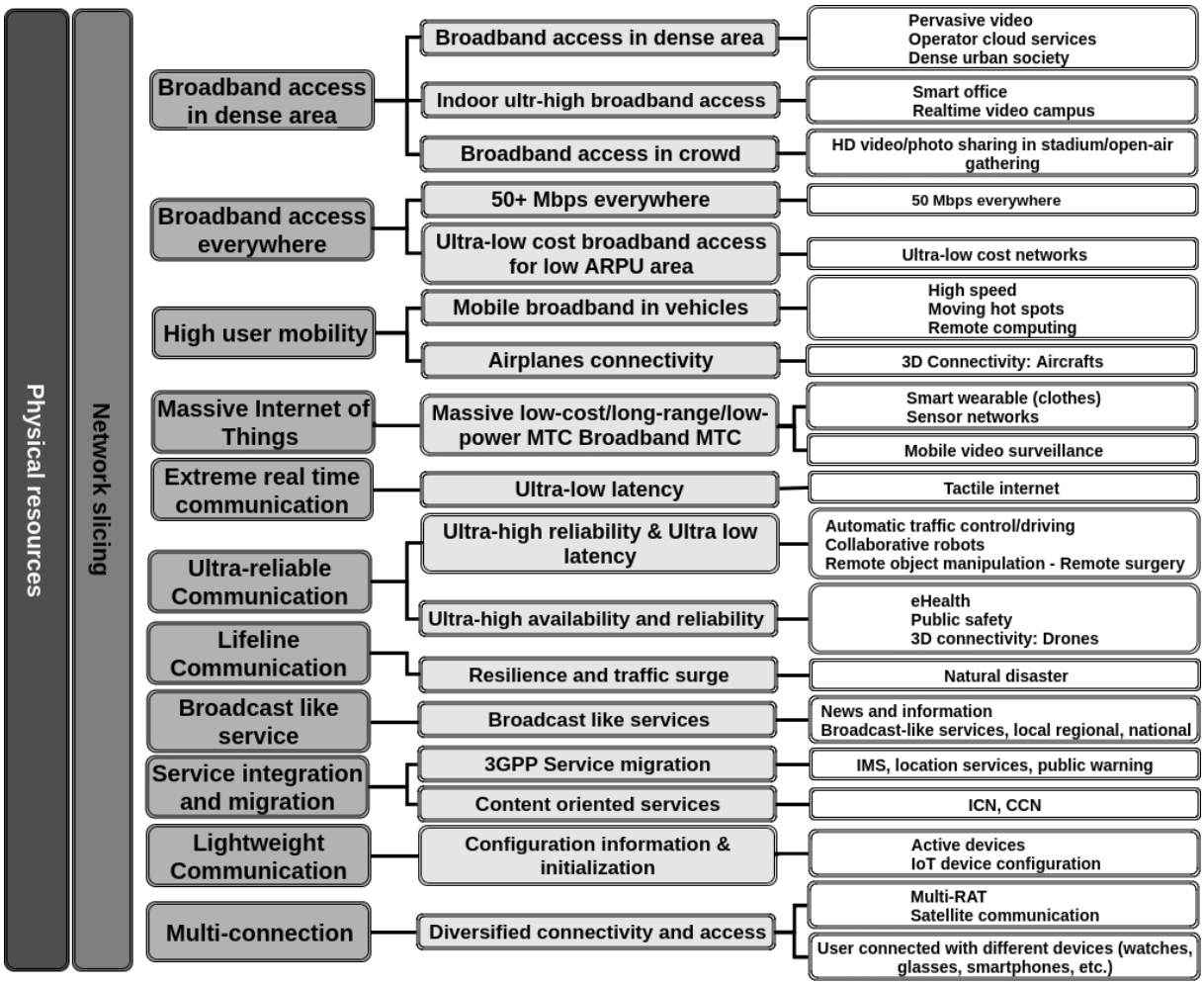


Figure 3: Use-cases family and category per 3GPP and NGMN [34].

Figure 3 summarizes the 5G system use-cases defined by 3GPP and NGMN. The 5G system use-cases are grouped into families, and each family includes one or more categories. Network slicing is involved in all the use-cases, as it is indispensable to enable all these use-cases concurrently over the shared physical infrastructure. Aiming at grouping the 5G use-cases in services, the METIS-II project[32] classifies the most representative 5G system use-cases according to their constraints. It shall be noted that the way of grouping 5G system use-cases has been later adopted by the 5GPPP document [31] on 5G system. METIS-II assumes that most representative 5G services may be classified in one of the following categories:

- Extreme or Enhanced Mobile Broadband (eMBB) [36] type, which requires both high data rates and low latency in some areas, and reliable broadband access over large areas. For example, dense urban areas require high bandwidth as users may upload HD video to their preferred social network application and also low latency because they may use Virtual Reality (VR), remote video presence and Augmented Reality (AR) streaming.

- Massive Machine Type Communication (mMTC) type, which needs efficient and reliable wireless connectivity for massive deployment of sometimes resource constraint devices. An example of this type of service is the deployment of a large number of sensors and actuators (over a million devices per square kilometers) to monitor or control a given area.

- Ultra-reliable and low-latency or Critical communications (uRLLC) [37] or ultra-reliable MTC, which covers all services requiring ultra-low latency connections. Notable examples include industrial control systems, real time control of vehicle and traffic, and public safety scenarios.

It is worth noting that other SDO (Standard Development Organizations) bodies, like 5GMF, has also defined 5G use-cases. In [35], 13 usage scenarios have been studied and expected to be realized in 5G mobile networks. They have been categorized into four facets; 1) Entertainment, 2) Transportation, 3) Industries/Verticals, and 4) Emergency and disaster relief ultimately giving a connected device perspective which spans across the three previous use cases combining multiple applications and services from each into a comprehensive service offering.

3 Background on Evolved packet System

3.1 EPS architecture

The Evolved Packet System (EPS) architecture depicted in figure 4 is a non-roaming model which is designed as a flat packet switched architecture, where all the interacting entities fundamentally communicate via IP addresses, hence, it is called an All IP network (AIPN). It broadly consists of two parts, the Radio Access part, termed the LTE (Long Term Evolution) and the Core Network part, dubbed the EPC (Evolved Packet Core). The EPC is also known as the system architecture evolution (SAE) of the 3GPP mobile network. While the radio access part (LTE) offers higher throughput and lower latency than its predecessors, the EPC supports mobility between multiple heterogeneous access network such as, Universal Terrestrial Radio Access Network (UTRAN) and GSM Edge Radio Access Network (GERAN). An LTE capable UE with the eNodeB are together called the Evolved UTRAN (E-UTRAN). Mainly, the support for higher throughput and lower latency amongst other features earned the LTE its position to be classified among the 4th generation mobile access networks. A major difference between the LTE eNodeB and its nodeB predecessors, is that, while the nodeB needs an additional component called the radio network controller (RNC) to handle users' mobility, 3GPP has already incorporated the complex functionalities of the RNC in the LTE enodeB standard, hence, dubbed evolved nodeB.

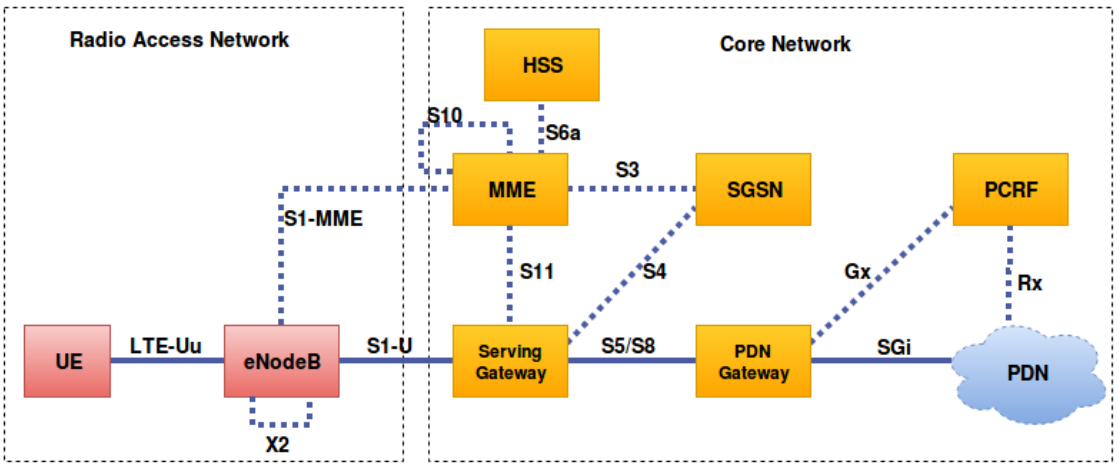


Figure 4: 3GPP Evolved Packet System Architecture.

3.1.1 The LTE RAN and Interfaces

As illustrated in figure 4, the LTE RAN consists of the eNodeB and the LTE capable UE. The UE is any electronic device e.g. a mobile phone, tablet, phablet or even a laptop which is equiped with a mobile broadband adapter capable of connecting to the eNodeB for communication purposes. The interface of communication between the UE and the eNodeB is the LTE-Uu radio interface. The protocol stack specified for data transmission over the LTE-Uu for both control and user plane information

is depicted in figure 5. While the user plane stack ends at the PDCP (Packet Data Convergence Protocol) layer and terminates at the eNodeB, the control plane stack extends with two additional layers, the RRC (Radio Resource Control) for control signaling with the eNodeB and the NAS-EPS (Non-Access Stratum) which terminates at the MME.

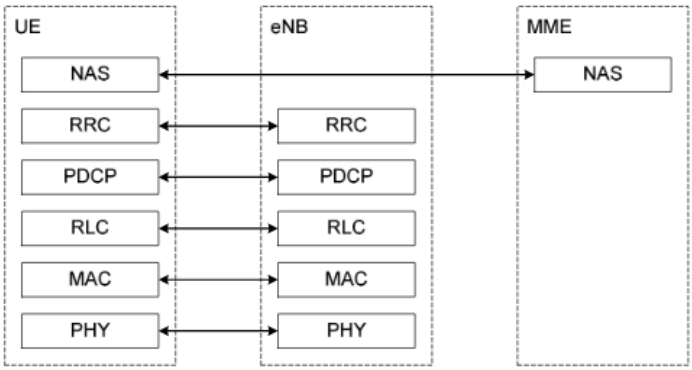


Figure 5: LTE-Uu Control and User Protocol stack [38].

The eNodeB communicates with other neighbouring eNodeBs on the X2 interface and connects with the core network, in particular the MME on the S1-MME interface for control signals exchange and the Serving Gateways (S-GW) on the S1-U interface for user plane data transmission. The LTE eNodeB uses the set of protocols illustrated in figure 6 to achieve mobility (handover) support between LTE RANs for ECM-CONNECTED users, carryout interference coordinations between cells, information exchange between eNodeBs, load management, etc between connected eNodeBs [39]

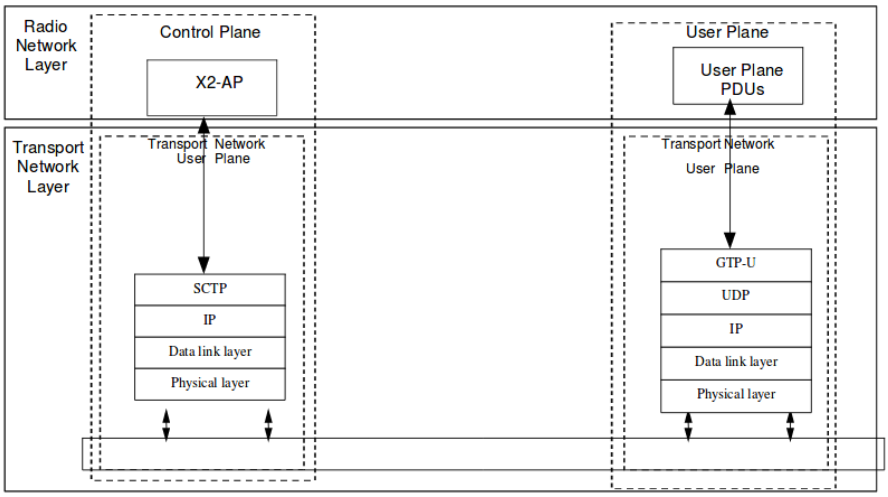
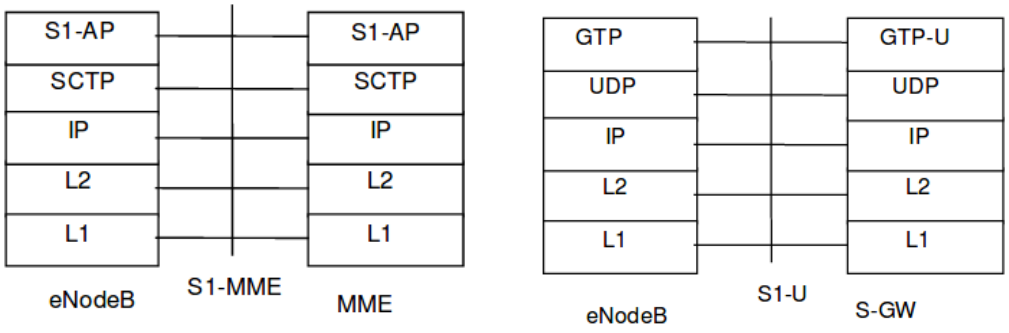


Figure 6: X2 Interface Protocol stack [39].

The S1-MME is basically the main signaling interface between the E-UTRAN and the EPC. It is used for the transmission of control related information traffic between

the eNodeB and the MME Using both SCTP (Stream Control Transmission Protocol) [42] and the S1AP (S1 Application) protocol specified in [40], hence, also called the S1-C interface. Its set of protocol structures as specified in [41] is illustrated in the figure 7(a).



(a) S1-MME Protocol stack (b) S1-U Protocol stack

Figure 7: S1 Protocol stack [41].

On the E-UTRAN, the another interface linking the eNodeB to the EPC is the S1-U interface. It is the reference point between the eNodeB and the EPC's S-GW used for conveying user plane information. It is setup using the GTP-U (GPRS Tunneling Protocol User-Plane) protocol during the EPS default bearer setup procedure. The protocol structure used for the S1-MME interface setup is shown in figure 7(a) while figure 7(b) shows the S1-U's protocol stack.

Finally we have the

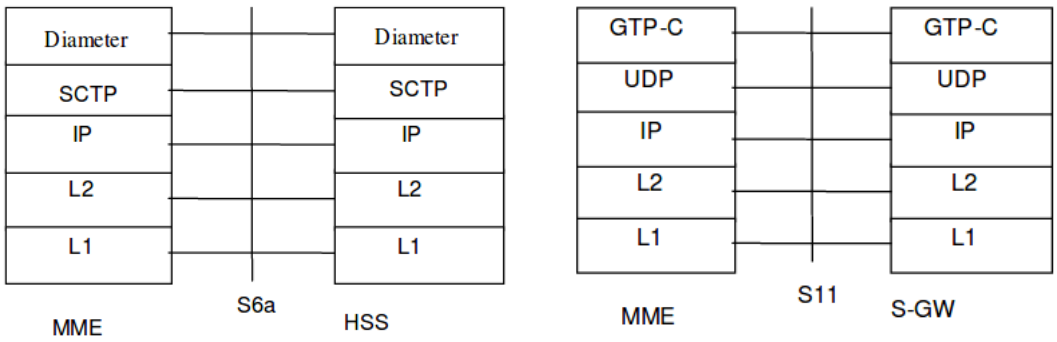
3.1.2 The Evolved Packet Core and Interfaces

The 3GPP specified evolved packet system termed its core network the evolved packet core. As illustrated in figure 4, it consists of five to six major components depending on the mobile network operator's implementation. The Mobility Management Entity (MME), Home Subscriber Server (HSS), Serving Gateway (S-GW), Packet Data Network Gateway (P-GW), Policy Charging Rule Function (PCRF) and the Serving GPRS Support Node (SGSN). The SGSN is out of the scope of this literature in that it belongs to the third generation mobile network system, however, it performs functionalities similar to the MME. The number of components may be considered five due to the standardization flexibility in the implementation and design of the EPC which allows mobile network operators to merge the functionalities of the S-GW and P-GW to become a single component dubbed SP-GW[41].

The MME is the single most important component in the EPC. In fact, it is the heart of the any mobile operator's system because virtually all control signals go through it and it is the only point of identity for any mobile network operator. It is the most essential processing node in the core network which coordinates and manages all the resources between the E-UTRAN and the EPC. Its main responsibilities are the management of the UEs' sessions, coordination between eNodeB's to manage

UEs' mobilities, coordination with HSS to manage security and authentication of UEs.

The MME is the single EPC node with the highest number of connection interfaces. Its reference point to the eNodeB is the S1-MME interface. It exchanges signaling data with the HSS through the S6a interface and initiates and maintains communication with the S-GW through the S11 interface. It coordinates with the SGSN to maintain and manage UEs' session and mobility for third generation technology capable UEs sharing the same network using its S3 interface. In a deployment scenario involving a pool of MMEs belonging to the same mobile network operator, the MMEs communicate with each other over their S10 interface. In a nutshell, the MME alone has five communication interfaces used for managing mainly the control plane data transmission.



(a) S6a Protocol stack

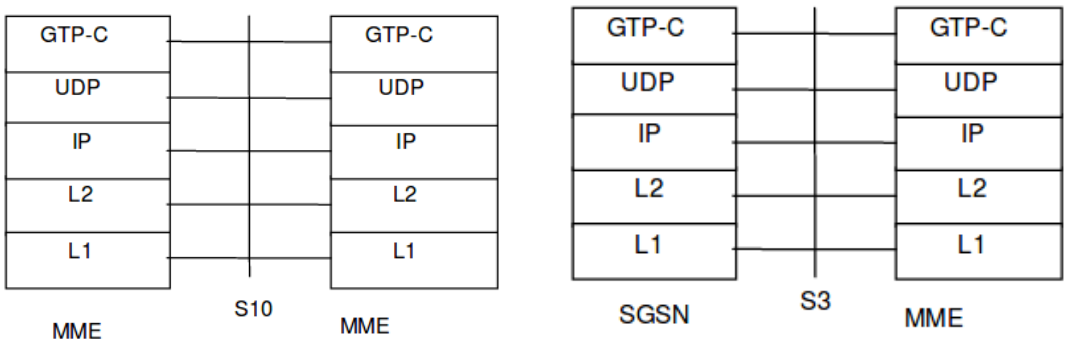
(b) S11 Protocol stack

Figure 8: S6a and S11 Protocol stack [41].

The S1-MME and its protocol stack has already been discussed in section 3.1.1 above. The S6a interface is a major control plane connection interface. It is used basically for Authentication Authorization and Accounting (AAA) and the UE location information update using the Diameter [43] application protocol which is established over the SCTP protocol as depicted in figure 8(a). It is a very important link through which the MME authenticates any UE seeking the PDN connectivity and Attach request procedure from the network using the UE's subscriber information. This information is securely authenticated against what the network operator has registered in the HSS.

The S11 interface is also very essential, it is the communication link between the MME and the S-GW. It is used by the MME to communicate the network resources need of a UE to the S-GW during the EPS bearer establishment. Since the MME does not have a direct link to the P-GW, it uses the S-GW through the S11 interface to allocate data plane resources and routes for the connected UE using the GTPv2-C (GPRS Tunneling Protocol version2 for Control Plane) protocol [44] above the UDP transport protocol to exchange control information as shown in figure 8(b).

Using the S10 interface, multiple MMEs communicate with each other. These MMEs usually belong to the same MME pool area. Depending on the mobile network operator's EPC configuration and deployment style, the MMEs could be used in the



(a) S10 Protocol stack

(b) S3 Protocol stack

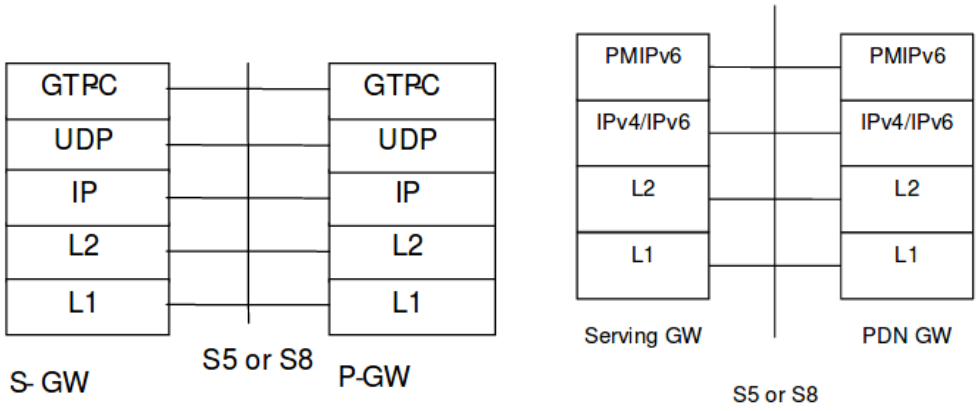
Figure 9: S10 and S3 Protocol stack [41].

same area for the purpose of load balancing or could be distributed across the areas the mobile operator supports for the purpose of MME relocation in order to offer a robust support for UE mobility. In any case, the control signal regarding connected UEs are exchanged between the MMEs over the S10 interface. Similar to the S10 interface, is the S3 interface where control information are as well exchanged between an MME and SGSN belonging to the same mobile network operator, in order to provide mobility support for UEs using 3G technology. The protocol structure for both the S10 and S3 interface is presented in figures 9(a) and 9(b) respectively.

The HSS is the database containing all the subscribers' information in the home network. It is the only register from where UE information can be stored, updated and retrieved. Depending on the mobile operator's deployment scenario, it can be either deployed as a single node or as a set of synchronized multiple nodes for the purpose of enhancing the performance. In coordination with the MME, it is used to store subscription profiles of UEs with respect to the type of services the network can provide for them. Also, the MME use the HSS to track UE's mobility (for mobility support) and for location updates. As discussed above, using the S6a interface, the HSS communicates with the MME to authenticate UE's identities during the ME (Mobile Equipment) identity check procedure [41] using the well known EPS AKA (Authentication and Key Agreement) challenge specified in [45].

The Serving Gateway (S-GW) is the local mobility anchor point of every UE which is attached to the EPS. At any given point in time, there can only be one S-GW associated with a UE. It does the job of a router and forwards packets originating from a UE to the assigned P-GW and vice versa. It is also used for lawful interception of packets. The role of the S-GW is very vital especially when it comes to accounting and ensuring QoS for based on the UE subscription profile. The S-GW also plays a vital role in ensuring a successful UE handover procedure by sending end-marker(s) to the eNodeB that initiated the handover procedure in order to assist in packet reordering after carrying out the path switching functionality[41]. The S-GW is the only source through which the MME could reach the P-GW. The P-GW maintains communication with the S-GW using the S5/S8 interface. And exchange control signals with the SGSN using the S4 interface. However, due to the scope limitation

of this article, the S4 as well as the SGSN will not be discussed.



(a) S5/S8 GTP-C based Protocol stack (b) S5/S8 PMIP based Protocol stack

Figure 10: S5/S8 protocol stack based on both GTP-C and PMIP protocols[46, 46].

The S5/S8 interface is very crucial in establishing the data plane communication path for any attached UE. The S5/S8 interface uses the GTP-C protocol to tunnel signaling messages between the S-GW and P-GW for the control plane data and the GTP-U protocol to tunnel user plane data belonging to UE after the established of the S5/S8 bearer as part of the EPS bearer setup procedure. Another protocol which can be used for communication on this interface is the PMIP (Proxy Mobile IP) protocol. Both the PMIP and GTP-C based protocol stack are detailed in [41] and [46] as shown in figures 10(a) and 10(b) respectively.

The PDN Gateway is the gateway which terminates the UE to any PDN network. Unlike the S-GW, a UE can be associated with multiple P-GWs at any point in time depending on the subscriber's profile information and if its accessing multiple PDNs. The P-GW is responsible for allocating IP addresses to the UEs during the PDN connectivity and Attach request procedure. IP addresses can be allocated using a pool of predefined set of IP addresses or by consulting a DHCP server[41] . The P-GW together with the S-GW, is responsible for per-user based packet filtering and lawful packet interception. It is responsible for enforcing the service-level DL (Down Link) and UL (Up Link) rate limiting and inter-operator charging support for the purpose of account. The P-GW communicates with both the PCRF and a PDN (e.g., Internet) node through the Gx and the SGi interfaces.

The Gx interface is the main link of communication between the P-GW and the policy and charging rule function (PCRF) node. It provides the policy and charging enforcement function (PCEF) residing in the P-GW with the necessary policy and charging control (PCC) rules to be enforced on a UE from the PCRF node based on the QoS the network is willing to offer the UE.

The SGi interface on the other hand is the reference point which terminates the EPC towards a packet data network. The is the interface which exposes the UE's allocated IP address to an external packet data network. The PDN may be a network for the provisioning of IMS services for the UE.

The PCRF is the policy and charging entity of the EPC. It terminates two interfaces, the Gx and the Rx. They are of two types, the H-PCRF (Home PCRF) and the V-PCRF (Visited PCRF) depending on the deployment scenario. The H-PCRF is deployed within the Home PLMN (Public Land Mobile Network) while the V-PCRF is deployed within the Visited PLMN (Public Land Mobile Network). The functionalities of the PCRF are detailed in [47].

The final interface is the Rx interface which terminates the PCRF towards an IP packet data network which may belong to the mobile network operator, such as the operator's IP services like IMS (IP Multimedia Subsystem) and PSS (Packet Switch Stream).

3.2 EPS Control and User Planes

The description given above focuses more on the individual interacting components of the EPS architecture and their interfaces, however, the EPS architecture also has a broad division along the lines of the functions and information flows especially from the point of view of the EU. This information flows and major functional descriptions of entities has broadly divided the EPS architecture into two planes, namely, the Control and User planes entities. Though, a number of the components and their interfaces have already been described indirectly along the user and control plane divisions, but it is necessary to also show this division from the perspective of the UE and in order to better understand the functionalities of some signaling protocol.

3.2.1 EPS Control Plane

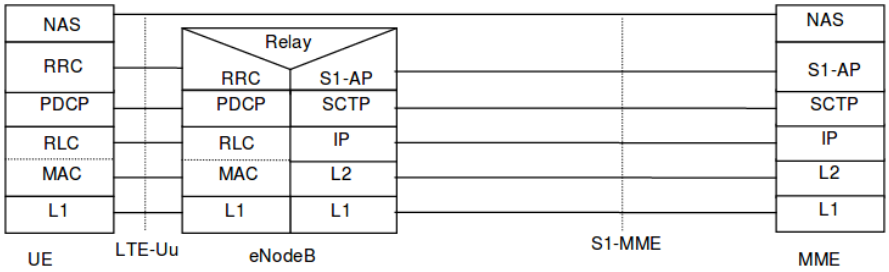


Figure 11: Control Plane UE - MME. [41]

The EPS control plane is made up of the components which interact using the dotted lines across their interfaces as shown in figure 4. These components are mainly involved in exchange of control signals. They use control protocols in their communication interfaces and carry out control functions and supports for the establishment of the user plane functions. The components are the eNodeB, MME, S-GW, HSS, SGSN and PCRF. While some of them are basically for control signaling, such as the MME, HSS, SGSN and PCRF, others also participate actively in the user plane such as the eNodeB and S-GW.

From the point of view of the UE, the control plane and the interacting protocol stack is seen as in the figure 11. This is the view of the UE using the LTE-Uu interface across the eNodeB all the way to the MME. As aforementioned in section 3.1.1 and shown in figure 5. It is important to that while the UE communicate directly with the MME over its LTE-Uu interface using the NAS application protocol as specified in [52], the eNodeB on the other hand, exchanges and maintains control signals with the MME using the S1AP protocol as specified in [40] over the S1-MME interface. These two signaling protocols are extremely crucial towards establishing UE's user plane communication path across the eNodeB, S-GW, P-GW all the way to the PDN, as a result, both of them will be described in subsequent subsections.

3.2.2 EPS User Plane

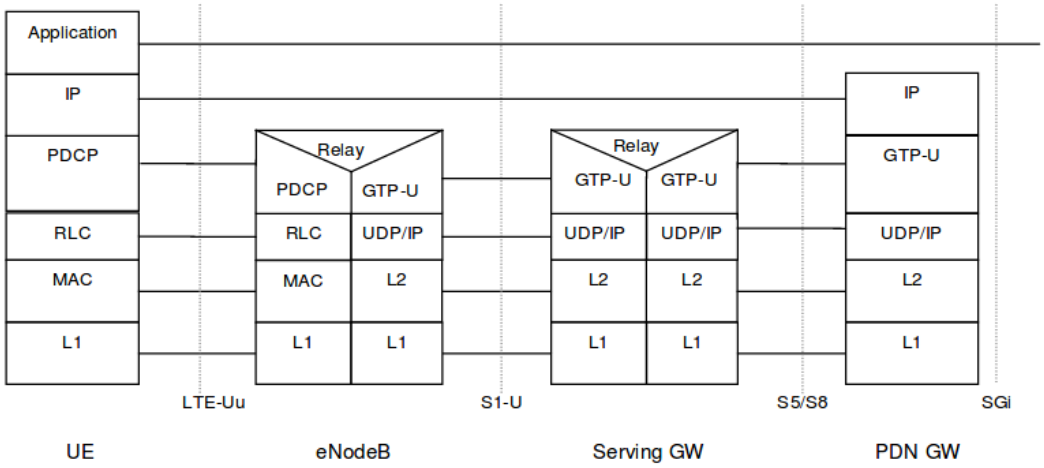


Figure 12: User Plane UE - eNodeB - S-GW - P-GW. [41]

Figure 12 shows the full end-to-end protocol stack of the EPS architecture's user plane. The EPS entities involved in the establishment of the user plane path from the UE's perspective are the eNodeB, S-GW and P-GW as illustrated in 4 with the thick continuous line. This user plane path establishment between the UE and P-GW is also known as the EPS bearer. This bearer is established over the IP protocol between the UE and the P-GW at the completion of the **PDN connectivity and Attach request** procedure. The Attach request is a major procedure of interest with through which we could determine whether or not a UE's request to attach to an EPS is successful or not. Hence, it shall be discussed later in subsequent sections.

The EPS bearer is made up of three main bearers, namely the S5/S8 Bearer, S1 Bearer and the Radio Bearer. The combination of the Radio Bearer and the S1 Bearer forms the Evolved Radio Access Bearer (E-RAB). The EPS bearer can either be default or dedicated bearer depending on the QoS Class Identifier (QCI) signified in the UE subscription information [38] . The default bearer is an always-on connection between the UE and the EPC and remains alive throughout the PDN session. The network is responsible for modifying a dedicated bearer for the same

PDN with an already existing default bearer depending on the QCI of the UE. The layered EPS service architecture is presented in Figure 13.

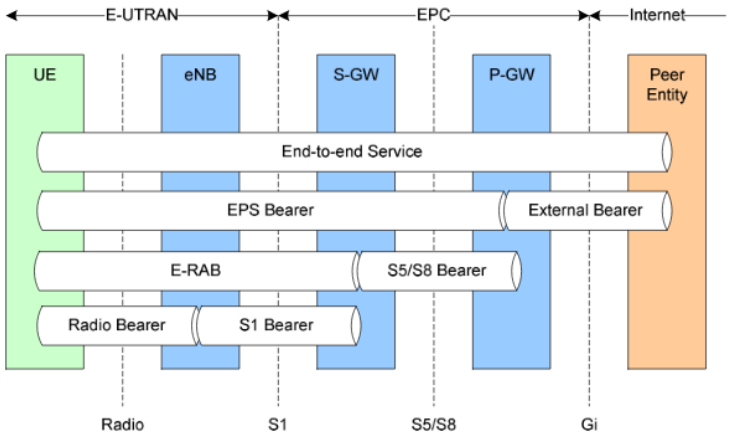


Figure 13: EPS Bearer Architecture. [38]

As illustrated in Figure 13, the Radio Bearer is responsible for transporting user plane packets from coming from the UE to the eNodeB. In addition to that, the S1 Bearer is responsible for tunneling the packets from the RAN to the EPC. When the packets get to the EPC, the S5/S8 bearer is responsible for transporting them from the S-GW to the P-GW. Once the packets get to the P-GW and onto the SGi interface, they will be transported to the Internet using the External Bearer. The EPS Bearer setup ends at the P-GW, and the combination of the EPS and the External Bearers form the End-to-end Service establishment path.

3.3 S1AP and NAS Protocol Description

As mentioned in the section 3.2.1, the NAS and S1AP protocols are in fact one of the most important protocols in the EPS. The description of the EPS will be incomplete without discussing the importance of these two protocols. While former is only critical to the UE's communication with the network, the latter is indeed quintessential to a successful exchange of information between the E-UTRAN and the EPC.

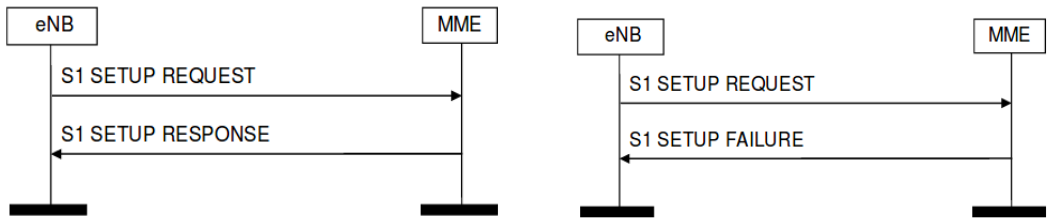
3.3.1 The S1AP Protocol

The S1AP protocol is very essential in maintaining signaling connection between the MME and the eNodeB over the S1-MME interface. However, this protocol does not only provide functions to support communication between the eNodeB and MME, but its supports also extends to the UE services. Therefore, the S1AP signaling support is divided into two major categories:

- UE associated services - These are S1AP functions that provide signaling support related to a single UE at a time. They are a set of S1AP functions

which directly support UE-associated signaling connection that is maintained for a particular UE.

- Non-UE associate services - These are S1AP functions that provide signaling support for the entire S1 interface between the eNodeB and the MME, while using a Non-UE associated connection.



(a) S1Setup Request Success Scenario (b) S1Setup Request Failure Scenario

Figure 14: S1Setup Request Procedures [40].

The S1AP protocol service functions comprising both the UE associated and Non-UE associated services are many, but a number of important ones will be enumerated [40] :

- S1 interface management functions such as S1Setup functionality for setting up the S1 interface between the eNodeB and MME and provide the necessary configurations for the initial setup.
- Provision of transport function for NAS signaling between the UE and the MME.
- E-RAB management functions, this includes functionalities necessary for setting up, modifying and releasing E-RAB.
- Mobility management functions, this includes functionalities needed to manage the mobility of UEs which are in LTE_ACTIVE mode with respect to handover.
- UE context modification function, for modification of the established UE context.
- S1 UE context release function, for managing the release of UE specific context.
- Paging function, for paging the UE.

When an eNodeB initiates a connection request to an MME, it usually sends the S1SetupRequest message which include all the necessary information needed by the MME to grant the request. On success, the MME replies the eNodeB with S1SetupResponse. However on failure, the MME replies the eNodeB with S1SetupFailure and the message will include the reason for the failure [40]. Both the S1setupRequest success and failure scenarios are illustrated in Figures 14(a) and 14(b) respectively.

3.3.2 The NAS Protocol

The NAS protocol is equally important for the UE to maintain control signaling connectivity with the MME over the radio interface. As illustrated in Figure 11, the protocol is stacked over the SCTP transport protocol and used by the UE to maintain a direct signaling relationship with the MME. This protocol offers the following services between the UE and the MME [52] :

- functions for the UE mobility support.
- functions for the support of session management procedures for the establishment and maintenance of IP connectivity between the UE and a P-GW.
- security support for integrity protection and ciphering of NAS signaling messages. In order to offer NAS security support, two procedures are specified [52]
  - procedure for EPS Mobility Management (EMM) and,
  - procedure for EPS Session Management (ESM).

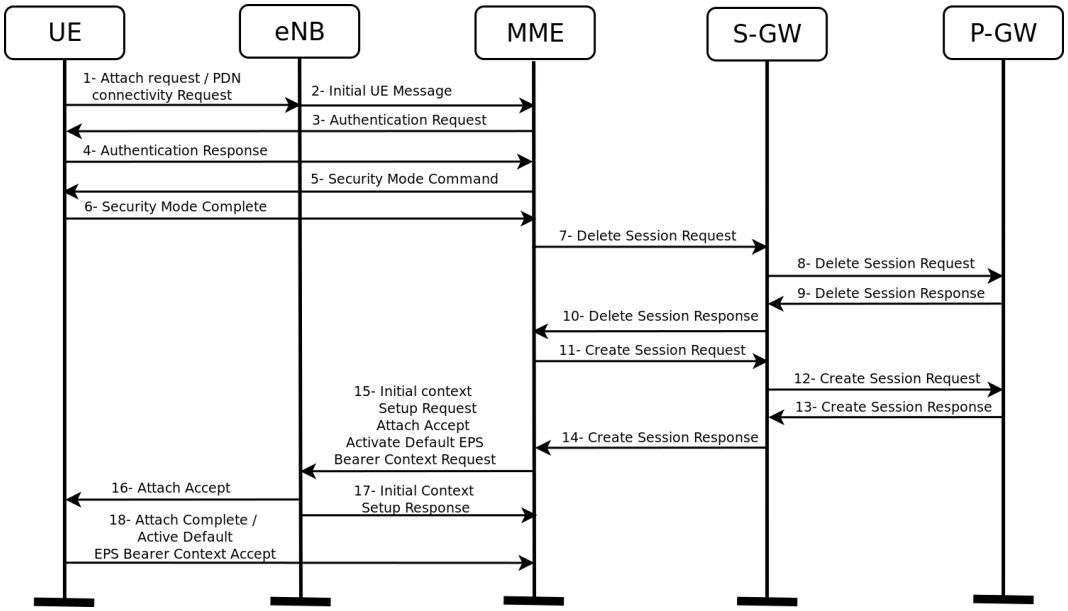


Figure 15: The UE Attach Request Procedure. [41]

Since the NAS protocol is used as the signaling protocol between the UE and the EPC, it implies that it will only become effective during the UE's PDN connectivity and Attach request procedure. This is the reason the UE Attach request is also called the UE NAS-attach request. This procedure normally starts with both the Attach Request and PDN connectivity Request coming from the UE to the MME. The MME in turn replies the UE as it for authentication. On success, the UE authenticates its identity using the authentication response message. Then, the MME initiates

the security mode command towards the UE and on success the UE replies with security mode complete. Then, from 7. to 14. the MME handles the session creation creation request with the rest of the EPC components until a create session response is received from the S-GW. Then, the MME sets up the initial context for the UE and activates the default EPS bearer. This message is sent to the eNodeB which will in turn forward it to the UE as attach accept. On the receipt of the initial context setup request by the eNodeB, the eNodeB will then reply the MME with an initial context response message on success. Finally the UE will then reply the an attach complete message in 18.

3.4 Existing virtualization solutions for EPS

3.4.1 Aalto core network

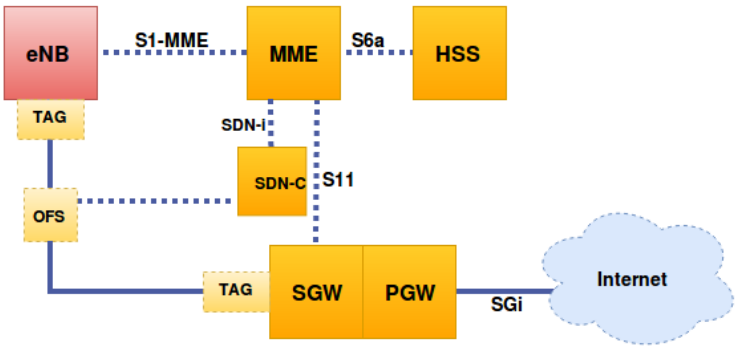


Figure 16: Aalto Implementation Setup.

The Aalto EPC network which was implemented within the framework of the CELTIC-plus project SIGMONA took the design and implementation of the core network to a whole new level. Having implemented the EPC in line with the legacy 3GPP standard specifications, the core network was also designed to leverage the SDN (Softward Defined Networking) technology as a means to address the service provisioning and optimization requirements as well as to demonstrate cost reduction[48].

The core network elements are deployed on commodity servers running as network functions on the cloud and the MME was designed in way that it supports communication with an OpenFlow protocol enabled SDN controller in the backhaul part of the EPS network. This is done in order to further simplify packet forwarding and enable data plane programmability. Integrating SDN controlling capability into the core network by interfacing the MME with an SDN controller which will directly interact with streams of packets coming from the eNodeB.

The SDN controller's job is simplified through the integration of a tagging component into both the Nokia provided eNodeBs and the S-GW. This tagging component is responsible for popping out the GTP headers of packets which are traditionally supported by the standard specifications of both the eNodeBs and the S-GW and are coming from the eNodeB, and pushing into the packets, layers 2

VLAN or MPLS headers which are understood by the OpenFlow protocol enabled switches towards the S-GW and vice versa as illustrated in figure 16. This is done also to allow load balancing and traffic engineering on backhauled packets which are forwarded between the eNodeB and the S-GW based on either the MPLS or VLAN identifiers[48].

3.4.2 Open Air Interface EPS solutions

the OpenAirInterface Software Alliance (OSA) popularly called the OpenAirInterface (OAI). This project is an open source project, developed and sponsored with the sole aim of softwarizing mobile network functions from the access network to the evolved packet core of the mobile network. It is a project which is aggressively supported and developed by a agile community of professional software developers both in academia and telecommunication industry. The OAI project is especially centered around virtualizing the virtualizable part of the access network and the entire EPC of the mobile network, so that these entities can respectively be deployed on virtual platforms. This is done in a bid to further reduce the cost of deployment of the mobile network and to make the role out of newer network functions faster and easier.

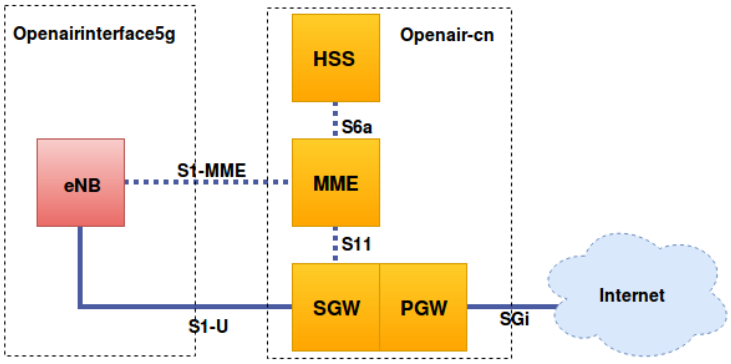


Figure 17: OpenAirInterface Implementation Setup.

The OAI's project is very timely in its development, in that it came to life just around the same time there is need for virtualization solutions for network functions which are normally integrated and tightly coupled to the hardware entities upon which they run as physical network functions. The need to virtualize network functions together with the proposed 5G technology and its all-encompassing potentials in solving most of the challenges faced by the current 4G technology is an additional motivation for developing such a mobile network softwarization solution. As it is no longer new, that the current 4G technology suffers greatly in meeting the quality of service requirements of different verticals and over-the-top players whose service demands are met using a single one-fits-all solution of the 4G technology. In solving this 4G technology deficiency and other accompanying sub-problems, the 5G technology looks very promising in particular with the introduction of network slicing concept[12].

Network slicing is the creation, instantiation and deployment of virtual instances of a complete mobile network from the access to the core network part, tailored towards a specific network service requirement or set of similar business demands. In a nutshell, it involves the logical partitioning of an E2E mobile network, stitching together the necessary chain of virtual network functions from the access network to the core network in order to deliver a complete functional E2E mobile network deployable on virtual platforms. The virtual platforms could be virtual machines running on commodity servers locally or far away on cloud datacenters. In light of supporting the 5G technology enabled through network slicing, the OAI project pioneers a holistic solution for a complete fully fledged virtual mobile network solution. This project is broadly divided into two parts, namely, the Openairinterface5g and the Openair-cn as shown in figure 17.

3.5 Related works on mobile network slicing

Already, there are a number of works that have proven the potential derivable advantages of RAN slicing amongst multiple tenants sharing network resources. Using the so called Multi-Operator Resource Allocation (MORA) criterion to mathematically prove the potential gains and cost saving benefits of utilizing a dynamic resource allocation amongst different mobile operators sharing a common RAN [5]. Another project [6] presented the idea of RAN Multi-tenant cell Slicing Controller (RMSC) which addresses these requirements using two different design approaches, a fully distributed and a fully centralized RMSC. Similarly, slicing solution based on spectrum sharing was described in another paper called Network Virtualization Substrate which focused more on sharing the resources of the RAN only in an LTE network [7]. It implemented a slice scheduler which could provide both Resource-based provisioning and Bandwidth-based provisioning slicing solutions.

A heuristic-based admission control mechanism was also developed in another work focusing on maximizing slice user's satisfaction based on RAN slice prioritization [8]. Also, there is the concept of RadioVisor [9] with a focus on the isolation of radio resources. Their architecture includes three main components, the device and application to slice mapping, the radio resource allocation and finally the isolation function and slice manager. Another project which was evaluated is the Programmable RAN (PRAN) [10], which proposed the design of a virtualization solution in form of a programmable RAN where the L1/L2 processing functionalities and scheduling tasks are moved to run on commodity servers. Another study [11], the Virtual Prioritized Slice (VPS) examined the classification of slices into two major categories based on their delay tolerance levels. These slices are classified into Realtime (RT) and Non-Realtime traffic slices which is done for all the service providers at the network scheduler before allocating network resources to the slices with a focus on scheduling the physical resource blocks of the access network using a proportional fairness algorithm.

Despite the fact that these various related works have proven the potential gains of deploying the present EPS network components in virtual environments, they have done so with focus on individual network parts only, and not considered a complete

slicing of the entire network end to end. In contrast to these works, we are proposing a system which considers an end-to-end network slicing taking into consideration the entire EPS network components, called E2E network slicing.

3.6 Existing network slicing architectures

Obviously, the current "one-fits-all" network architecture is not efficient to support the different needs of 5G services, in terms of latency, bandwidth and reliability, especially because the different service classes are expanding on the direction of one specific requirement (more capacity, more devices connected and low latency) in the detriment of the others. Enabling network slicing in mobile networks, and building network slices tailored to each service, represent one of the solutions towards supporting 5G services. In this context, several 5G initiatives from industry and academia alike have been proposing a new mobile network architecture, featuring network slicing; mainly based on Software Defined Networking (SDN), Network Function Virtualization (NFV) and cloud computing.

In the 5GNorma project [13][14] funded by the European Commission, a new programmable and flexible mobile architecture is proposed. The aim is to enable multi-tenancy over a shared physical infrastructure, and hence network slicing. To this end, the 5GNorma introduces three enabling functional blocks: Software Defined for Mobile networks (SDM)-Orchestrator(O), SDM-Control(C) and SDM-X (Coordinator). SDM-O interfaces the network slice infrastructure to the business domain. The SDM-O handles the slice creation, and translates the slice requirements to network resources in terms of Virtual Network Functions (VNF) and Physical Network Functions (PNF). The SDM-O places and orchestrates the VNFs in the networks, since the SDM-O has a complete view of the network. The resources assigned to a network slice are managed by the SDM-C. The SDM-C builds the forwarding paths used among VNFs and PNFs, while sustaining and managing the constraints and requirements defined by the SDM-O. The SDM-C is also monitoring the slice resources, in case of QoS degradation. The SDM-C is also allowed to request more resources from the SDM-O in a situation where the resources allocated to a slice is not enough to meet its desired level of QoS. While SDM-O is in charge of scaling up and down of the slice specific resources (i.e. compute, storage, transport), the SDN-X is in charge of scaling up and down shared resources among slices (e.g. radio resources).

In [12] 3GPP has provided first analysis of the implications arising from network slicing, without including the RAN. While the RAN remains as a common network segment that includes a new element for slice identification and selection, the Core Network (CN) slicing will be based on the eDECOR [16] model. In the latter the CN instances (slices) are connected to the shared RAN using the classical S1 interfaces. The eNodeB is able to steer the slice traffic to the correct CN instances using the slice ID communicated by the UE during the Radio Resource Control (RRC) procedure. The slice ID could be hard encoded in the UE (i.e. USIM) or encoded through the Public Land Mobile Network (PLMN). In case the devices come without a slice ID, a redirect mechanism was added to the 4G Evolved Packet Core architecture [15]

in which a first MME makes the authentication and authorization of the UE and based on the authorization redirects the request to the appropriate MME for further operations.

Therefore, a mobile network slice is basically made up of VNFs of a CN and a RAN connected together to form a fully functioning substrate of a mobile network enabled by the NFV technology. The ETSI NFV architecture has not only introduced a dynamic construct of an NF forwarding graph which aids a flexible NF deployment and dynamic network operation, it has also standardized the concept by defining three major operational domains, namely [17]: (1) VNFs - this operational domain covers the network functions implemented in softwares deployed to run on virtual platforms with an underlying physical infrastructure, (2) NFV Infrastructure - this domain consists of both the physical and virtualized network resources on which the VNFs are running, (3) NFV Management and Orchestration - this operational domain is responsible for the management of all virtualization-specific functions in the NFV architecture, from the life-cycle management of the VNFs to the orchestration and management of the network resources.

Assuming that a network slice is a composition of physical and virtual resources, which might be instantiated over multiple domains, the 5GEx funded EU project [19] has proposed a new architecture extending the concept of ETSI NFV architecture to cover multiple domains. The new architecture is composed of three layers: Resource domain, single domain resource and multi-domain resource. The resource domain represents the lower layer of the architecture. It exposes domain resources to the single domain orchestration layer via specific interfaces. According to 5GEx, a domain may refer to a technological domain or operator domain. The single domain orchestration layer, the middle layer, includes the domain specific orchestrator, which performs resource and service orchestration of a specific domain using the interfaces exposed by the domain resource layer. Domain specific orchestrator are using interfaces to communicate and coordinate. The top layer of the architecture is the multi-domain orchestration, which includes the multi-domain orchestrator. Each multi-domain orchestrator is connected with one or multiple single-domain orchestrator, and managed by the Orchestrator Admin Domain using business-to-business (b2b) interface. Moreover, the multi-domain orchestrator are connected to other multi-domain orchestrators using the same b2b interface. Finally, the Multi-domain orchestrator exposes a customer-to-business (c2b) interface to consumers.

4 Network slicing across multiple cloud domains

In order to actualize network slicing across multiple domains which is our ultimate future goal, the architecture shown in figure 18 is proposed. The architecture introduces and describes a global orchestrator which is called a multi-domain slice orchestrator. This orchestrator interfaces with each domain specific orchestrator in order to orchestrate and manage the life-cycle of the resources needed from each of the respective domains. The resources are seen separated per technology domains depending on the technology type (Figure 18):

- Virtualization Infrastructure (VI) is consisting of all the nodes which are offering compute and storage resources as well as the networking to interconnect these resources. Examples of VI include data centres and edge compute units.
- Radio - represents the radio resources in terms of spectrum and allocable spectrum areas and the allocation of the communication channels within the spectrum.
- Transport - consists of the networks which interconnect the VIs and the radio resources within the same or in different administrative domains.

Each of the technology domains has its own Resource Orchestration (RO). We argue this choice by the different structures of the resources, which from the perspective of their respective specifications, should be handled by separate ROs, each specific to a technology domain. It is foreseen that for example, for the transport network, an SDN WAN controller will act as the RO while for a data centre, a typical VI Manager (VIM) (e.g., OpenStack) from the perspective of ETSI Management and Orchestration (MANO) may be used. To be able to use the resources in an appropriate manner across the administrative domain, an administrative domain resource orchestrator (RO), named Aggregation RO is considered on top of the technology specific ROs. The Aggregation RO will aggregate all the resources into the same RO, through this operation, the network resources will become transparent to the domain-specific slice orchestrator. This aggregator RO can be seen as a hierarchical type of resource aggregation, mainly for efficient deployment of the system. For example, in Figure 18, an Aggregation RO can be considered for the radio technology domain across the different technologies and spectrum used. Please note that the resources of the different administrative domains may be interleaved as in the case where one domain handles the data centers located on the connectivity path while other administrative domains handle the connectivity between the data centers.

Another reason for introducing the additional Aggregation RO is to have an overview of all the resources inside an administrative domain in order to place VNFs and create related NFFG (Network Function Forwarding Graph) across different resources (including multiple data centers, transport, different wireless accesses) efficiently. The underlying reasons for this architecture are manifold. Primarily, with the global view of all the resources inside an administrative domain, the global RO can best place VNFs with optimum usage of the underlying resources according to

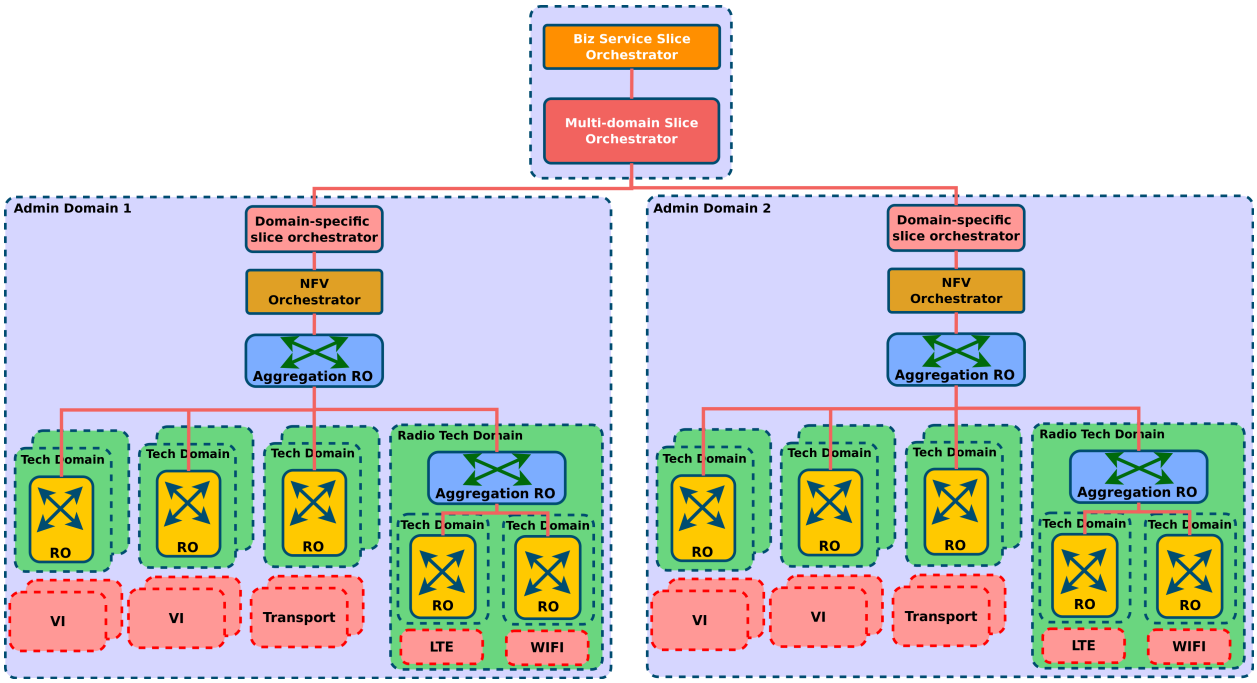


Figure 18: Recursive resource orchestration.

VNF's requirement. Such requirement could for example be affiliation between NFs and special hardware requirements.

Besides, inside an administrative domain, the environment could be multi-technology as well as multi-vendor, but a network slice orchestration would always follow exactly the same set of procedure. Such recursive orchestration procedure enables clear separation of each domain's responsibilities, facilitates reliability and manages scalability within the administrative domain. It also enables the enforcement of different policies in each domain. The domain specific slice orchestrator is in charge of end-to-end orchestration with the interaction of the global RO. In this case, the domain specific slice orchestrator is similar to NFV Orchestrator (NFVO) defined by ETSI MANO but with additional functions required to interact with the multi-domain slice orchestrator. For example, the northbound APIs for communicating with the multi-domain slice orchestrator. The entity which is handling the end-to-end orchestration has to be able to collect and transmit the contact points which enable the interconnection with other administrative domains. The information may include IP addresses within the technology domains use in connecting different technologies from different administrative domains, IP addresses used to bound virtual networks from one domain to another, technologies of the virtual networks binding the two domains and VNF level IP addresses. In case the slice network is explicitly distributed across the domains (IP addresses are allocated in the slice based on a common addressing schema which is done through the orchestration as in the case of any PaaS or SaaS and not for IaaS where only resources are allocated and the tenant has to create the network through its own administrative means) the VNF level IP addresses can still be collected and transmitted. To be able to broker and to bind resources within multiple administrative domains, a multi-domain slice orchestrator is introduced

into the system. The multi-domain slice orchestrator is communicating with the orchestrator in the administrative domains to be able to stitch a slice across the multiple administrative domains, by using resources allocated in each of them. A business service slice orchestrator is added in the logical multi-domain management to be able to interact with the administrator of the slice in the management of the life-cycle operations as well as to offer the administrative entry points to the software elements from which the slice is composed of.

4.1 Multiple domain slicing architecture description

To address the above-mentioned requirements in chapter II, and enable the multi-slice concept, a set of high level architectural reference models are proposed as shown in Figures 18 and 19. Depending on the perspective towards the system, the orchestration architecture, whereby the specifics of the service deployed in the multi-slice architecture are transparent (i.e., it can work for any type of slice) and the slice architecture, wherein the wholistic functionality within the slice template is detailed towards the appropriate functionalities for each of the features are presented. The two solutions are detailed in the following subsections, including the functional definition of the network elements.

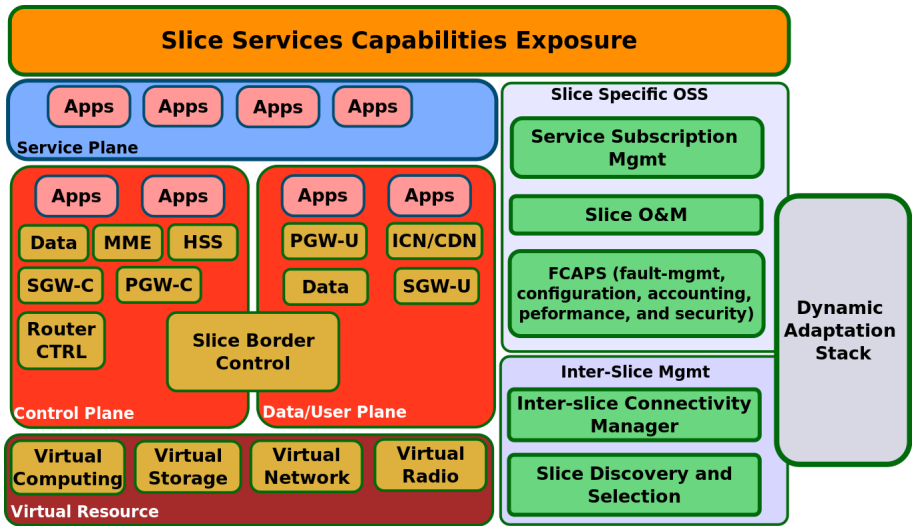


Figure 19: Slice high-level architecture.

The slice architecture includes all the components that compose the network of virtual network functions within the network slice. The proposed slice architecture is illustrated in Figure 19 and follows the slice template model described in chapter 5.1. From the perspective of the slice administrator, the slice represents a complete virtual network, thus, it is a transpose of the current physical network towards the virtual environment. However, the slice architecture has to account for the underlying differences when compared to a physical infrastructure. Therefore, new functional features are added to them in order to form the basis for new benefits in running fully softwarized networks. This type of network slice architecture will perfectly support the idea propagated by the ANYthing As A Service (ANYaaS) [20] concept

whereby a service orchestrator which is capable of delivering services such as dynamic video caching, traffic offloading, light-weight machine type communication EPC on-demand all at the same time is discussed. The slice is running on top of virtual resources (virtual computing, virtual storage, virtual networking and virtual radio) which are acquired on demand through the orchestration architecture. Different from the physical architecture, the resources are varying in time and in place, thus making the slice flexible in terms of capacity and deployment needs. The deployment of a completely virtualized mobile network component (for example, the evolved packet core (EPC) as a service in the cloud has already been evaluated and different deployment options proposed in [21] and a light-weight EPC for MTC in [22]) is no longer new, but developing a high-level template architecture which has all of the VNFs to support any of the slice use-case groups at any particular point in time to enable the deployment of a complete network slice in our opinion is state of the art. For the data/user plane within the slice, a set of components are considered including:

- Data storage and processing components,
- Data plane components related to the connectivity (e.g., Serving Gateway User Plane - SGW-U, Packet Data Network Gateway User Plane - PGW-U),
- Data plane components related to the content routing and storage (e.g., Information-Centric Networking - ICN, Content Delivery Network - CDN), and
- Deep data plane programmed components.

With these, the slice accounts for the possibility to carry out processing of the data directly at the data path, which is mainly possible due to the virtualization of the resources. Moreso, the fact that the slice does not require a separation of the work-flow towards other Apps in the service plane as in the current architecture is an additional benefit. For the control plane within the slice, a set of additional components are considered, providing the connectivity and data control for the specific data plane. It includes functionality for:

- Control of data storage and processing components,
- Control of connectivity related components (HSS, MME, SGW-C, PGW-C),
- Control of forwarding plane (routing and forwarding control), and
- Control of the apps deployed at the data plane level.

In the service plane, a set of Apps (i.e. Application Servers in 3GPP terminology) are deployed enabling the specific service deployment. For managing the slice (this including all the layers of the service), a set of components have to be deployed:

- Slice O&M [23] - the slice operations and management have as main functionality the installation of the specific slice functionality and its maintenance. For the installation related operations, the slice O&M should be able to request

on-demand the addition of a new network function to a running slice (e.g. the addition of a new firewall) in case it is needed based on the information available in the service catalogue and by addressing a momentary communication need. For the maintenance part, the system has to be able to support continuous integration and replacement of network functions with one that offers better functionality (resulting probably in more complex function descriptors) or an entirely new version. Within the NFV environment, a large part of the O&M can be automated under the supervision of the slice administrator. Additionally, O&M is strictly related to the service, hence, less of the operations are generic. For this reason, the slice O&M cannot be centralized in the MANO stack.

- Slice FCAPS (Fault Management, Configuration, Accounting, Performance and Security)[24] – the FCAPS represents the main management functionality of the system. Based on the monitored information coming from the different components and from the infrastructure, the FCAPS system has to provide the appropriate decisions in order to maintain the slice at the appropriate functioning parameters. It includes the following high level functionalities:
  - Fault management - fault monitoring, correction, detection and mitigation actions including failures at the network function level as well as failures in the functioning of the different components.
  - Configuration - including the specific functioning policies and adapted policies which flexibly change depending on the scaling of the service, beyond the simple configurations provided by the VNF Manager (VNFM)
  - Accounting - gathering usage statistics of the slice
  - Performance - gathering network monitored information, making decisions and enforcing them on the components themselves as well as towards the NFVO in order to be able to maintain the expected service level for the users
  - Security - defining the authentication and the encryption mechanism as well as the access control (firewall) rules for the system and adapting them according to the flexibility of the system as well as changing the network topology in case of threats (e.g. pushing towards sandbox networks users which are perceived as possible threats).
- Subscriber configuration management – one specific type of configuration is related to the subscription profiles. Although it is not foreseen that the subscription profiles will be frequently modified during the runtime of the slice, however, two major operating strategies have to be considered:
  - Completing the database information for authentication, authorization and access control rules (i.e. the subscription profile) for all the users at the deployment of the slice; this highly depends on the number of users projected to connect as well as on a possible previous completed database with such subscription profiles.

  - Adding new subscription profiles during runtime on-demand.

Furthermore, it is expected that the slice will be stitched with other external services or with other slices. For this, a set of inter-slice network functions are considered in order to be able to properly interconnect the network slices. The functionality includes:

- Inter-slice management functionality – enabling the peering between the different slices. The inter-slice management functionality has the following functionalities:
  - Slice discovery and selection – based on the information received from the tenant during the deployment, this functionality enables the discovery of the peering slices to connect to. Note, this is a service level stitching between running slices, complimentary to the slice deployment on top of multiple domains.
  - Inter-slice connectivity management - provides the peering between the different slices for exchanging information on the contact points of the slices as well as on the protocol stack (including encryption) for the connectivity between the contact points. If any other connectivity related policies have to be exchanged (rate limiting, availability, etc.), they will also be exchanged over this interface.
- Slice Border Control (SBC) – the SBC functions at the control and data plane levels enables the peering of the control and data plane layers between the different slices.
  - SBC-Control - ensures the interconnection at protocol level between the different components within the slices. It may include for Diameter peering a Diameter Router Agent (DRA) and for IMS communication a Session Border Controller, both with the role of peering with the foreign domain, appropriately routing the requests to the other domain, as well as the anonymization of the private slice information and the encryption of the communication.
  - SBC-User - ensures the proper interworking between the data path components in case the communication requires other protocols than IP only. The functionality may include GPRS Tunneling Protocol (GTP) peering (as in the case of packet core roaming), SFC (Service Function Chaining) peering, multimedia transcoding, and content compression.

Similar to the slice orchestration, there are several functions where a dynamic adaptation may be considered, beyond the current management system. This addresses the following management operations:

- Inter-slice connectivity management policies – can be adapted depending on the momentary network function placement e.g. if functions of two components of the different slices are co-located, it could be better to establish between

them a connection compared to components which are located in different data centres.

- FCAPS operations – FCAPS functionality is the main beneficiary of the dynamic adaptation stack which offers a large amount of possible adaptation actions. This would be a comprehensive extension of the current FCAPS functionality deployed for legacy physical system towards complex events processing and towards adaptations which are possible only in the NFV environment. These functionalities could be for instance, actions for re-creating the network on components' failure, configurations depending on the dynamic network as established by the NFVO during the runtime, differentiated accounting systems depending on services, time of day, etc. It could also involve tasks to enhance the performance and security optimizations of the system through adaptation of functions such as deployment of more appropriate VNFs to a momentary situation, reconfiguration of the components depending on a momentary topology of the system for increasing the resilience and the availability, ensuring of the service Key Performance Indicators (KPIs) across deployments on top of heterogeneous infrastructures to the environment.

- Slice O&M – bringing new components into operation in an already running slice including the dynamic deployment of network slices for continuous integration and automation of the maintenance operations. Also the auditing of the components' performance based on the event log and the adaptation of the running policies according to any detected anomalies.

4.2 Multiple domain slicing orchestration architecture

The orchestration architecture represents the perspective on the system from the multi-slice system management side. The main functionality is related to the life-cycle management of the slice and less to the slice functionality itself, thus being the same, no matter the deployed slice type and regardless of the domain in which the slice is deployed.
A set of existing functions from the NFV environment as well as from dynamic adaptation stack are included in the system. In the following, they are described together with the other new components introduced into the system, making references towards existing specifications when needed.

4.2.1 NFVI

The Network Function Virtualization Infrastructure (NFVI) as seen from the perspective of the slice management, there are no modifications to the NFVI compared to the existing infrastructure proposed in the high level ETSI NFV architecture. However, a specific implementation of the virtual network is considered covering deep data plane programmability and inter-data centre WANs.

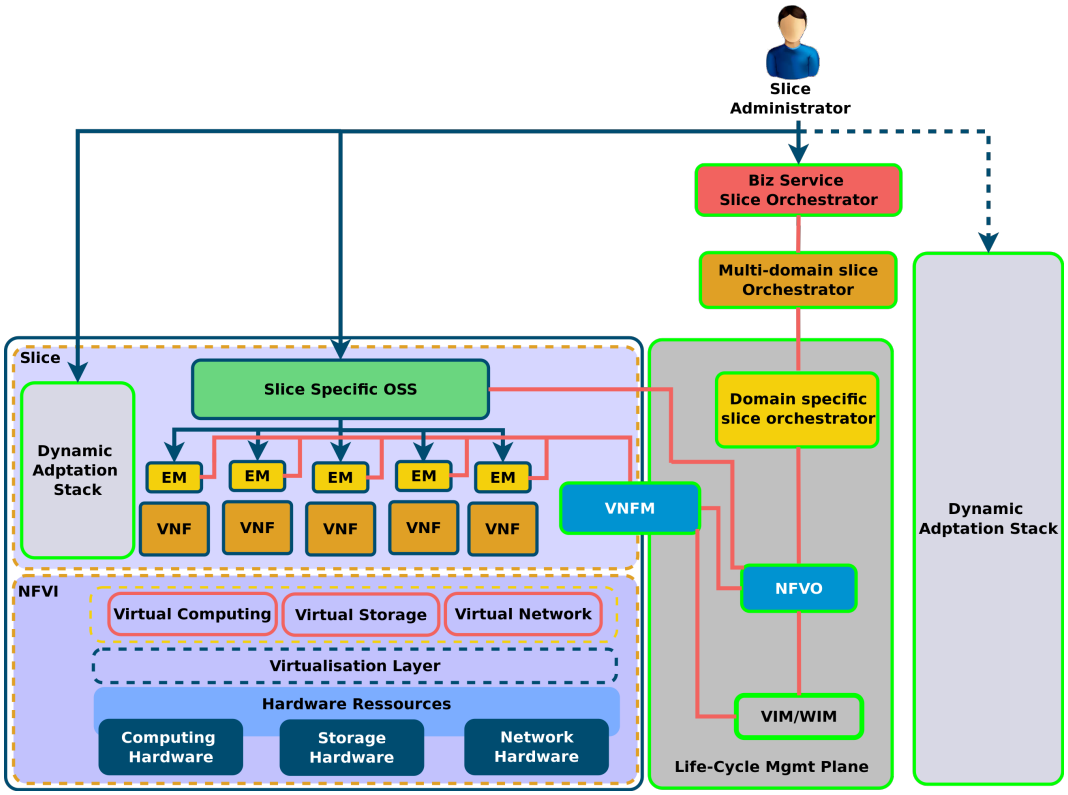


Figure 20: Orchestration architecture.

4.2.2 VIM/WIM

The Virtual Infrastructure Manager (VIM) is defined in the ETSI NFV architecture. Additional functionality of the VIM includes the capability to control the user/data plane functionality such as in the form of an SDN controller or an ICN or CDN information and content control in order to be able to provide a separation of the data plane when the data traffic is directly routed through the network (i.e. deep data plane programmability). The Wide Area Network Infrastructure Manager (WIM) has the role to define the virtual networks between different parts of a slice on top of common transport networks (i.e. the inter-data centre environment sharing rules).

4.2.3 NFVO

The NFVO has the functionality defined by ETSI NFV with its two roles of:

- Resource Orchestrator (RO) enables the brokering of the NFVI resources between the multiple parallel slices. The NFVO represents an aggregation point for the administrative domain for resources management. The NFVO communicates with multiple VIMs and WIMs and is able to allocate the resources appropriately across them.
- Network Service Orchestrator (NSO) provides indications on how the system should scale and where the network functions should be placed following the Network Service Descriptor (NSD) information.

- Additionally, the NSO is extended to support additional commands which may result in the dynamic changing of the NSD information. By this, the active service can be dynamically modified during runtime with additional actions compared to the static NSD based decisions. For example, with this new functionality, new VNFs can be added during runtime to a running system (e.g. a more resilient firewall in case of a network attack).

4.2.4 VNFM

The VNF Manager (VNFM) has the role defined by ETSI MANO specification to:

- Allocate resources to the VNFs appropriately or to delegate this operation to the NFVO,
- To receive events on the completion of the specific operations and information on the dynamic configurations, and
- To configure through the Element Management (EM) the VNFs with the dynamic configurations similar to the operating notion shared in [25]

4.2.5 Domain Specific Slice Orchestration

The domain specific slice orchestrator is able to communicate information on the specific slice split between the different NFVOs with multiple NFVOs located in the same domain. Additionally, the domain specific slice orchestrator receives information on the life-cycle management of part of the slice which is allocated to run on the specific domain from the multi-domain slice orchestrator. Note that this functionality is already offered by the northbound API of the NFVO in the form of the processing of NSDs. It shall be noted that in the envisioned architecture (Figure 20), a domain-specific slice orchestrator and NFVO may be the same entity.

4.2.6 Multi-domain slice orchestration

The multi-domain slice orchestrator has as main role to provide a slice on top of multiple administrative domains. It contains the following functionalities:

- Receive requirements from the business service slice orchestrator on the requirements for the specific slice. The requirements may be received in a static description form such as TOSCA or an NSD file.
- Establish secure connections to the multiple domain specific slice orchestrators
- Acquire, if permissible, knowledge on the available resources in the specific administrative domains in terms of available infrastructure and available services (e.g. stored virtual machine images)
- Negotiate with the domain-specific slice orchestrators the resources and their locations to be allocated for a slice customer

- Make decisions based on the requirements received on the split of the slice functionality across the multiple administrative domains
- Command the installation of the slice over the multiple administrative domains
- When the installation is successful, exchange connectivity parameters between the different domain specific orchestrators to be able to stitch together the slice
- Announce the tenant through the business slice orchestrator on the successful installation of the slice as well as on the connectivity and management points
- Inform the tenant, through the business slice orchestrator and/or the slice-specific OSS, of any SLA breaches or any other types of major failures of the deployed slice

4.2.7 Business Service Slice Orchestration

The business service slice orchestrator has the role of a portal to advertise the possible services, to trigger their deployment and in case of success, to transmit to the slice administrator the specific entry points to the new slice management.

4.2.8 Dynamic adaptation stack (for the life-cycle management plane)

The life-cycle management plane has multiple points in which and through specific policies, the functionality of the system may be adapted. Based on the monitored information from the slice, the NFVI and the life-cycle management components, and the dynamic adaptation stack can provide the following adaptations:

- VIM level - migration of virtual machines, fault management and mitigation at the VM levels, configuration of the infrastructure, infrastructure security protection, authentication and authorization, resources scheduling for performance and resilience for example using such technique proposed in [26];
- WIM level - establishment of new data paths, traffic steering between multiple data paths, QoS classification and differentiation, application differentiation through deep data plane programmability;
- NFVO level - network functions placement in the domain, scaling policies, automatic fault management, resilience and security through application independent mechanisms, modifying the policies in selecting domain specific ROs;
- Domain specific slice orchestrator - modifying policies in selecting NFVOs
- Multi-domain slice orchestrator - modifying the policies in selecting administrative domains, SLA breaching reports;
- Business service slice orchestrator - transmitting to the tenant events in regard to the system on top of which the slice is deployed (i.e. normal behaviour)

4.2.9 Slice Administrator

Using the system, the slice administrator is able to:

- Request a services catalog from the business service slice orchestrator
- Select and configure a slice based on the services provided in the catalog
- Trigger the deployment of the slice according to the configured services
- Administrating the dynamic adaptation stack in both the orchestration and within the slice as much as allowed and possible through policies within the policy engine. Most probably this will be done through pre-defined templates. Administrating the services within the slice through policies within the slice specific OSS as well as through user profiles.

4.2.10 Dynamic Policy Based Management

One of the major advantages of software slices, deployed on top of common infrastructures, is that the system can be dynamically adapted to new network conditions. This includes the adaptation within the slice (i.e. the slice management which is part of every slice due to the fact that flexible resources can adapt the functionality of the system to the most appropriate conditions). Additionally, it includes the adaptation at the life-cycle management (i.e. the life-cycle management can adapt the resources allocated to the specific slices depending on their momentary needs as well as through brokering the available resources). With a physical system, there was not much liberty in terms of events that could happen and not too many actions possible. In NFV, due to the flexible virtual infrastructure used which can scale on-demand and due to the decoupling from the physical infrastructure, new events may be generated. These events could sometimes be highly complex combining information from different metrics of different components. Also, the software system has more possibilities into adaptation including scaling opportunities, network function placement and reconfigurations during runtime for the new network conditions. For this reason, the basic event and logging system which accompanies at this moment the network management stack is not sufficient to optimally operate in a software network environment. With this, new dynamic, policy based management stacks are created for different network functions, enabling them to take appropriate decisions on specific events.

As depicted in Figure 21, the dynamic policy based management stack includes the following classical components, adapted and applied in the new NFV environment:

- Monitored elements - this represents the elements from which the information is gathered, be it part of the service, of the management of the service or as part of the life-cycle management of the service. To reduce the communication needs, the monitored elements may aggregate part of the monitored information.
- Monitoring (server) - the monitoring server receives all the monitored information without any processing or qualification (all information from the monitored

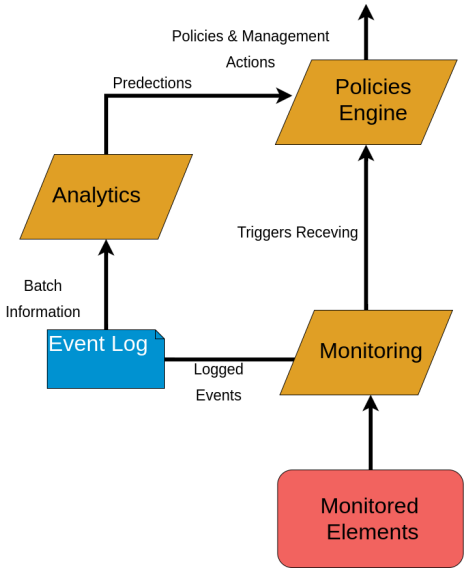


Figure 21: Dynamic policy based management.

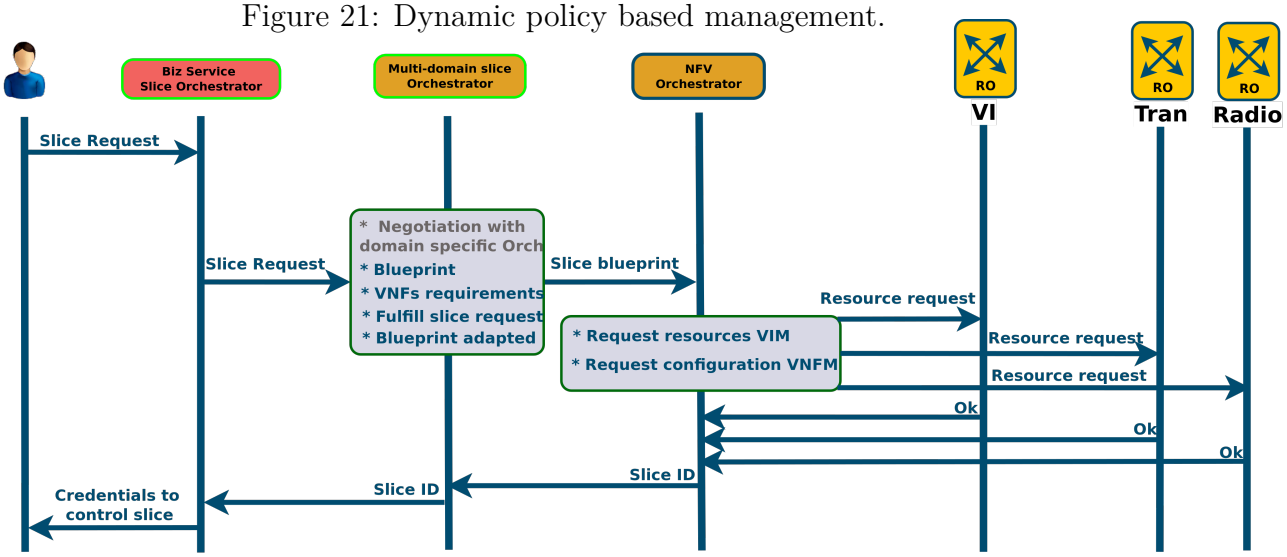


Figure 22: Single domain slice creation via direct interaction with "multi-domain slice orchestrator".

elements is uniform). Based on threshold policies, the monitoring server is either logging the events and raising alarms (as in current management systems) or it provides basic events (e.g. CPU over 90% for a component for 3 times in the last 5 minutes) to the analytics and to the management policy engine.

- The Event Log stores information on the outstanding basic events which are logged from the monitoring server. Alternatively, it can be increased by adding more complex events.

- The Analytics component has the role to generate more complex information from the basic events. Depending on the type of analytics, it may provide different granularity level events such as root cause analysis in case of component failure or even subscriber usage pattern information. Regarding the latter,

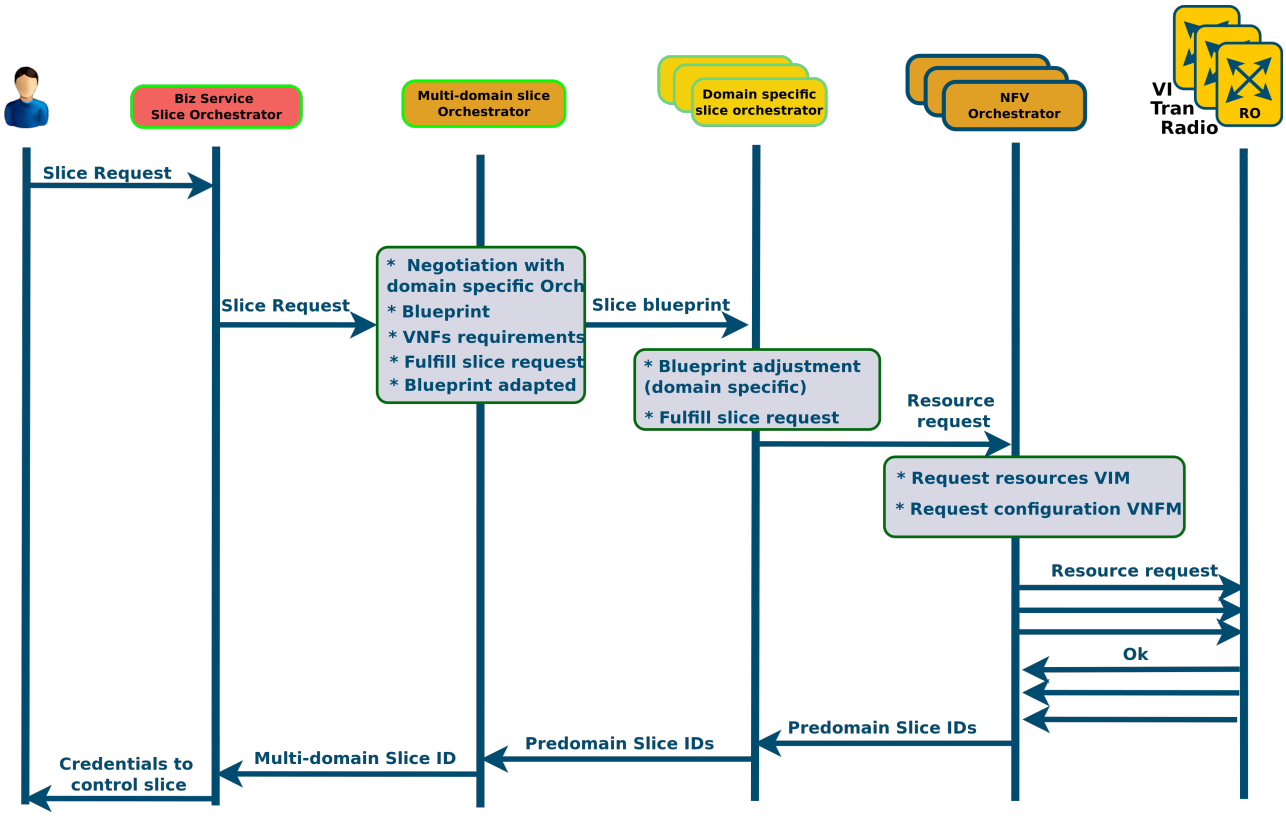


Figure 23: Multi-domain slice creation via direct interaction with "multi-domain slice orchestrator".

per-subscriber monitoring is technically possible through the processing of information available at the core network (i.e., Home Subscriber System - HSS), however, this operation is highly complex and coping with privacy violation may be a challenge. The complex information is transmitted in the form of policy triggers to the policy engine or in the form of new policies to be added to the system.

- The policy engine is the central decision entity of the dynamic adaptation stack. Based on the received triggers from either an analytics engine or directly from the monitoring, it checks the system conditions and based on this, it generates a set of mitigation actions which may result in the modification of the running system.

Additionally, to this system, an event broker may have to be added to the interconnection between the various analytics engines, the monitoring server and the policy engine. The event broker has the role to properly route the events between the different components.

4.2.11 Multi-domain Network Slice Orchestration Procedure

In this section, we will use the architecture as well as the elements defined in the precedent sections to demonstrate the creation of two types of end-to-end slices, single domain and multi-domain slice.

Figure 22 shows the case of creating a single domain slice via the multi-domain slice orchestrator. This would represent the case, where the BSS-O has no information on whether the resources should be created from one domain or more. After receiving a request from the customer, the BSS-O sends a slice creation request to the multi-domain slice orchestrator. The latter uses its blueprint model to build the slice blueprint, which will be communicated to the domain specific slice orchestrator. It is important to note that the multi-domain orchestrator selects the domain to be used for deploying VNFs using a local logic, which takes into consideration the available resources information communicated by the domain specific orchestrator(s), and other information like the geographical area to cover, etc.

Then, the slice blueprint is created. In some cases, after building the slice blueprint, the multi-domain specific slice orchestrator may update its blueprint model according to the information received from the domain specific slice orchestrator. In this use-case, the multi-domain orchestrator selects only one domain for deploying the VNFs. On receiving the slice blueprint, the Domain specific slice orchestrator adjusts the slice blueprint according to its domain specific model. After that, using the updated slice blueprint, the VNFO follows the same steps, as described in the precedent case, to deploy the VNFs.

Figure 23 displays the creation of a multi-domain slice via the multi-domain slice orchestrator. The main differences with the precedent case are:

- The multi-domain orchestrator selects mulit-domain resources to deploy the slice.
- For each domain, a slice blueprint is created. Each one indicates a part of the slice to be deployed. For instance, one domain may deploy only the radio resources, while another domain may instantiate both the virtual infrastructure and the transport network resources. The slice blueprints are sent to each domain specific slice orchestrator in order to be enforced.
- The mutli-domain orchestrator merges the slice IDs, to create a new slide ID along with its credentials, which will be communicated to the customer.

5 Mobile network slicing for enabling 5G networks

To build an end-to-end network slicing, two significant aspects are required. First, we need to carefully define an end-to-end network slice from UE to cloud data centers using programmable resources per application service. This means that there is a need to enable dynamic creation, modification, maintenance and disposing of network slice(s) to serve user's needs from the radio access to the packet core networks. The slice creation technique has to be meticulously planed and coordinated especially across fixed networks and radio boundaries, i.e. the so called mobile packet core slicing and RAN (Radio Access Network) slicing. Each network slice is made up of a virtualized air-interface, radio access network and mobile packet core network, and transport network combined. Second, in order to support various 5G network applications and service requirements, as well as legacy information networking services, a viable slice architecture should manage and operate a large number of slices in a scalable, dynamic and on-demand, and reliable manner. Such kind of slice instantiation, maintenance and termination capabilities would strongly require the establishment of a highly sophisticated distributed processing scheme and "deeply programmable" E2E networking. In what follows, we describe the network slice requirements as dissected and assimilated in the 5G!Pagoda project.

5.1 Main overview of the proposed slice architecture

As stated earlier, a slice offers a dedicated full network system needed to serve an application, similar to what a current network is offering, replicated and customized as best as possible to satisfy the requirement needs of the connected UEs. For this reason, a slice has to include all the functionalities which are currently available in a physical network e.g. a 4G mobile network. Additionally, as the different slice components are implemented in software on top of common hardware resources, a set of optimizations are considered, especially by adding the flexibility and dynamicity made available by SDN and NFV to the system.

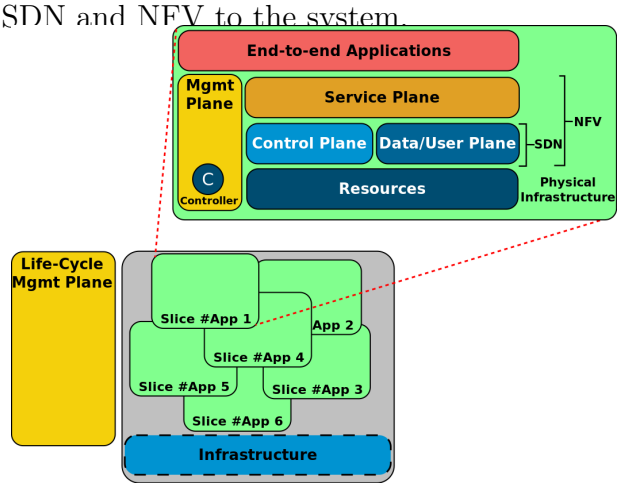


Figure 24: 5G slice template and instantiated slices on top of a common infrastructure.

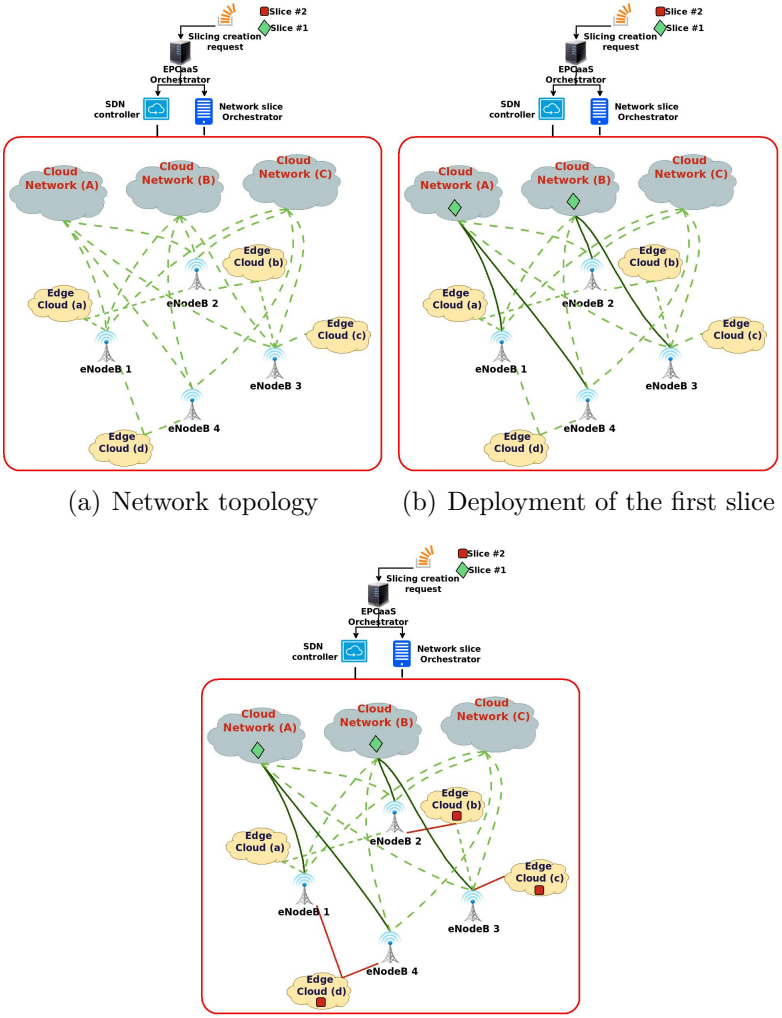
Basically, the underlying high-level functionalities for all the network slices are similar, therefore, the main differences would be in the customization and parametriza-

tion of each slice instance to serve a specific application need efficiently. As a result, all the slices can be implemented following the proposed slice template, as illustrated in Figure 19. On top of a set of common resources, a Data / User Plane is implemented enabling the communication of information between end devices and a network slice, as well as within a network slice. The Data/User Plane is controlled by a separate Control Plane, following the principles of carrier-grade telecom networks enabled through the use of SDN technology whose potential challenges has already been examined and addressed in [27]. Immediately above both the Control and User Plane, there is a Service plane which is established with different application enablers in order to offer the appropriate connectivity service to the specific application(s) using the network slice. Below the End-to-end Applications plane and Next to the Service, Control and Resources planes is the management plane, which controls and manages the appropriate operations of all the other planes and their resources. Considering the latest technological advancements in telecommunication and networking, the control and data/user plane would be implemented following the SDN principles while all the connectivity layers would follow the translation of physical network functions to software modules running on top of a common hardware (generically named softwarization), a principle proposed by ETSI NFV.

A network slice is expected to include all the network components such as a RAN, a transport and a core network, application enablers (e.g., video streaming optimizer) and the applications themselves as well as the management for these technology domains which is necessary to provide a specific service to the end customers. However, in order to optimize the network slice functionality, some of the classical network components may be shared using the current network sharing system (without software customization) and not be included in a network slice. For example, the RAN could be shared between multiple slices thereby making a network slice to only include the rest of the components (e.g., core network components and packet data network components – caches, servers, etc.).

Since multiple slices are deployed on top of virtual resources, there is a need to introduce a new capability to operate multiple slices, as a life-cycle orchestration, this functionality is described in this subsection. As illustrated in Figure 24, different 5G slices are instantiated and are running in parallel and in isolation on top of the same infrastructure. An infrastructure may be operated and managed by single telecom operator or may consist of multiple sub-infrastructures operated by multiple operators and providers (for instance, telecom operators, MVNOs, cloud providers etc.). This allows the deployment of various types of slices, which are deployed on top of different infrastructural configurations made possible mainly due to the fact that the slices are running on virtualized isolated environment dedicated to serve the need of a particular service.

As the resources are virtualized, the slices can receive dynamic resources during their runtime as well as different resources placement, through this, the infrastructure becomes more flexible and available in a different combination on demand and flavours. Bearing the flexibility and dynamicity of the system in mind, the life-cycle orchestration of network slices is not only able to deploy a network slice according to the specific configuration needs of the slice, but also is able to adapt the network slice



(a) Network topology (b) Deployment of the first slice

(c) Deployment of the second slice

Figure 25: Proposed architecture overview

to different usage conditions based on the behaviour of the slice users. In addition, the network slice can evolve to accommodate exceptional network situations based on for instance, the available network resources, unexpected fault detection and management, performance optimization and possible network security compromise. All these functionalities have to be covered by a new set of functional elements (especially, to differentiate from the internal management plane operations which are specific and private to each network slice), generally named life-cycle orchestration in this specification.

5.2 Towards enabling end-to-end mobile network slicing

In this section, we present our suggested framework that will enable the management of EPC as a service (EPCaaS) platform across multi-domains clouds. This will be enabled thanks to E2E network slicing that enables the visualization and slicing of

major network components including RAN resources and core network. As will be mentioned in the following sections, enabling technologies such as Network Function Visualization (NFV) and Software Defined Networking (SDN), will play a crucial rule in enabling the E2E mobile network slicing. While NFV [**?**] technology will enable the elasticity and flexibility for creating different E2E mobile network slicing across multiple domains, the SDN technology will enable the programmability of OpenFlow enabled switches for ensuring the connectivity between different VNFs (eNodeB and Core Networks) of the same network slice.

Fig. 25 shows the main architecture overview of the proposed framework. As depicted in this figure, the EPCaaS orchestrator receives the requests, for creating different E2E mobile network slices, from the network slice administrator. Then, the orchestrator will run a smart algorithm that specifies the Virtual Network Functions which should be created, as well as their locations in the different cloud networks. Moreover, the smart algorithm will enable the orchestrator to specify also the network interconnections between the different VNFs created in variant cloud networks. Those information (VNFs, their locations and connections) would be communicated from the EPCaaS orchestrator to the NFV orchestrator and SDN controller, respectively. The SDN controller will enforce the creation and the interconnection of the different VNFs across multiple domains. The challenges that may accompany E2E mobile network slicing and the requirements to be fulfilled are enormous but a few major ones are described below. These challenges appeared in the two main parts involved in this slicing approach, the RAN and the Core part of the network. For enabling E2E mobile network slicing in the suggested framework, we face a number of challenges as identified and discussed below.

1) Resource Allocation: In order for mobile network operators to maximize their profit, they tend to always keep the wireless frequency channels busy as much as possible, for this practice to continue and for network slice owners to make profit, the network virtualization standard to be adopted in the future $5G$ system architecture should be able to dynamically allocate (schedule) wireless resource blocks seamlessly across slices [1][**?**] using a process otherwise known as scheduling. In allocating wireless resources, a clear definition of what a resource is, well designed mechanism for allocating the resources to the network slices, a robust method to partition the resources, a mechanism to update changes in resource assignment amongst other things has to be carefully considered [2]. As for the EPCaaS orchestrator which will be managing and controlling slice resource allocation in the core network, it is important to design one which will ensure fairness in resource allocation to the requesting network users.

2) Isolation: Isolation is a process of ensuring that resources allocated to a particular network slice does not affect another or that the quality of resources allocated to a slice remains the same over a duration of time and the resources are absolutely for that particular slice [2]. This is particularly difficult to achieve due to the variability of radio frequency channels over a duration of time. In addition, slice isolation in wireless networks is especially challenging also due to the mobility of end users getting services from different slices [1]. Making sure that these factors do not affect resources allocated to other slice should be of utmost importance in the

standardization approach of the potential 5G architecture. More so, the EPCaaS orchestrator residing in the core network should have as part of its functions an updating mechanism which always keeps the current record of the total allocated resources and the available allocatable network resources, so that, the quality of the already orchestrated resources will not be jeopardized in a bid to allocate a set of network resources to incoming request(s). It must ensure that service slices do not experience both inter and intra slice interferences [8]. The slices running on the separate containers on the UE too should not experience data contamination or privacy breaches.

3) Customization: Flows belonging to different slices should be customizable for example, depending on the quality of service offered, slices should be able to determine individual flows' quality of services independent of another slice [1]. There should be a programmable interface provided by the network virtualization solution to be adopted which will enable customization of flows attributes. The level of customization which could be offered is solely dependent on the allowable flexibility available through the virtualization solutions deployed and the service level agreement existing between the infrastructure provider and the network operators sharing the network resources. Ensuring a balance between the customization of slices and slice isolation is another challenge especially when it comes to different provisioning methods such as resource-based and bandwidth-based provisioning as described in [7].

As depicted in Fig. 25, the envisioned EPCaaS orchestrator should offer a RESTful API that allows the slice administrator to specify E2E mobile network slices and their features, such as network latency and bandwidth communication. Moreover, the user can specify different management rules and policies for the instantiation and auto-scaling of different VNF instances created in variant clouds networks. According to the received requests, the EPCaaS orchestrator enforces the rules for a specified slice by communicating them to the virtual infrastructure manager (VIM). In this figure, dashed arrows between EnodeBs, edge clouds and core networks, indicate the network connectivity between them. The length of dashed arrow means the distance of an EnodeB to an edge cloud or a core network. The longer the distance between an EnodeB to a core network is, the higher the latency and the lower the bandwidth becomes.

This figure shows the creation of two E2E mobile network slices. While Fig. 25(b) shows the deployment of the first E2E mobile network slice that will be used for connecting UE variants, Fig. 25(c) shows the deployment of the second slice used for the purpose of connecting IoT devices. As the first slice does not have any special requirements, the different E2E core network VNFs are instantiated in different core networks without any restrictions. Meanwhile, the second slice requires low latency and high bandwidth, for this reason the variant core network VNFs are instantiated close to eNodeBs at the edge clouds. Fig. 26 shows the RAN slicing for connecting the two E2E core network slices dedicated to UEs and IoT devices, respectively.

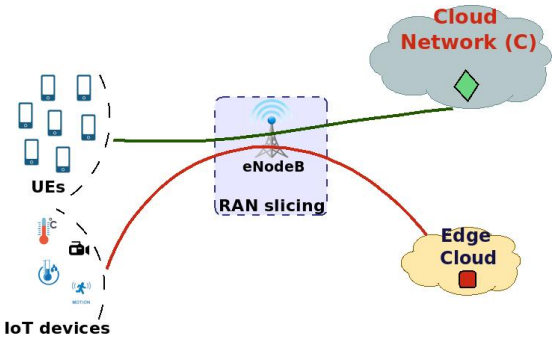


Figure 26: E2E Slicing Scenario

5.3 Implementation of Multi-tenancy in the Radio Access Network

As mentioned earlier in section 4, the final goal of the 5G!pagoda project is to enable network slicing across multiple domains in flexible and dynamic fashion, however, the scope of this thesis work is to bring us a step closer to achieving that goal by enabling network slicing all the way to the RAN as a means for supporting the mobile network evolution towards the 5G technology through multi-tenancy.

As already mentioned, network slicing is a tool to support multi-tenancy and customize network architectures for diverse service scenarios and use cases. The wide range of service scenarios and use cases that the 5G mobile network is expected to support may have conflicting service requirements. Therefore, there is need for mobile network operators who would be providing these different services to different consumers to slice the entire network end-to-end. Slicing the network end-to-end means creating and tailoring a dedicated logically isolated slice of network resources from the access, to the transport and then to the core of the network to efficiently serve a particular 5G use case scenario. Hence, network slicing is meant to share the same physical infrastructure with multiple communication services of tenants with divergent requirements. A network slice can be defined as an end-to-end logically isolated network that includes access, transport and core network functions.

Network slicing can not be end-to-end except the network is sliced from the access part of the network i.e. the radio access network (RAN) all the way to the core part of the network i.e. evolved packet core (EPC) in the context of the 4G mobile. In other words, enabling network slicing capabilities at the RAN is very essential to achieve end-to-end network slicing. To slice the RAN, the deployed radio access technology (RAT) has to offer connection interfaces which will support incoming connections from multiple core network instances simultaneously. Moreover, such interfaces at the RAN should not only support the simultaneous handling of multiple connections of core network instances but should be also able to separate traffic of both the control and user-data planes of each individual connected core network instances. In addition to that, the RAN should be also able to allocate physical radio resource blocks (PRBs) to the UEs which will be connected to the different network slices based on their respective characteristics and the requirements of their received services. The network shall be able to effectively navigate the traffic of end-users,

both data and control planes, from the access through the serving core networks and then to their respective destination packet data networks.

The 4G LTE RAN readily supports the configuration of a connection interface for multiple core networks called the S1-Flex[50]. The configuration of the S1-Flex interface allows for the sharing of a radio access network between multiple core network entities. When an LTE RAN is configured to use the S1-Flex interface, an active form of sharing of the eNodeB is activated and enabled between multiple MMEs belonging to EPC variants which may be owned by different mobile network operators. The MMEs will use their respective classical S1-MME interfaces to communicate with the shared eNodeB. The S1-Flex interface can also be configured for the implementation of load balancing among MMEs belonging to the mobile network operator. In this case, each LTE eNodeB of the network operator is configured to connect to the pool of MMEs under the management of the network operator or communication service provider. However, merely configuring and enabling connection interface such as the S1-Flex for a flexible management of multiple EPC's instances connections is not enough to support network slicing at the RAN. Instead, a full network slicing support at the RAN also entails the configuration of the radio resource scheduler; otherwise known as the MAC scheduler to allocate radio resources to the RAN sharing core network slice instances.

For the limited radio frequency resources of the eNodeBs to be efficiently shared among different network slices, two approaches can be explored: (1) the dedicated and (2) the shared approaches. In the dedicated approach, the resources as well as the processing functional modules of the eNodeBs have to be statically configured and made slice-specific. This means that every network slice would have its own instance of the RAN stack, from the radio resource control (RRC), to the Radio link control (RLC), then to the packet data convergence protocol (PDCP) all the way down to the medium access control (MAC)(RRC/RLC/PDCP/MAC) which will be dedicated to manage a statically allocated percentage of the physical radio resource blocks (PRBs).

This approach will ensure a near-perfect isolation of RAN slice resources for every connected network slices, thereby ensuring both the security of the network slice traffic and the QoS as stipulated in the service level agreement.However, on the other hand, the flexibility needed in handling and managing the RAN resources will be missing as well as slice resource update (scaling) during the lifecycle of the network slices. The fact that the PRBs are statically allocated means that even when they are not in use by the slice owner, they cannot be used by other slices. As a result, this approach may lead to a waste of radio resources thereby leading to inefficiency in network slices resource consumption.

The shared approach, on the other hand, allows the use of shared control plane functions, the MAC scheduler and the PRBs to be scheduled. In this scenario, the MAC scheduler shared the total PRBs amongst all the users of the network slices, perhaps, based on the user traffic priorities. This approach brings the much-needed flexibility and elasticity, in fact, the multiplexing gains to the RAN as a result of network slicing support at the RAN.

5.3.1 Enabling the S1-Flex interface and sharing the physical resource blocks

Collectively, both the software-based mobile network solutions of OAI and Aalto University together provided us with a rich source of experimental platform for the deployment of virtual mobile networks as a means to enable network slicing. We successfully deployed and tested both the cloud RAN network solution and the virtual EPC of the OAI solutions using programmable SIM cards on Commercial of The Shelf (COTS) UEs to connect to the Internet.

Similarly, using the OAI's RAN network solution, we also successfully tested the Aalto University developed virtual EPC solution using programmable SIM cards on commercial UE. However, in a bid to test both virtual EPCs side by side while running concurrently and implement the RAN network slicing solution as shown in figure **??** over the OAI's radio access solution, we had to develop the S1-flex interface on top of the virtual RAN solution. This S1-flex configuration allows a complete separation of the connected variant EPCs on both the control and data planes. This functionality was successfully implemented on top of the virtual RAN solution of the OAI and tests were conducted using both Aalto's and OAI's virtual EPCs by connecting different COTS UEs to different PDN networks through the respective virtual EPCs. By so doing, we successfully sliced the resources of the RAN between two virtual EPCs as shown in section 6.

Bearing in mind that the main functions of an access network is broadly divided into two, namely, the baseband functions and the radio frequency functionalities [51]. In our deployment, while the OAI RAN solution is deployable on virtual platforms in order to carryout the baseband functions, the radio frequency functionalities are accomplished using the software defined radio solution offered by National Instrument's USRP B210 embedded board. However, for better performance of the OAI's virtual RAN solution especially on resource constrained virtual machines, it is recommended to deploy the RAN solution directly on a bare metal personal computer due to real-time processing requirements of the RAN solution.

Our current implementation of RAN slicing is done in conformity with the OAI's RAN coding style and configuration solutions. So similar to theirs, we have also used configuration files on the eNodeB to reflect the connection with multiple MME IP addresses and identities in particular, PLMNs. When the eNodeB is enabled to share its resources between multiple EPCs, the module (multiple source files) which is responsible for parsing the various parameters that are defined in the configuration file is modified to be able to handle multiple EPCs parameters. Also, the module which is responsible for setting up the S1AP (S1 Application protocol) between the eNodeB and MMEs through the S1setupRequest and S1setupResponse procedure in order to handle the calls for multiple MMEs in parallel is also improved. Similarly on the EPC side, we enabled the functions which were designed originally to handle only a unicast Tracking Area Identity (TAI) during the S1setupRequest and S1setupResponse procedure to now handle broadcast TAI. The modules which are affected in our implementation are quite essential and significant in order to allow the eNodeB to adequately navigate the traffics of the respective connected UEs to

```
56341 1434.4850320… 195.148.127.84   195.148.127.240  SCTP    84 INIT
56342 1434.4852138… 195.148.127.84   195.148.127.51   SCTP    84 INIT
56343 1434.4852643… 195.148.127.240  195.148.127.84   SCTP   308 INIT_ACK
56344 1434.4853004… 195.148.127.84   195.148.127.240  SCTP   280 COOKIE_ECHO
56345 1434.4853858… 195.148.127.240  195.148.127.84   SCTP    62 COOKIE_ACK
56346 1434.4856142… 195.148.127.84   195.148.127.240  S1AP   132 id-S1Setup, S1SetupRequest
56347 1434.4856894… 195.148.127.51   195.148.127.84   SCTP   308 INIT_ACK
56348 1434.4857103… 195.148.127.84   195.148.127.51   SCTP   280 COOKIE_ECHO
56349 1434.4857844… 195.148.127.240  195.148.127.84   SCTP    64 SACK
56350 1434.4858315… 195.148.127.51   195.148.127.84   SCTP    62 COOKIE_ACK
56351 1434.4859990… 195.148.127.84   195.148.127.51   S1AP   132 id-S1Setup, S1SetupRequest
56352 1434.4860785… 195.148.127.51   195.148.127.84   SCTP    64 SACK
56353 1434.4862703… 195.148.127.240  195.148.127.84   S1AP    92 id-S1Setup, S1SetupResponse
56354 1434.4862954… 195.148.127.84   195.148.127.240  SCTP    64 SACK
56355 1434.4870398… 195.148.127.51   195.148.127.84   S1AP    92 id-S1Setup, S1SetupResponse
56356 1434.4870708… 195.148.127.84   195.148.127.51   SCTP    64 SACK
```

Figure 27: S1Setup Request and S1Setup Response

```
59363 1501.2… 195.148.127.84  195.148.127.51   S1AP/NAS-EPS   212 id-initialUEMessage, Attach request, PDN connectivity request
59370 1501.2… 195.148.127.51  195.148.127.84   S1AP/NAS-EPS   144 SACK id-downlinkNASTransport, Authentication request
59417 1501.4… 195.148.127.84  195.148.127.51   SCTP            64 SACK
59421 1501.5… 195.148.127.84  195.148.127.51   S1AP/NAS-EPS   124 id-uplinkNASTransport, Authentication response
59422 1501.5… 195.148.127.51  195.148.127.84   S1AP/NAS-EPS   124 SACK id-downlinkNASTransport, Security mode command
59423 1501.5… 195.148.127.84  195.148.127.51   S1AP/NAS-EPS   136 SACK id-uplinkNASTransport, Security mode complete
59447 1501.5… 195.148.127.51  195.148.127.84   S1AP/NAS-EPS   280 SACK id-InitialContextSetup, InitialContextSetupRequest , Attach accept, Activate default EPS bearer context request
59487 1501.6… 195.148.127.84  195.148.127.51   S1AP           204 SACK id-UECapabilityInfoIndicationUECapabilityInformation
59532 1501.8… 195.148.127.51  195.148.127.84   SCTP            64 SACK
59533 1501.8… 195.148.127.84  195.148.127.51   S1AP/NAS-EPS   184 id-InitialContextSetup, InitialContextSetupResponse id-uplinkNASTransport, Attach complete, Activate default EPS bearer contex
59589 1502.0… 195.148.127.51  195.148.127.84   SCTP            64 SACK
60845 1528.8… 195.148.127.240 195.148.127.84   SCTP           100 HEARTBEAT
60846 1528.8… 195.148.127.84  195.148.127.240  SCTP           100 HEARTBEAT_ACK
```

Figure 28: UE1 successfully attached

their respective EPCs during and after the PDN connectivity and Attach request procedure for the user plane establishment.

```
89794 1824.0… 195.148.127.84  195.148.127.240  S1AP/NAS-EPS   148 id-initialUEMessage, Attach request, PDN connectivity request
89801 1824.0… 195.148.127.240 195.148.127.84   S1AP/NAS-EPS   144 SACK id-downlinkNASTransport, Authentication request
89841 1824.1… 195.148.127.84  195.148.127.240  S1AP/NAS-EPS   148 SACK id-uplinkNASTransport, Authentication failure (Synch failure)
89842 1824.1… 195.148.127.240 195.148.127.84   S1AP/NAS-EPS   144 SACK id-downlinkNASTransport, Authentication request
89882 1824.2… 195.148.127.84  195.148.127.240  S1AP/NAS-EPS   140 SACK id-uplinkNASTransport, Authentication response
89883 1824.2… 195.148.127.240 195.148.127.84   S1AP/NAS-EPS   120 SACK id-downlinkNASTransport, Security mode command
89888 1824.2… 195.148.127.84  195.148.127.240  S1AP/NAS-EPS   136 SACK id-uplinkNASTransport, Security mode complete
89915 1824.3… 195.148.127.240 195.148.127.84   S1AP/NAS-EPS   272 SACK id-InitialContextSetup, InitialContextSetupRequest , Attach accept, Activate default EPS bearer context request
89992 1824.3… 195.148.127.84  195.148.127.240  S1AP           132 SACK id-UECapabilityInfoIndicationUECapabilityInformation
90057 1824.5… 195.148.127.240 195.148.127.84   SCTP            64 SACK
90058 1824.5… 195.148.127.84  195.148.127.240  S1AP/NAS-EPS   184 id-InitialContextSetup, InitialContextSetupResponse id-uplinkNASTransport, Attach complete, Activate default EPS bearer conte
90153 1824.7… 195.148.127.240 195.148.127.84   SCTP            64 SACK
92618 1843.6… 195.148.127.51  195.148.127.84   SCTP           100 HEARTBEAT
92619 1843.6… 195.148.127.84  195.148.127.51   SCTP           100 HEARTBEAT_ACK
```

Figure 29: UE2 successfully attached

6 Results and analysis

The results of the implementation are shown using Wireshark to display the captured packets during the setup. Our setup consists of two COTS UE (a smart phone and a Samsung LTE dongle), a Lenovo ThinkPad laptop, two HP laptops and a National Instrument's USRP B210 embedded board. All of the three laptops have 8GB RAM, Intel Core i5 CPU and 500GB HDD installed on them. The OAI's eNodeB with our RAN slicing solution was deployed on the Lenovo laptop while two instances of vEPCs were deployed on virtual machines running on the HP laptop.

All necessary configurations needed to successfully instantiate the vEPCs and eNodeB were done as described on [53] except the part that we have added which is needed by the eNodeB in order to support multiple instances of vEPCs which is provided in the appendix. Figure 27 shows two vEPCs addresses which are successfully connected to the eNodeB upon the S1SetupRequest messages that they both received from it. As shown on the figure, the eNodeB's IP address is 195.148.127.84, while the first and second vEPCs' IP addresses are 195.148.127.240 and 195.148.127.51 respectively. As shown on the figure, both the vEPCs' IP addresses replied with an S1SetupResponse message each to the eNodeB after receiving the request from it. This shows that the eNodeB has successfully setup the S1-MME interfaces in parallel with each of the vEPCs. This further shows that the control plane signaling was successfully established.

Next, we needed to test if the eNodeB can properly navigate the traffic of any UE seeking to connect to the respective vEPCs on both the user and control planes. So, in order to test this, we deployed our first UE (i.e., Samsung LTE dongle). We first inserted a programmed SIM card in the LTE dongle and then plugged it to a laptop with a disabled Wi-Fi interface. The dongle detected our mobile network through the radio signal from the USRP B210 eNodeB radio head and launched an Attach request through its GUI. The result of the request is shown in Figure 28.

Similarly, we inserted a second programmed SIM card whose IMSI reflects the PLMN ID of the second vEPC, in the UE (i.e., a smart phone). The settings of the smart phone was changed so that it connects only to LTE network. Once the phone was turned on, it detected the nearby LTE eNodeB signal and also initiated an Attach request. The result of the request is also shown in Figure 29.

Both results show that the two devices successfully connected to their respective vEPCs and a default EPS bearer was created for them thereby affirming that our RAN slicing solution works.

7 Conclusion

This thesis report discusses the concept of E2E network slicing as an important vision of 5G mobile systems, highlighting its key enabling technologies. RAN slicing is identified as an integral part of this concept. In addition, high level description of the relevant architecture are showcased and how E2E mobile network slicing can be achieved, leveraging the OAI source code and the AALTO CN solution. The E2E mobile network slicing depends principally on the S1-Flex concept, as per 3GPP standards, and the dynamic sharing of RAN resources.

Hence, the main aim and objective of this project was to enable cloud RAN slicing solution through the implementation of the S1-Flex concept and enabling the radio resource block sharing by leveraging the Openairinterface5g RAN solution. This is seen as major leap forward towards enabling the 5G technology requirements. This implementation is part of a larger project known as the 5G!pagoda project which is co-funded between EU and Japan.

We successfully built a testbed using our implementation which proves that slicing the resources of the RAN is very feasible and is worthy of being a major enabler towards achieving the 5G technology diverse requirements.

For an efficient E2E network slicing architecture, a number of challenges are yet to be tackled. This includes determining the optimal amount of physical resource blocks to be assigned to a slice type in the RAN and the total resources (CPU and memory) to be allocated for the orchestration of the EPC VM using an EPCaaS orchestrator as proposed in our architecture[49]. Whilst the introduced framework represents a simplistic, yet important step towards a practical implementation of the E2E network slicing concept, it is the author's hope that the presented work would stimulate further research activities from the relevant community of researchers in developing efficient algorithms and mechanisms that would support such concept in line with the envisioned 5G system architecture.

References

[1] R. Kokku, R. Mahindra, H. Zhang and S. Rangarajan, "NVS", Proceedings of the sixteenth annual international conference on Mobile computing and networking - MobiCom '10, 2010.

[2] M. Richart, J. Baliosian, J. Serrat and J. Gorricho, "Resource Slicing in Virtual Wireless Networks: A Survey", IEEE Transactions on Network and Service Management, pp. 1-1, 2016.

[3] P. Rost, A. Banchs, I. Berberana, M. Breitbach, M. Doll, H. Droste, C. Mannweiler, M. Puente, K. Samdanis and B. Sayadi, "Mobile network architecture evolution toward 5G", IEEE Communications Magazine, vol. 54, no. 5, pp. 84-91, 2016.

[4] I. Afolabi, A. Ksentini, M. Bagaa, T. Taleb, M. Corici and A. Nakao,"Towards 5G Network Slicing over Multiple-Domains", IEICE Transactions on Communications, 2017.

[5] P. Caballero, A. Banchs, G. de Veciana, X. Costa-Perez, "Multi-Tenant Radio Access Network Slicing: Statistical Multiplexing of Spatial Loads", arXiv Networking and Internet Architecture,'07,2016.

[6] P. Garces, X. Perez, K. Samdanis and A. Banchs, "RMSC: A Cell Slicing Controller for Virtualized Multi-Tenant Mobile Networks", 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), 2015.

[7] X. Costa-Perez, J. Swetina, T. Guo, R. Mahindra and S. Rangarajan, "Radio access network virtualization for future mobile carrier networks", IEEE Communications Magazine, vol. 51, no. 7, pp. 27-35, 2013.

[8] M. Jiang, M. Condolusi and T. Mahmoodi, "Network slicing management and prioritization in 5G mobile systems", European wireless 2016; 22 nd European wireless conference, 2016.

[9] A. Gudipati, L. Li and S. Katti, "RadioVisor", Proceedings of the third workshop on Hot topics in software defined networking - HotSDN '14, 2014.

[10] W. Wu, L. Li, A. Panda and S. Shenker, "PRAN", Proceedings of the 13th ACM Workshop on Hot Topics in Networks - HotNets-XIII, 2014.

[11] A. Abdelhamid, P. Krishnamurthy and D. Tipper, "Resource Allocation for Heterogeneous Traffic in LTE Virtual Networks", 2015 16th IEEE International Conference on Mobile Data Management, 2015.

[12] "Study on Architecture for Next Generation System", the 3rd partnership project (3GPP), TR 23.799, version 14.0.0, Dec. 2016.

[13] P. Rost, C. Mannweiler, D. Michalopoulos, C. Sartori, V. Sciancalepore, N. Sastry, O. Holland, S. Tayade, B. Han, D. Bega, D. Aziz and H. Bakker, "Network Slicing to Enable Scalability and Flexibility in 5G Mobile Networks", IEEE Communications Magazine, vol. 55, no. 5, pp. 72-79, 2017.

[14] A. Banchs, M. Breitbach, X. Costa, U. Doetsch, S. Redana, C. Sartori and H. Schotten, "A Novel Radio Multiservice Adaptive Network Architecture for 5G Networks", 2015 IEEE 81st Vehicular Technology Conference (VTC Spring), 2015.

[15] "General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access", the 3rd Generation partnership Project (3GPP), TS 23.401, 14.0.0, 2016-12-16.

[16] “Enhancement of Dedicated Core Networks selection mechanism", 3GPP, TR 23.711, release 14.

[17] ETSI NFV, Network Functions Virtualisation (NFV); Architectural framework , GS NFV 002, Oct. 2013.

[18] Open Networking Foundation, “SDN architecture overview", version 1.0, Dec. 2013.

[19] R. Guerzoni et al. “Multi-domain Orchestration and Management of Software Defined Infrastructures: a Bottom-Up Approach", in Proc. of European Conference on Networks and Communications, 2016, Athens.

[20] T. Taleb, A. Ksentini, and R. Jantti, “Anything as a Service for 5G Mobile Systems”, in IEEE Network Magazine, Vol. 30, No. 6, Dec. 2016

[21] T. Taleb, M. Corici, C. Parada, A. Jamakovic, S. Ruffino, G. Karagiannis, and T. Magedanz, "EASE: EPC as a Service to Ease Mobile Core Network," in IEEE Network Magazine, Vol. 29, No. 2, Mar. 2015. pp.78 – 88.

[22] T. Taleb, A. Ksentini, and A. Kobbane, “Lightweight Mobile Core Networks for Machine Type Communications,” in IEEE Access Magazine, Vol 2, Oct. 2014. pp.1128-1137

[23] IEV operations and maintenance definitions for operations, maintenance support and maintenance, last visited on 29.11.2016, http://www.electropedia.org/

[24] ISO FCAPS standard, last visited on 29.11.2016, http://standards.iso .org/ittf/PubliclyAvailableStandards/s014258_ISO_IEC_7498-4_19 89(E).zip

[25] F.Z. Yousaf and T. Taleb, “Fine Granular Resource-Aware Virtual Network Function Management for 5G Carrier Cloud,” in IEEE Network Magazine, Vol. 30, No. 2, Mar. 2016. pp. 110 – 115.

[26] I. Farris, T. Taleb, A. Iera, H. Flinck, “Lightweight Service Replication for Ultra-Short Latency Applications in Mobile Edge Networks,” in Proc. IEEE ICC 2017, Paris, France, May 2017.

[27] T. Taleb, “Towards Carrier Cloud: Potential, Challenges, & Solutions,” in IEEE Wireless Communications Magazine, Vol. 21, No. 3, Jun. 2014. pp. 80-91.

[28] GENI: Slice-based Federation Architecture specification, groups.geni .net/geni/raw-attachment/wiki/SliceFedArch/SFA2.0.pdf, last visited on 07.03.2017.

[29] “Study on New Services and Markets Technology Enablers", the 3rd partnership project (3GPP), TR 22.891, version 14.2.0, Sep. 2016

[30] “Flare: Open deeply programmable network node architecture," http://netseminar.stanford.edu/seminars/10-18-12.pdf

[31] “5G PPP 5G Architecture," the 5G PPP Architecture working Group, white paper, https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-July-2016.pdf

[32] METIS-II Project, https://5g-ppp.eu/metis-ii/

[33] “Planetlab," http://www.planet-lab.org, 2012

[34] “MGMN 5G WHITE PAPER", NGMN Alliance, white paper, https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1 _0.pdf, Feb. 2015.

[35] “5G Mobile Communications Systems for 2020 and beyond," the 5th Generation Mobile Communications Promotion Forum (5GMF), white paper, http://5gmf.jp/wp/wp-content/uploads/2016/07/5GMF_WP100_Executive_Summary-E.pdf, July 2016.

[36] 3GPP TR 22.863, Feasibility study on new services and markets technology enablers for enhanced mobile broadband; Stage 1, Rel.14, Jun. 2016.

[37] 3GPP TR 22.862, “Feasibility study on new services and markeets technology enablers for critical communications"; Stage 1, Rel.14, Jun. 2016.

[38] 3GPP TS 36.300, “Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRAN); Overall description"; Stage 2, Rel. 13, Jan. 2016.

[39] 3GPP TS 36.420, “LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); X2 general aspects and principles", Rel. 10, Oct. 2011.

[40] 3GPP TS 36.413, “LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); S1 Application Protocol (S1AP)", Rel.12, Sep. 2014.

[41] 3GPP TS 23.401, “LTE; General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access", Rel.8, June 2011.

[42] IETF RFC 4960: "Stream Control Transmission Protocol"

[43] 3GPP TS 29.272, “Universal Mobile Telecommunications System (UMTS); LTE; Evolved Packet System (EPS); Mobility Management Entity (MME) and Serving GPRS Support Node (SGSN) related interfaces based on Diameter protocol" Rel.9, Jan. 2012.

[44] 3GPP TS 29.274, “Universal Mobile Telecommunications System (UMTS); LTE; 3GPP Evolved Packet System (EPS); Evolved General Packet Radio Service (GPRS) Tunnelling Protocol for Control plane (GTPv2-C); Stage 3", Rel.12, Oct. 2014.

[45] 3GPP TS 33.401, “Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; 3GPP System Architecture Evolution (SAE); Security architecture", Rel.10, July 2012.

[46] 3GPP TS 23.402, “Universal Mobile Telecommunications System (UMTS); LTE; Architecture enhancements for non-3GPP accesses ", Rel.10, June 2011.

[47] 3GPP TS 33.401, “Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Policy and Charging architecture", Rel.13, Mar. 2016.

[48] J. Costa-Requena, J. Santos, V. Guasch, K. Ahokas, G. Premsankar, S. Luukkainen, O. Perez, M. Itzazelaia, I. Ahmad, M. Liyanage, M. Ylianttila and E. de Oca, "SDN and NFV integration in generalized mobile network architecture", 2015 European Conference on Networks and Communications (EuCNC), 2015.

[49] I. Afolabi, M. Bagaa, T. Taleb, H. Flinck, “End-to-End Network Slicing Enabled Through Network Function Virtualization,” To appear in IEEE CSCN 2017, Helsinki, Finland, September 2017.

[50] 3GPP TS 36.401, “LTE; Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Architecture description", Release 12, April 2015.

[51] A. Checko, H. Christiansen, Y. Yan, L. Scolari, G. Kardaras, M. Berger and L. Dittmann, "Cloud RAN for Mobile Networks; A Technology Overview", IEEE Communications Surveys & Tutorials, vol. 17, no. 1, pp. 405-426, 2015.

[52] 3GPP TS 24.301, “Universal Mobile Telecommunications System (UMTS); LTE; Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3", Rel.10, June 2011

[53] "Howtoconnectcotsuewithoaienbnew · Wiki · oai / openairinterface5G", GitLab, 2017. [Online]. Available: https://gitlab.eurecom.fr/oai/openairinterface5g/wikis/HowToConnectCOTSUEwithOAIeNB [Accessed: 10- Jul- 2017].