Software Defined Networks and Network Function Virtualisation Testbed within FIRE+

5G Mobile Core Network Slicing on an Orchestrated and Virtualised Infrastructure

Deployment of virtual mobile core network slices on the SoftFIRE infrastructure

December 2017

Table of Contents

Table of Contents 2
List of Figures 3
1 Introduction 4
2 Overview of the 5G Core Network at 5GIC 5
2.1 Context-aware network core 6
2.2 Control and user plane separation (CUPS) in the mobile core 7
3 Network Slices in the 5GIC Component Testbed of SoftFIRE 7
3.1 Deployment of the User Plane Node Network Slice on the Virtualisation Infrastructure 8
3.2 Deployment of FDC components as VNFs in SoftFIRE 9
3.3 Deployment of the network service NS(UPN) 10
4 Demonstrations of the Virtualised 5G Core 10
5 Use of the UPN Network Slices by SoftFIRE Experimenters 11
5.1 SOLID 11
5.2 Experience 13
6 SoftFIRE Experimenters with Own Network Slicing Solutions 14
7 Concluding Remarks 14
Bibliography 17
List of Acronyms and Abbreviations 18

List of Figures

Figure 1. 3GPP 5G Reference Non-roaming Architecture [13]. 5

Figure 2. Flat Distributed Cloud (FDC) architecture [12]; 5gD: 5G Device, i.e. a 5G UE. 6

Figure 3. UPN(CM): UPN of a micro cell managed by a CM, and UPN(CC): UPN of a macro cell managed by the CC. 8

Figure 4. Network slices provided to a SoftFIRE experimenter; N1,...,N6: 5G core network interfaces. 8

Figure 5. Demo architecture for Skype call between two UPN slices A and B. 11

Figure 6. GridNet[23]'s SOLID Experiment Architecture on the SoftFIRE testbed. 12

1 Introduction

Network Function Virtualisation (NFV) [1][2] is a revolutionary paradigm that makes it possible to rapidly deploy a self-contained network function as software on-demand, whenever the network operator requires to do so. The demand for a new instance of *virtual network functions* (VNF) could be for various reasons, such as the need for dedicating instances of the same VNF to different multiple customers or different vertical markets, or to provide quick response to sudden surges in demand for that particular network function. In the case of different customers with their dedicated VNFs, the customers may be running various services, and also addressing the requirements of differing vertical markets by configuring their dedicated VNF instances specifically for their provided services.

The NFV paradigm is now causing a shift in operators' approach to providing their services. Instead of running network functions on dedicated and specialised hardware, it has become an increasingly attractive option to run virtualised network functions (VNF) on standard off-the-shelf blade server equipment, decreasing capital expenditure (CAPEX) and operational expenditure (OPEX). Therefore, NFV is envisioned to shape the future of networking services. The ability to quickly instantiate a VNF on-demand, to simultaneously run multiple instances of the same VNF to support demand peaks, and to dedicate VNFs to different vertical markets or different customers will provide network operators with substantial benefits, such as high flexibility, scalability, robustness.

Project SoftFIRE [3] has an ambition to prove the power of NFV, especially in the context of Fifth Generation (5G) networks [4]. Through engagement with its experimenters, the project has facilitated NFV-based services offered by various SMEs, and further exploited the power of virtualisation infrastructure in a federated multi-site infrastructure. To provide a virtual mobile network core aligned with the Third Generation Partnership Project (3GPP) [5] directions, project partners developed virtual Network Services (NS) for the mobile core network, each consisting of multiple VNFs that run different components of the mobile core. In particular, in the 5G Innovation Centre (5GIC) [6], an already softwarised mobile core implementation (which was developed by 5GIC) has been virtualised in the context of SoftFIRE platform and development activities [7] The mobile core network software complies with the specifications published by 3GPP, in particular the concept of Control and User Plane Separation (CUPS) [8].

The virtualisation platform, which was prepared as part of project SoftFIRE, made it possible to demonstrate the full functionality of a virtualised more core network in numerous workshop events that were organised by 5GIC. In these workshops, a fully virtualised mobile core, orchestrated by the ETSI NFV MANO (Management and Orchestration) [9] compliant orchestrator Open Baton [10], was demonstrated. This orchestrator was developed by the SoftFIRE project partner Fraunhofer FOKUS [11] and functionally extended by the SoftFIRE project. 5GIC engineers demonstrated that on-demand orchestrated instantiation of a mobile core network can be accomplished in only under two minutes. This proves the power of NFV and the functional adequateness of the SoftFIRE platform. This flexibility and the speed of deployment are notable advantages for a mobile network operator, which normally can deploy a core network using standard hardware in several weeks. The orchestrated virtualised mobile

core can run multiple network slices, each of which can be associated with a separate vertical market.

This white paper outlines the mobile core network slicing technology provided by SoftFIRE to its experimenters. The rest of the paper is organised as follows. First, an overview of the 5G core network architecture and its concepts are introduced in Section 2. Then, in Section 3, the network slices that were developed in SoftFIRE based on this architecture are presented, providing details of the virtualisation packages. This is followed by Section 4, in which the demonstration of the network slices is described, and some of the demonstrations performed by 5GIC engineers are described in some detail. Then, Section 5 briefly outlines the use of these network slices by some of the SoftFIRE experimenters. This is followed by the brief description of an experimenter that deployed and tested their own network slicing solution on the SoftFIRE infrastructure, as described in Section 6. Finally, Section 7 presents concluding remarks and some recommendations for future solutions.

2 Overview of the 5G Core Network at 5GIC

The 5G core network at 5GIC follows the Flat Distributed Core (FDC) [12] architecture that was designed in 5GIC. The concepts of this architecture were introduced to 3GPP and similar components and functionalities later appeared in the system architecture document 3GPP TS 23.501 [13]; the reference 3GPP 5G non-roaming system architecture is illustrated in Figure 1.

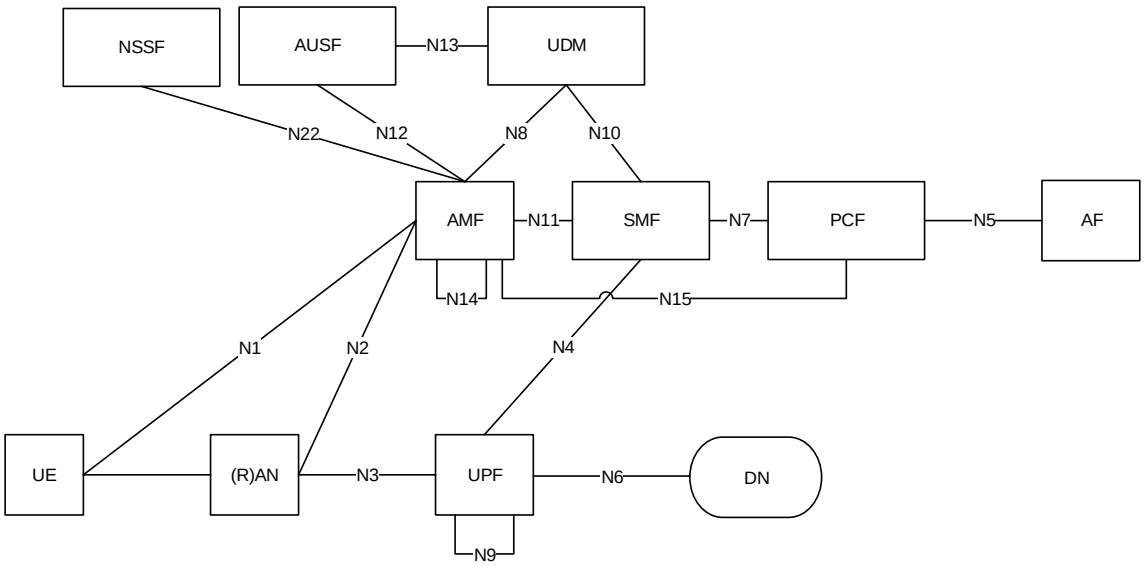

Figure 1. 3GPP 5G Reference Non-roaming Architecture [13].

The FDC architecture is based on the concept of Control and User Plane Separation (CUPS) [8], context-aware user plane anchoring, and distributed network functions envisioning ultra-dense deployments. Figure 2 below illustrates the FDC architecture.

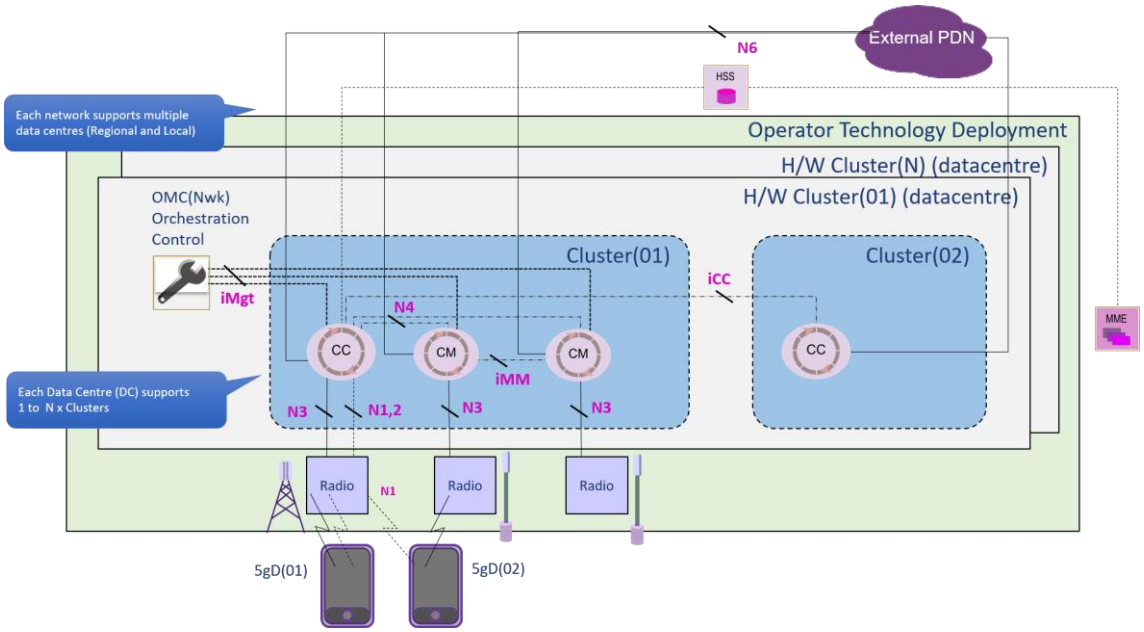

Figure 2. Flat Distributed Cloud (FDC) architecture [12]; 5gD: 5G Device, i.e. a 5G UE.

In this architecture, the network is organised as clusters of multiple micro cells, where each cluster includes a single macro cell. In FDC, a mobile has separate connectivity sessions: control plane and user plane. The control plane interface now appears in TS 23.501 as the N1 interface between the UE and the Access and Mobility Management Function (AMF). Within a cluster, the control plane of a user equipment (UE) is always carried and maintained through the macro cell of that cluster through interface N1, whereas the user plane is either on one of the micro cells or on the macro cell, depending on the mobility context of the user. The UP interface is named as N3 in TS 23.501. The Cluster Member (CM) entities in the FDC architecture are each logically associated with a single cell, governing the cell's UP operations. The combination of all CM functions is now included in the Session Management Function (SMF) entity in TS 23.501. CC and CM entities are explained in more detail in the following.

2.1 Context-aware network core

The FDC architecture is designed to perform context-aware operations, based on a user's context, such as mobility. To support context-aware operations, each UE has a control plane connection to a new entity called the Cluster Controller (CC). There is a single CC entity which is logically associated with a macro cell, i.e. each macro cell has a separate and single instance of CC, controlling its user-context based operations. One fundamental context-aware operation is user plane anchoring, which means that the user plane (UP) of a UE is either on a micro cell (when the user is stationary or has low mobility) or on a macro cell (when the user has medium to high mobility). The decision on where UP anchoring is provided at can depend on various factors, such as currently running network services at the cluster, the services requested by the UE, the UE's geographical location, and its mobility.

The FDC architecture also includes a separate user plane control (UPc) component, called the Cluster Member (CM). There are multiple instances of a CM in each cluster, each of which is associated with a single micro cell. Each micro cell has a single CM associated with it. The CC

node associated with the macro cell communicates with the CM instances in its cluster, and performs operations based on user context, and governs all UE's user plane, i.e. the data connectivity provided to the UEs when they are anchored at the macro cell.

2.2 Control and user plane separation (CUPS) in the mobile core

In the FDC architecture, the gateway nodes of a conventional LTE (Long Term Evolution) network, i.e. the Packet data network Gateway (PGW) and the Serving Gateway (SGW), are each divided into two: control plane and user plane components. The control plane functionalities of a PGW are componentised as PGWc, and its user plane functionalities are now a separate component called PGWu. Similarly, SGW has been divided into a control plane entity SGWc and a user plane entity SGWu. This separation has enabled the core network to separately instantiate user plane and control plane functions in different network slices, which was necessary to dedicate a user plane slice to a separate vertical market or a customer. In the 3GPP 5G system architecture, the operations of SGWc and PGWc are collocated at the SMF entity, which conforms to what was proposed in the FDC architecture.

A further improvement in the core network architecture is the collapse of the GTP (GPRS Tunnelling Protocol) tunnel called S5, which appears between PGW and SGW in an LTE evolved packet core (EPC) network. This has led to a single user plane component that consists of a PGWu and SGWu. This component is referred to as the Packet Processing Entity (PPE), as shown in Figure 3. This concept of a unified user plane component has been adopted by 3GPP, and now appears as the User Plane Function (UPF) in TS 23.501 [13], as seen in Figure 1. In FDC, each cell is provided with at least one PPE, i.e. macro cells and micro cells all have at least one PPE which provides user plane functions to users.

3 Network Slices in the 5GIC Component Testbed of SoftFIRE

Section 2 presents the FDC architecture and the software components of the core network provided by the 5GIC component testbed in SoftFIRE. In this section, the network slices that are made up of the mobile core network VNFs, which are derived from these software components, are explained. Network slicing technology in the 5GIC testbed is motivated by not only the flexibility and scalability provided by the NFV paradigm, but also by the fact that future mobile network solutions must be increasingly context-aware. This goal of 5G networks gave rise to context-aware operations, which are inherently enabled by the FDC components CC and CM.

Since all context-aware operations in a cluster in FDC are maintained by the CC component, CC is associated with the macro cell of the cluster. When virtualised, the CC VNF can have simultaneously running multiple instances, each of which can be provided to a separate experimenter in SoftFIRE. Furthermore, as CC is the entity that governs the user plane control functions for those users whose UP is anchored at the macro cell of a cluster, this is integrated with the PPE of that cluster's macro cell. Similarly, to manage the user plane operations in a micro cell, the PPE of the micro cell is associated with a CM instance. This design has led to a new core network component called the User Plane Node (UPN). A UPN in a macro cell consists

of a single CC and a PPE, and is called UPN(CC), whereas the UPN of a micro cell consists of a CM and a PPE, and is called UPN(CM). This is depicted in Figure 3.

Figure 3. UPN(CM): UPN of a micro cell managed by a CM, and UPN(CC): UPN of a macro cell managed by the CC.

The UPN(CC) provides a cluster-wide APN (Access Point Name) towards the Internet, whereas UPN(CM) is a local breakout point for Mobile Edge Computing (MEC) applications at micro cell level. One of the SoftFIRE experimenters used their dedicated local breakout UPN(CM) to support their virtualised MEC solution, as presented in Section 5.

The control plane (CP) of the EPC of an LTE network is also combined and componentised into a single Control Plane Node (CPN), which consists of HSS (Home Subscriber Service), MME (Mobility Management Entity), as well as the new control plane components PGWc and SGWc. SoftFIRE provides a single CPN instance shared by all experimenters running their experiments at the 5GIC component testbed; however, it is technically possible to simultaneously run multiple instances of CPN when system load is high, or if CP operation is desired to be dedicated to different vertical markets. Hence, the flexibility of the FDC architecture has been well-aligned with different network slicing options both in CP and UP. Separating the CP and UP in FDC has facilitated creating multiple UP slices, each with an individual APN.

3.1 Deployment of the User Plane Node Network Slice on the Virtualisation Infrastructure

In SoftFIRE, the 5GIC component testbed provides a dedicated UP slice to each experimenter. Each UP slice consists of a UPN(CC) and a UPN(CM), packaged as a UPN Network Service, called NS(UPN).

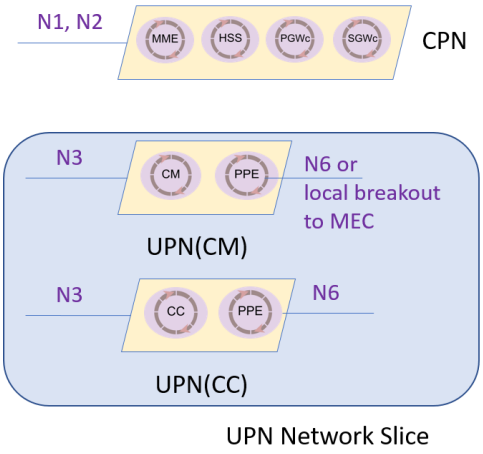

Figure 4. Network slices provided to a SoftFIRE experimenter; N1,...,N6: 5G core network interfaces.

The UPN(CC) is the Internet UP breakout, and has been designed as a separate VNF, in which CC and PPE are separate software components. Similarly, UPN(CM) is the local UP breakout VNF, where CM and PPE are its software components. The combination of a UPN(CM) and a UPN(CC) is the UPN network service (UPN(NS)) provided to an experimenter of SoftFIRE, as illustrated in Figure 4.

3.2 Deployment of FDC components as VNFs in SoftFIRE

The components of the FDC, i.e. UPN(CC) and UPN(CM) are deployed in separate Virtual Machines (VM) in OpenStack[1] [14]. Such deployment has been performed on several recent versions of OpenStack, most notably Liberty [15] and Newton [16]. Currently, OpenStack Newton version is the stable infrastructure controller[2] in the 5GIC component testbed of SoftFIRE.

As per Open Baton user instructions, VNF packages are prepared for UPN(CC) and UPN(CM) separately. Each package contains a *vnfd.json* file. This file contains the VNF Descriptor (VNFD) for the corresponding network function, which provides metadata to describe the VNF, including its name, type, virtual deployment unit (VDU), deployment flavour, virtual link, and lifecycle events.

Virtual deployment unit provides the details of the VM where the VNF is to be deployed, such as the Virtual Infrastructure Manager (VIM) instance (i.e. 5GIC testbed VIM), Operating System (OS) image (e.g. cloud compatible versions of Ubuntu [18], Centos [19], or CirrOS [20] operating systems), and the public IP address to use. The *deployment flavour* defines a VM's virtual compute resources, i.e. CPU cores, RAM, and disk storage space. *Virtual link* defines the name of the internal network[3] to use in the OpenStack environment. Lifecycle events define the necessary steps to perform the deployment of the VNF on the VM: instantiation, configuration, start, and terminate. Once a VM boots, its instantiation takes place where any necessary software is installed automatically. Once instantiated, 'configure' events occur, where necessary properties are fetched as software dependencies; configuration parameters are used to create the necessary variables. The VNFD file also refers to separate script files for each lifecycle event; the script files are also included as part of the virtualisation package, and provided to the orchestrator.

Besides the VNFD file, a *metadata.yaml* file is also required for deployment of a VNF on a VM. The name referred to in this file is used to identify the required image to use to deploy the VM.

[1] OpenStack is an open source tool for managing pools of compute, storage and network resources, within a datacentre.

[2] Initially, the developer version of OpenStack, i.e. DevStack [17] was used to optimize performance and the functionality was proven with live demonstrations. However, following several unplanned power-cuts and the following difficulty in retaining the stability of the OpenStack platform due to a complex matrix of dependencies among its software components, the 5GIC engineering team decided to deploy the packaged version of OpenStack, called PackStack, which proved to be more stable, yet with limited capability to modify its deployment configuration that would boost mobile network performance.

[3] OpenStack networking defines internal and external networks, where an external network has a publicly reachable address space, whereas an internal network's address space is internal to OpenStack. Each VM has an internal and an external IP address.

When the image is already present in the VIM, it is directly used. However, in the event that the requested image identified by its name is not already present in the VIM, this file provides details that are to be used to create the required OS image.

The lifecycle event scripts of the UPN(CC) and UPN(CM) scripts must explicitly define the IP address of the VM where these components are to be deployed. This is necessary for two reasons:

(i) IP address planning in the public IP space,

(ii) UEs and the Random Access Network (RAN) equipment require definitive IP addresses, i.e. it is not practical to re-program the rest of the network (which require manual reconfiguration) when OpenStack assigns VM IP addresses randomly.

Since the IP address of each VM can only be specified in a VNFD, each UPN requires its own VNF package. Generation of VNF packages are automated by scripts, which can then be fed into the orchestrator.

3.3 Deployment of the network service NS(UPN)

The Network Service NS(UPN) represents a single slice of the user plane functions of the FDC, and its deployment on OpenStack is performed via the orchestrator Open Baton in project SoftFIRE. The network slice is illustrated as the encircled part in Figure 4.

To deploy NS(UPN), an NS Descriptor (NSD) is prepared, which defines how UPN(CC) and UPN(CM) VNFs can be combined into a network service. It is important to note that deployment of VNFs on VMs, as outlined in Section 3.2, happens when the NS(UPN) is deployed on the VIM.

The NSD includes various conventional fields, such as its vendor, version, name, list of VNFDs to include in the NS, and the virtual link descriptor (VLD), which is the name of the network to be used by the NS. The NSD also includes an important field: the dependencies. Dependencies are a VNF's properties and configuration parameters that are to be shared with another VNF (or a set of other VNFs), as the deployment or correct execution of that other VNF(s) "depends" on these values. For instance, UPN(CC) and UPN(CM) need to communicate so as to perform context-aware user plane decision making operations. This communication is only possible if UPN(CM) knows the external IP address of UPN(CC), and UPN(CC) knows the internal IP address of UPN(CM)[4].

4 Demonstrations of the Virtualised 5G Core

The virtualised 5G core network and the network slicing functionality have been demonstrated to the members of the 5GIC network [21] in numerous occasions, as well as European Commission members, on the 5GIC component testbed of the SoftFIRE platform. In these demonstration events, instantiation of the NS(UPN) network service as a network slice on the

[4] Please note that such dependencies are not requirements for correct operation of the FDC, but merely implementation details; i.e. the CC and CM implementations can be modified so that these components perform module discovery upon start up. Current implementation is based on IP address dependencies.

5GIC Openstack VIM (in Guildford, UK) was performed via interfacing with the Open Baton orchestrator instance located in the FOKUS component testbed of SoftFIRE (in Berlin, Germany).

The demonstrations include instantiation of a new UPN network slice, which takes less than two minutes. Once deployed, a Skype [22] call is demonstrated between two mobile phones, each of which receiving data connectivity from a different network slice, one of which is the newly instantiated UPN slice, as illustrated in Figure 5. These UPN network slices are on different APNs. The demonstration also includes 4K video download to a mobile phone through the newly instantiated network slice.

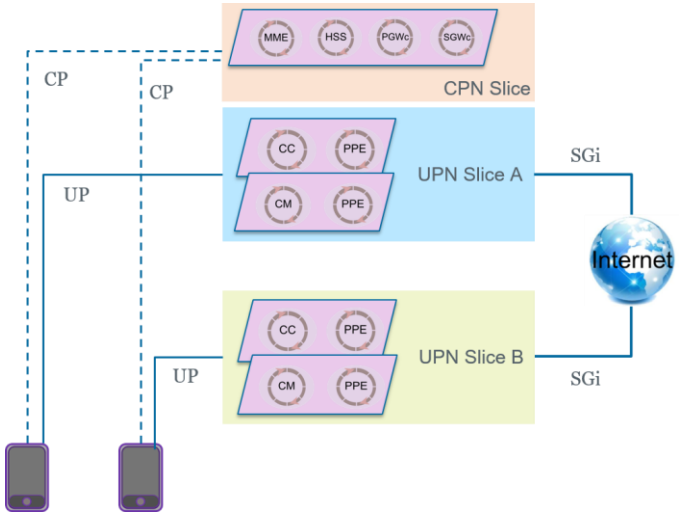

Figure 5. Demo architecture for Skype call between two UPN slices A and B.

5 Use of the UPN Network Slices by SoftFIRE Experimenters

In this section, two selected experiments that were performed on the SoftFIRE testbed are briefly presented; these experiments used the NS(UPN) network service as a network slice at the 5GIC component testbed of SoftFIRE.

5.1 SOLID

This experiment was performed by the company Gridnet [23] during the Wave 1 of Experimentation of the SoftFIRE project. The experiment aimed to demonstrate virtualized intelligent multi-access user data traffic control and traffic offloading. Many solutions have been proposed and trialed so far to achieve the goal of multi-access user bearer control, such as MP-TCP, SIPTO and LWA. However, all of these techniques have significant drawbacks, or are a generation away as they require significant modification, as follows:

MP-TCP: is generally outlawed by operators due to its inherent security risks in passing through firewalls,

SIPTO: has largely been neglected due to its lack of support for Lawful Intercept (LI) and accounting

LWA: is likely to become a staple method for multi-access control; however this approach may take several years to become de-facto as it requires significant upgrade of Wi-Fi and LTE base stations in the field as the technique requires modification of the MAC level.

The solution offered by SOLID was different to existing offloading solutions. The main objective was to leverage the technologies of SDN and NFV provided by the SoftFIRE facilities in order to build a sophisticated offloading framework for heterogeneous networks (LTE/LTE-A & Wi-Fi) that is driven by the end-user perceived QoS.

The SOLID experiment operated a tunneling protocol TUP between User Equipment(s) (UE) over both LTE and Wi-Fi towards a new entity called an EPC-Bridge that can make intelligent routing decisions in coordination with each UE in order to optimize usage of the dynamic performance of each available Radio Bearer (Wi-Fi and/or LTE-A). The multiple access technologies used in the experiment were Wi-Fi and LTE although in practice others could also be used.

SoftFIRE provided the experimenter with a dedicated EPC(UPN) slice, and since the experimenter's UEs were static, the local breakout UPN slice component, i.e. UPN(CM) was used. The SGi interface (called N6 in 5G system architecture) was where the Experimenter's modified Open vSwitch (OVS) solution, called the EPC-Bridge, was connected. The EPC-Bridge function was virtualized and connected on the north side of the UPN(CM) through the N6 interface, acting as a "bridge" between the UPN(CM) and the Internet.

The EPC-Bridge was also connected to a WiFi Mesh Gateway solution. The traffic from the UPN(CM) slice and from the WiFi gateway both passed through the EPC-Bridge, making it possible to monitor both the traffic through the UPN(CM) slice and the WiFi UP traffic. The tunnel end-points of the LTE and the WiFi networks were bridged with the Internet. The complete experiment architecture on the SOLID experiment by GridNet on the SoftFIRE testbed is illustrated in Figure 6.

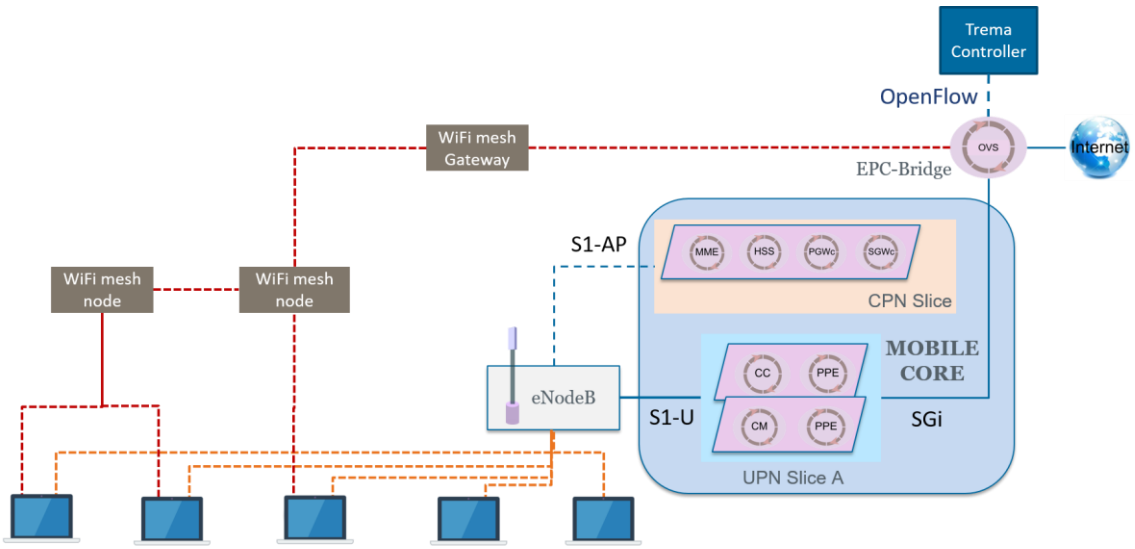

Figure 6. GridNet[23]'s SOLID Experiment Architecture on the SoftFIRE testbed.

The EPC-Bridge was programmed using a Trema [24] controller, which was deployed on the same VM as the EPC-Bridge as a single combined VNF. The controller programmed the EPC-Bridge, so that the UP traffic from devices that were attached to the 5G core could be monitored, and then a decision as to whether to inform a device to use WiFi (instead of 5G core) could be made when the load on the UPN(CM) was considerably higher than that on WiFi. Hence, the offloading mechanism decided which user traffic flow would be offloaded to the Wi-Fi network.

UEs sent their Service Level Agreement (SLA) values to the Controller, regarding their required downlink (DL) traffic. The offloading mechanism at the controller then detected possible SLA violations and undertook necessary decisions to offload traffic, based on the output of the dedicated network function which estimates the network performance of the LTE and Wi-Fi networks, based on the monitored traffic on the EPC-Bridge, from the WiFi Gateway and the UPN(CM).

5.2 Experience

The Experience experiment is conducted by the company Intellia ICT [25] during the currently running 3rd Wave of Experimentation of the SoftFIRE project. The experiment aims at making performance analysis of the company's provided virtual Augmented Reality (AR) solutions that run on the virtualization platform offered by the SoftFIRE.

The motivation for this experiment is as follows: 5G networks are to offer enhanced mobile broadband connections, but will also support low latency and increased Quality-of-Experience (QoE) for all the users of the network, which are necessary requirements for immersive AR applications. AR content includes new formats, such as stereoscopic, high dynamic range (HDR), and 360°, videos and 3D objects at increased resolutions (8K+) and higher framerates (90+ fps). Although basic implementations of these formats can be delivered through 4G networks, large-scale adoption of applications that use these formats will soon congest 4G network, thus rendering the user experience intolerable.

The AR scenarios are decompiled in a series of VNFs (AR content, storage, execution, content delivery) that are accessed via the 5G UPN(CM) slice dedicated to the experimenter; the experiment has been allocated with a separate UPN(CM) slice that connects the VMs running AR VNFs and the UEs that run AR applications. The experiment investigates the capability to insert these VNFs in an on-demand way, with respect to AR content storage, processing, and delivery, so as to achieve programmability in the network infrastructure. To achieve this, an external NFV controller software as well as a custom monitoring manager that includes a pre-configured Zabbix server are run by the experimenter.

A series of stress testing scenarios are to be applied to assess the performance of the infrastructure under different requirements imposed by AR applications in terms of three directions: (i) *network capacity*, (ii) *network latency, and (iii) uniform user experience.* This exercise is aimed to reveal best practices and adaptive strategies for the optimal delivery of AR content to UEs.

6 SoftFIRE Experimenters with Own Network Slicing Solutions

Network slicing has been a popular technology advancement in recent years. Several companies throughout the globe have been developing solutions to provide various network slicing deployments to their customers. One of the SoftFIRE experimenters, Cumucore [26], also demonstrated their own slicing solution in an experiment called 5GNaaS (5G Network as a Service).

The experiment ran two separate VMs, each including a different type of network slice. The first VM included both control plane and user plane parts of an LTE network. This slice had a dedicated Femto cell equipment connected to it on a dedicated PLMN ID. The second slice had a separate LTE network core on another PLMN, yet only its control plane; the user plane of this second LTE network was deployed on a physical server on the path in between a second Femto cell and this second VM. This second LTE network was deployed to demonstrate the User Plane Function (UPF) of the newly introduced 5G system architecture [13], at the edge of the network.

The UPF in this experiment consisted of the entire SGW and PGW, retaining the UP and CP parts as in LTE, yet was deployed at the network edge as a proof-of-concept demonstration of UPF deployment to support MEC applications. In addition to showcasing deployment of custom mobile network core slices, either as CP-only or CP and UP combined, the experiment also included Software Defined Networking (SDN) functionality, identifying traffic flows based on PLMN IDs, which was needed to program an Open Virtual Switch (OVS) [27] that ran at the same server where the edge UPF ran, so that traffic redirection could be performed to this edge UPF.

7 Concluding Remarks

5G mobile networks are envisioned to provide flexibility and scalability to network operators, enabling support for various types of network services whilst effectively isolating the traffic of these network services. With the advancements in Release 14 of 3GPP specifications regarding control and user plane separation now possible in mobile core network, operators can run dedicated instances of network services just for the user plane, i.e. the core network functions that support data traffic. Recent technologies called NFV and SDN will make it possible to have virtualized network services running on standard COTS server equipment, and enable instantiating multiple core network slices at the same time. With NFV, network slices can be dedicated for different vertical markets, or customers, and can be used to support increasing load on either control plane or the user plane operations of a mobile core network.

The core network slicing solution presented in this white paper has been created and operated within the SoftFIRE context both by 5GIC as a partner of the SoftFIRE project and by the experimenters using the SoftFIRE platform during its waves of experiments. The provided network slicing solution is a prototype that has increased the know-how around this important feature of 5G networks.

Project SoftFIRE has made it possible to virtualize a mobile core network, and demonstrated its operation in numerous occasions to the industry and the academia. Furthermore, the project

achieved to instantiate network slices via the ETSI NFV compliant orchestrator Open Baton in just under two minutes, making it one of the first demonstrations of orchestrated mobile core network. This achievement of SoftFIRE also made it possible to provide a virtual core network instance to SMEs that have participated the Waves of Experiments organised and managed by the project. In doing so, the project has proved the power of NFV, and the possibilities it will bring to 5G networks, thus creating an incubator of this technology, attracting researchers and engineers from industrial organisations and academic institutions to provide various virtualisation solutions, as observed in various experimentation waves of the project.

Strengths

The project has proven the following strengths of NFV with respect to mobile network core:

- Rapid deployment of a user plane core network slide under 2 minutes,
- Multiple core network slices can easily share the same RAN,
- Rapid re-arrangement of sharing configuration.

Opportunities

The project has also highlighted a number of opportunities:

- There is a substantial potential market for NFV and SDN platforms (e.g. orchestrators and controllers),
- Further standardisation of the control procedures and interfaces is required across platforms at the controller and MANO levels
- Need a much better NFV platform controller to complement MANO based platforms, i.e. OpenStack and the like should be tailored to provide improvements and meet the particular performance requirements of mobile operators,
- A scalable support form of FOSS with service contracts from FOSS to fully paid up maintenance and roadmap support is desired for platform software,
- A single OVS per Compute server in OpenStack is a performance bottleneck. External OVS to NFV platforms could provide scalability / pre-emption. Firmware OVS with formal interface to NFV platforms could be introduced to reduce latency in user plane.

Weaknesses

The project has also determined a number of weakness in the current virtualisation platforms, that should be addressed to make virtualisation more attractive to operators:

- Integrated OVS in OpenStack is a performance bottleneck. There is only one OVS per Compute server (irrespective of the number of VMs). As a result, the operator cannot control performance between VMs on the same host machine,
- Subversion of networking programming of OpenStack to an external SDN controller avoids unnecessary UP migration between OpenStack servers, but is complex to set up,
- OpenStack requires improvement to become a robust codebase for telecoms at present,
- OpenDaylight programming approach requires improvement to simplify programmability, ensure security, and provide isolation between users,

Threats

- Complexity, support, and latency of NFV and SDN FOSS platforms need more improvement, otherwise telecoms operators may build /commission proprietary platforms, or sidestep this technology.

Bibliography

[1] "Network Function Virtualisation: State-of-the-Art and Research Challenges", Rashid Mijumbi, Joan Serrat, Juan-Luis Gorricho, Niels Bouten, Filip De Turck, Raouf Boutaba, *IEEE Communications Surveys & Tutorials*, vol 18, no 1, 236-262, September 2015,

[2] "Network Functions Virtualisation— Introductory White Paper", ETSI, 22 October 2012, retrieved 20 June 2013.

[3] EU SoftFIRE project, https://www.softfire.eu/

[4] Fifth Generation Mobile Networks (5G), 3GPP Release 15, http://www.3gpp.org/release-15

[5] Third Generation Partnership Project (3GPP), http://www.3gpp.org/

[6] 5G Innovation Centre, University of Surrey, http://www.surrey.ac.uk/5gic

[7] 5GIC Demonstration Events to its members: Orchestrated Network Slicing, and press release, https://www.surrey.ac.uk/mediacentre/press/2017/world-first-demonstration-virtualised-5g-architecture

[8] TR 23.714, "Study on Control and User Plane Separation of EPC nodes", 3GPP.

[9] ETSI MANO specification, http://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_NFV-MAN001v010101p.pdf

[10]OpenBaton, https://openbaton.github.io/

[11] Fraunhofer Fokus, FUSECO Playground, https://www.fokus.fraunhofer.de/go/en/fokus_testbeds/fuseco_playground

[12] "Flat Distributed Cloud (FDC) architecture", 5G Innovation Centre, https://www.surrey.ac.uk/sites/default/files/5G-Network-Architecture-Whitepaper-(Jan-2016).pdf

[13] TS 23.501, System Architecture for the 5G System, 3GPP.

[14] OpenStack open source cloud computing software, https://www.openstack.org/

[15] OpenStack Liberty, https://www.openstack.org/software/liberty/

[16] OpenStack Newton, https://www.openstack.org/software/newton/

[17] DevStack, https://docs.openstack.org/devstack/latest/

[18] Ubuntu operating system, https://www.ubuntu.com/

[19] CentOS operating system, https://www.centos.org/

[20] CirrOS operating system, https://launchpad.net/cirros

[21] 5G Innovation Centre member network, https://www.surrey.ac.uk/5gic/members/network

[22] Skype, https://www.skype.com/en/

[23] Gridnet, http://gridnet.gr/

[24] Trema controller, https://trema.github.io/trema/

[25] Intellia ICT, http://www.intellia.gr/

[26] Cumucore, https://cumucore.com/

[27] Open Virtual Switch (OVS), http://openvswitch.org/

List of Acronyms and Abbreviations

Acronym	Meaning
3GPP	Third Generation Partnership Project
5G	Fifth Generation Mobile Network
5GIC	5G Innovation Centre
AF	Application Function
AMF	Access and Mobility Management Function
APN	Access Point Name
AUSF	Authentication Server Function
CC	Cluster Controller
CM	Cluster Member
COSS	Commercial Open Source Software
CP	Control Plane
CPN	Control Plane Node
CPU	Central Processing Unit
CUPS	Control and User Plane Separation
DN	Data Network
EPC	Evolved Packet Core
ETE	End-to-End
ETSI	European Telecommunications Standards Institute
FDC	Flat Distributed Cloud
FOSS	Free Open Source Software
HSS	Home Subscriber Server
IP	Internet Protocol
LTE	Long Term Evolution
MANO	Management and Orchestration
MEC	Mobile Edge Computing
MME	Mobility Management Entity
NS	Network Service
NSD	Network Service Descriptor
NFV	Network Function Virtualisation

NFVO	Network Function Virtualisation Orchestrator
OS	Operating System
OVS	Open Virtual Switch
PCF	Policy Control Function
PGW	Packet data network Gateway
PGWc	PGW control
PLMN	Public Land Mobile Network
PPE	Packet Processing Entity
RAM	Random Access Memory
RAN	Radio Access Network
SDN	Software Defined Networking
SGW	Serving Gateway
SGWc	SGW control
SMF	Session Management Function
TS	Technical Specification
UDM	Unified Data Management
UE	User Equipment
UP	User Plane
UPF	User Plane Function
UPN	User Plane Node
VDU	Virtual Deployment Unit
VIM	Virtual Infrastructure Manager
VLD	Virtual Link Descriptor
VM	Virtual Machine
VNF	Virtual Network Function
VNFD	Virtual Network Function Descriptor
VNFM	Virtual Network Function Manager

Disclaimer

SoftFIRE has received funding from the European Union's Horizon 2020 research and innovation programme under Grant Agreement no. 687860.