SELFNET 5G mobile edge computing infrastructure

Chirivella-Perez, Enrique; Marco-Alaez, Ricardo; Hita, Alba; Serrano, Ana; Alcaraz Calero, Jose M.; Wang, Qi; Neves, Pedro M.; Bernini, Giacomo; Koutsopoulos, Konstantinos; Gil Pérez, Manuel; Martínez Pérez, Gregorio; Barros, Maria João; Gavras, Anastasius

Published in:
Software - Practice and Experience

DOI:
10.1002/spe.2681

Published: 06/04/2020

Document Version
Peer reviewed version

Citation for published version (APA):
Chirivella-Perez, E., Marco-Alaez, R., Hita, A., Serrano, A., Alcaraz Calero, J. M., Wang, Q., Neves, P. M., Bernini, G., Koutsopoulos, K., Gil Pérez, M., Martínez Pérez, G., Barros, M. J., & Gavras, A. (2020). SELFNET 5G mobile edge computing infrastructure: design and prototyping. *Software - Practice and Experience*, *50*(5), 741-756. https://doi.org/10.1002/spe.2681

ORIGINAL ARTICLE

Journal Section

SELFNET 5G Mobile Edge Computing Infrastructure: Design and Prototyping

Enrique Chirivella-Perez[1] | Ricardo Marco-Alaez[1] | Alba Hita[1] | Ana Serrano[1] | Jose M. Alcaraz Calero[1] | Qi Wang[1] | Pedro M. Neves[2] | Giacomo Bernini[3] | Konstantinos Koutsopoulos[4] | Manuel Gil Pérez[5] | Gregorio Martínez Pérez[5] | Maria João Barros[6] | Anastasius Gavras[6]

[1] The University of the West of Scotland, Paisley, United Kingdom

[2] Altice Labs, SA, Portugal

[3] Nextworks, Italy

[4] Creative Systems Engineering (C.S.E.) Monoprosopi EPE, Greece

[5] Faculty of Computer Science, University of Murcia, Murcia, Spain

[6] Eurescom GmbH, Germany

Correspondence

Jose M. Alcaraz Calero, School of Computing, Engineering and Physical Sciences, The University of the West of Scotland, Paisley, United Kingdom

Email: Jose.Alcaraz-Calero@uws.ac.uk

Funding information

EU Commission Horizon 2020 Programme, Grant Number H2020-ICT-2014-2/671672; VP Fund 5G Video Lab.

This paper presents the design and prototype implementation of the SELFNET Fifth Generation (5G) Mobile Edge Infrastructure. In line with current and emerging 5G architectural principles, visions and standards, the proposed infrastructure are established primarily based on a Mobile Edge Computing (MEC) paradigm. It leverages Cloud Computing, Software-Defined Networking (SDN) and Network Function Virtualization (NFV) as core enabling technologies. Several technical solutions and options have been analysed. As a result, a novel portable 5G infrastructure testbed has been prototyped to enable preliminary testing of the integrated key technologies, and to provide a realistic execution platform for further investigating and evaluating SDN and NFV based application scenarios in 5G networks.

KEYWORDS

5G, Mobile Edge Computing, Software-Defined Networks, Network Function Virtualization, Deployment, Infrastructure

1 | INTRODUCTION

Whilst the Fourth Generation (4G) or Long Term Evolution (LTE) networks represent a significant step forward in terms of connecting people and providing advanced services to nomadic customers, the emerging Fifth Generation (5G) networks target to empower a fully connected global mobile society to create unprecedented socio-economic impact beyond the Year 2020. Such an ambitious vision is driving further innovation and underpinning the design and implementation of novel 5G network architectures. As part of the global 5G initiatives, the SELFNET project [1] has been launched in Europe under the EU Horizon 2020 5G-PPP (5G Infrastructure Public-Private Partnership) programme. In strategic terms, 5G networks are expected to deliver substantially improved performance defined by a set of Key Performance Indicators (KPIs) [2]. One of the key performance indicators is to reduce the time required to provision network management services within a 5G network from 90 hours to 90 minutes, in order to enhance the competitivity of the telecommunication operators in terms of both capital and operational costs and make their infrastructures agiler against constant business requirements. This KPI has been the main motivation for this research work. It is essential that the architectural design of the SELFNET framework is clearly aligned with 5G architectural visions and compliant with 5G standards under development. In particular, the development of SELFNET has been in line with the vision from the 5G-PPP such as [3], the 5G principles from the Next Generation Mobile Networks (NGMN) Alliance such as [4] and related 5G standards, especially ETSI (European Telecommunications Standards Institute) MEC (Mobile Edge Computing) [5][6] and ETSI NFV (Network Function Virtualization) [7].

The main contributions of this paper is to present the architectural design of SELFNET's 5G Mobile Edge Computing infrastructure together with a detailed explanation of how this infrastructure has been technically validated. This paper integrates the following scientific contributions as indicated below:

- Novel 5G network management architecture where a significant number of technologies, protocols and standards including SDN, NFV, Cloud Computing, Mobile Edge Computing and so on are combined and validated in a fully functional infrastructure.
- Highly coordinated automation amongst all the management planes available in the different layers of the architecture to provide zero-touch orchestration from bate metal.
- Significant extension to the ETSI NFV MANO (Management and Orchestration) standard to cover the management of the physical machines of the Mobile Edge Computing architecture.
- Achievement of the ambitious 5G-PPP KPI on service deployment time under 90 minutes in the proposed architecture, validated through different empirical stress tests.

The remainder of this paper is structured as follows. Section 2 presents an overview of the architectural design of the SELFNET infrastructure, Section 3 focuses on the management plane of the infrastructure and Section 4 describes the functional validation of the prototype testbed implementation. Section 5 shows the testbed and the empirical results of the 5G service deployment. Finally, Section 6 concludes the paper by explaining the potential development path being considered within the SELFNET consortium.

2 | SELFNET INFRASTRUCTURE DESIGN

Mobile Edge Computing. The architecture proposed is mainly based on, while significantly extending and improving, the Mobile Edge architecture envisioned by ETSI. Figure 1 depicts a number of network edges (two of which are shown), geographically separated from the data centre. These are used to allocate the operational and management services that need to be deployed close to the user in order to meet performance requirements. Figure 1 also shows a Cloud Radio Access Network (C-RAN) deployment [8] in a network edge (EDGE 1). Therefore, Edge 1 controls a pool of geographically dispersed antennas identified in the figure as Radio Last Mile (RLM) locations. The proposed does not aim to provide any new 5G air interfaces. This is an area currently being explored by a number of other 5G-PPP projects in Europe [9] and by other researchers across the globe (e.g., [10]).

C-RAN. Some assumptions about the future 5G air interface have been made when designing the proposed mobile edge architecture. In particular, it is assumed that the new 5G air interface would be compatible with a C-RAN deployment, which offers two locations where functionalities related to the data and control plane of the new air interface could be placed. This C-RAN approach follows the RAN cloudification trend in 5G and Mobile Edge Computing paradigms.

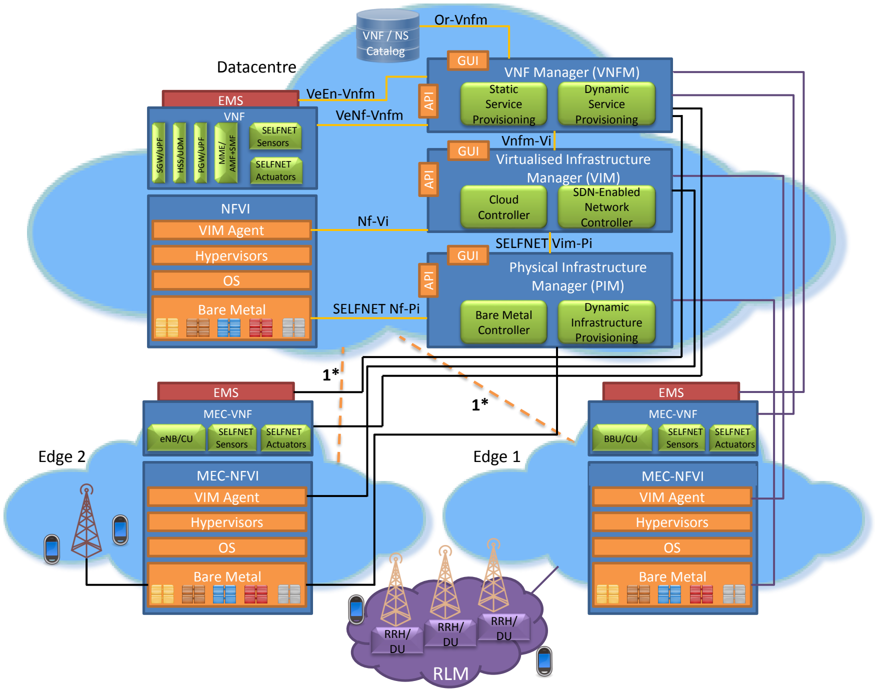


FIGURE 1 Architectural overview of the 5G SELFNET infrastructure

Optical Plane. Communication links between the edges and the data centre (1* in Figure 1) are expected to be of high density with very high data rates and very low latency. In a production environment, this connectivity would be complex and may encompass several technologies such as Wavelength Division Multiplexing (WDM) with Reconfigurable Optical Add-Drop Multiplexers (ROADM) at the ends of the communications link in order to meet critical 5G KPIs. The architecture does not focus on the data plane, consequently, this aspect has been simplified when prototyping the connection links between different edges. By using Commercial Off-The-Shelf (COTS) computing equipment for allocated software services in both datacentre and network edges, the proposed architecture is expected to significantly reduce capital investment costs. This can be achieved while providing services at the most appropriate location which may be at the edge closer to the user where required to improve the efficiency of a service.

IP Networks. It is assumed the use of IP networks. This architectural decision has previously driven the evolution from 2G/3G (Second/Third Generation) [11] to 4G [12] mobile networks and will continue being adopted in 5G networks. Such an assumption is necessary in order to both facilitate the execution of services on top and maximise its impact in the market.

Operating System (OS). In order to accommodate the widest range of services, the proposed infrastructure has been designed as a heterogeneous OS environment supporting the wide variety of OSs available in the market. Therefore, system architects and service designers will be able to select the most appropriate operating system for each service. For example, the allocation of a Baseband Unit (BBU) service at the network edge may require a real-time enabled OS, whereas the allocation of services for management purposes may tolerate traditional non-real time OSs.

Virtual Infrastructure. The use of virtualization is paradoxical, with significant pros and cons. Therefore, it is important to consider whether it should be an indispensable or optional component of 5G architectures. The main advantages of virtualization can be defined in terms of enhanced management, reliability, isolation and control of computational resources. Conversely, the main disadvantage is the performance penalty incurred by introducing this layer. Therefore, a trade-off between functionality and performance may need to be considered when designing 5G architectures. An investigation of the state of the art virtualization technologies has been carried out in order to analyse the level of performance penalties that they may incur.

Amit et al. [13] have recently provided a significant improvement in the data plane of virtualised workloads with intensive Input/Output (I/O) required in 5G infrastructures. A throughput penalty in the range of 0% to 3% for intensive I/O applications has been reported together with a latency overhead of around 2% in comparison to bare-metal performance. This performance penalty when using virtualization is largely negligible for modern servers and is outweighed by the benefits of virtualization, which justifies our employment of virtualization technologies in the proposed infrastructure. It is worth noting that, in order to support a wide range of use cases, the proposed infrastructure is able to deal with both virtualised and bare-metal service deployments (without virtualization). Furthermore, support for heterogeneous virtualization technologies may be required within the 5G architectures managed on top of the infrastructure. It is important to note that, regardless of the specific virtualization technology being used, the usage of virtualization implies the use of virtual switches implemented in software. These are to interconnect different Virtual Machines (VMs) allocated within the same physical machine. Different hypervisor technologies have been analysed using well-known hypervisors such as KVM (Kernel-based Virtual Machine) [14][15], QEMU (Quick Emulator) [16][17], VMWare [18][19], LXC (Linux Containers) [20], VirtualBox [21] and XEN [22]. As a preliminary result of this analysis, KVM has been recommended as a promising candidate to achieve hardware-based virtualization that meets the required performance levels. In addition, LXC has been recommended when kernel-based light virtualization providing solutions suitable for supporting software with real-time requirements is needed. Management of the virtualization layer provides multi-tenancy support over the virtual infrastructure by smartly configuring virtual switches and VMs. Use of a virtual infrastructure enables the deployment of virtual topologies. By adopting a Software-Defined Network (SDN) approach, the management of the different network segments and the control of the connection of VMs to a given network segment can also be enabled. Thereby giving the functionality needed to create any potential topology required in 5G architectures, and enable the self-management of such virtual topologies using the proposed infrastructure [23].

4G/5G Infrastructure. Figure 1 depicts an overview of a basic LTE (including LTE-Advanced) architecture, its current main architectural components and how they are deployed across different locations within the network. Initially, LTE has been employed to build the SELFNET framework, although going forward, the project will track the continuous evolution from LTE to 5G networks and continue to ensure that the framework remains aligned with ongoing 5G developments. Conceptually, LTE is divided into a control plane and data planes. LTE [24] has the following

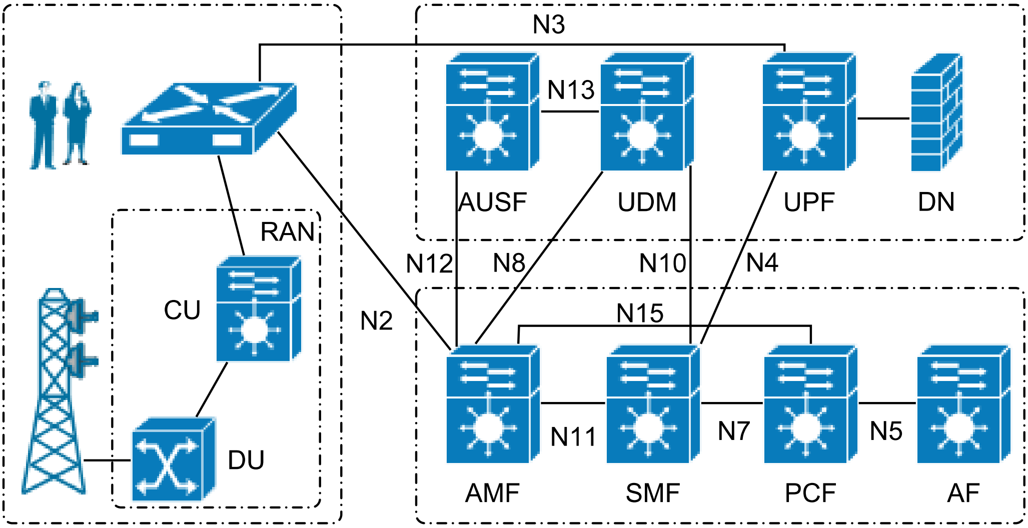


FIGURE 2 5G reference architecture

components:

- **Authentication Server Function (AUSF).** This function is part of the 3GPP 5G Architecture, which is used to facilitate 5G security processes.
- **Unified Data Management (UDM).** This component is related to the 3GPP 5G Architecture, supporting the ARPF (Authentication Credential Repository and Processing Function) to store long-term security credentials used in authentication for Authentication and Key Agreement. In addition, it stores subscription information.
- **Core Access and Mobility Management Function (AMF).** This function is part of the 3GPP 5G Architecture. Its primary tasks include Registration Management, Connection Management, Reachability Management, Mobility Management and various functions related to security and access management and authorization.
- **Session Management Function (SMF).** This function is related to the 3GPP 5G Architecture and is one of the main functions in the Next Generation Core. As such, it includes various functionality relating to subscriber sessions such as session establishment, modification and release.
- **Policy Control Function (PCF).** The PCF is related to the 3GPP 5G Architecture. This function supports the unified policy framework that governs network behaviour. In so doing, it provides policy rules to control plane function(s) to enforce them. In order to facilitate this, subscription information is gathered from the UDM function.
- **Application Function (AF).** The AF is a logical element of the 3GPP PCC framework which provides session related information to the PCRF in support of PCC rule generation.
- **User Plane Function (UPF).** The User Plane Function is related to the 3GPP 5G Architecture. It is similar to the roles played by the Serving/Packet Gateway in a 4G LTE system. The UPF supports features and capabilities to facilitate user plane operation. Examples include packet routing and forwarding, interconnection to the Data Network, policy enforcement and data buffering.
- **Data Network (DN).** The Data Network is related to the 3GPP 5G Architecture. It identifies Service Provider services, Internet access or third-party services.

Any UE gains access to the network via a connection to the antenna which is controlled by the E-UTRAN/Evolved Node B (eNB) or, in a Cloud-RAN environment, the BBU+Radio Remote Head (RRH) whose functionalities cover control of the air interface and provision of connectivity to both data and control planes of the mobile network. A more comprehensive description of the different architectural components available in LTE and their architectural relationship with the SELFNET project can be found in [25]. It is also worth noting that LTE infrastructures currently provide a number of mechanisms that enable LTE architectural components to be shared by different mobile operators. Several open source and closed implementations were analysed in order to inform our choice of the software stack to be used for prototyping purposes. These included the LTE-EPC (Evolved Packet Core) Network simulator LENA [26] running in the Network Simulator (NS-3) [27], OpenAirInterface (OAI) [28] and a number of proprietary solutions. Multi-tenancy capability support in the antenna, known as Multi-Operator Core Network (MOCN) [29], has been standardised by the 3G Partnership Project (3GPP) [30]. It is expected that 5G architectures will provide similar, or enhanced, mechanisms for sharing architectural components in a multi-tenancy environment. The main intention of multi-tenancy provisioning is to significantly reduce both capital and operational expenditures (CAPEX and OPEX, respectively).

SDN Controller. The proposed infrastructure decouples the data and control planes of the network and, in doing so, provides a logically centralised control element capable of governing the complete set of devices available in the network. This controller (henceforth referred to as the SDN Controller) has a holistic view of the network enabling it to enforce the set of actions that need to be performed in the network elements in order to correctly handle a new data flow passing through the network. This centralised management requires the use of SDN-enabled network elements capable of being configured by an SDN Controller. Importantly, centralization of control of the network does not imply a central point of failure or a bottleneck in terms of performance. In fact, modern SDN Controllers use clustering approaches, high availability and other methods to ensure their scalability, performance and to enforce governance. The SDN Controller underpins the network management and governance described in Section 3.

3 | NETWORK MANAGEMENT ON 5G SELFNET INFRASTRUCTURE

In SELFNET, great importance is attached to the reduction of provisioning time for new services. Across a range of use cases, automation is used as a key enabler to reduce provisioning time. This is accomplished in a range of scenarios where: i) new hardware is being inserted, replaced or removed from the infrastructure; ii) new services are being deployed, undeployed or redeployed; iii) support for the management of virtual infrastructures is inserted; iv) configured and controlled; v) virtual infrastructures are created, destroyed and migrated; and vi) virtual services are deployed, undeployed or redeployed in virtual infrastructures; etc. Figure 1 additionally depicts our vision for the management plane of the physical layer of Mobile Edge Computing infrastructures for 5G architectures. The vision is completely aligned with the standardised ETSI NFV MANO architecture [31], while providing significant extensions and improvements to the standard. In the interests of clarity, the naming of SELFNET components has been aligned with that of the ETSI NFV MANO standard [32].

Physical Infrastructure. The proposed architecture considers a number of architectural considerations. For example, management of multiple geographically separated physical locations, i.e., edges and data centre. Multi-tenancy hardware resource sharing amongst telecommunication operators (telcos), energy monitoring and management of network hardware and automated OS and software installation are provided by the SELFNET Physical Infrastructure Manager (PIM). SELFNET also enables management of the physical infrastructure, from bare-metal, through Cluster Management Configuration and Provisioning Tools such as Metal-as-a-Service (MaaS) [33], Rocks Cluster [34], HP Cluster Management Tool (CMT) [35]. The Physical Infrastructure Manager also provides the business logic for management and synchronization of the physical infrastructure [36]. This architectural element significantly optimises provisioning time and enhances the reliability and availability of the OS provisioning service for managed nodes.

Energy Management. One of the targets of the use cases proposed in SELFNET is to reduce end-to-end energy consumption. To do so, we need to firstly meter the energy consumed by all elements involved in the end-to-end provision of a service, starting from the wireless access network with its physical antennas and continuing with the computing resources that will be used to execute the software components in the virtual infrastructure. Two different approaches can be taken to monitoring energy usage: out-of-band interfaces or in-band interfaces. Open Energy Monitor [37] is an interesting example of an out-of-band tool, an open source monitoring system based on Arduino [38] that measures electricity consumption. This tool can be combined with SEGmeter [39], another Arduino based tool. eNOS [40] is a cloud-based open source energy management system that measures electricity, consumption, etc. and also provides a toolkit to build a cloud-based energy dashboard. The architecture has to provide a mechanism by which physical network resources can be remotely powered up or instructed to hibernate to meet the energy usage targets of the network. The Intelligent Platform Management Interface (IPMI) mechanism provides power management facilities by communicating, through a dedicated management network interface, with the Baseboard Management Controller (BMC). In-band power management can be also achieved using the Wake on LAN (WoL) mechanism. This legacy approach offers the ability to send control packets to the Network Interface Card (NIC), which in turn signals the power supply unit or motherboard to start/up shutdown. WoL uses MAC addressing rather than IP addresses. It lacks the functionality in remote access to Basic Input/Output System (BIOS) and Unified Extensible Firmware Interface (UEFI) and flexibility in out-of-band and sideband connectivity of the IPMI interface.

Service Provisioning. Once the physical infrastructure is provisioned with an OS, the infrastructure is ready for the provisioning of the software services. In fact, the same approach can also be adopted when a virtual infrastructure is already provisioned with an OS. The process of customizing an image to provide a specific service can be performed following a static or a dynamic approach. The static approach requires an image, which contains the appropriate software preconfigured, i.e., NVF software that it starts automatically after booting. The dynamic approach relaxes these constraints and allows the image to be customised after being started. The static approach to customization of volume images depends on operations staff to create a volume image prior to deployment, in which the OS and all services to be provided are already installed and properly configured according to the requirements of the organization. This approach creates standardised images that can be reused to deploy similar variants of the service but has many limitations. Firstly, the operation staff needs to perform the installation and configuration of all required services manually. Secondly, images must be maintained on a regular basis, applying patches for security and other reasons. Thirdly, each time the image is changed, the new version has to be uploaded. This might be very time consuming since a typical volume can easily be many GBs in size and must be transferred over the network. Finally, it is difficult to offer flexibility in configuration, since every configuration option leads to a possible new image that needs to be created, uploaded and maintained by operations staff. The dynamic approach has also advantages and disadvantages. Firstly, everything could be performed automatically, reducing the time to provision the services. Secondly, only the required software packages rather than the whole volume image needs to be sent over the network, making the deployment process much agiler. Finally, the maintenance of the base volume image is easier since there are fewer base images. The main disadvantage is that required time to boot an image with respect to a ready-to-run image previously configured using a static approach is significantly higher. Both approaches have been carefully analysed in the proposed architecture, resulting in a decision to support both approaches in order to offer flexibility in deciding which one is better for each use scenario and to allow a combination of what are essentially complementary approaches. The static provisioning of services can be provided by tools like Ghost [41], Clonezilla [42], PartImage [43] or FOG [44] at a physical layer, or by the virtualised file system provided by almost all the hypervisors available in the market such as Virtual Disk Image (VDI) [45], virtual Machine Disk (VMDK) [45], Quick emulator (QEMU) Copy on Write (QCOW2) [46], to name but a few. On the other hand, dynamic provisioning is usually performed by configuration management tools. There are a number of tools of this type available in the market such as Puppet [47], CHEF [48][49], SmartFrog [50][51], Juju [52] and Ansible [53][54]. These tools enable the automatic deployment and configuration of software services within the virtual and physical infrastructure. Figure 1 illustrates these two deployment capabilities together with the control of the life cycle of the deployed services by means of the Virtual Network Functions (VNF) Manager –VNFM– and its correspondent interfaces to the VNFs and Element Management System (EMS). The implementation of this automatic deployment of services is another important step in the reduction of the time required to deploy network services within the new 5G architecture.

Virtual Infrastructure. The virtual infrastructure proposed offers virtualised resources (network, computing and storage), cloud-based multi-tenant support, monitoring [55] and management of virtual resources, and VNF deployment support. The management of the virtual infrastructure is achieved through Infrastructure-as-a-Service (IaaS) [56][57]. Figure 1 shows the Virtual Information Manager (VIM) made up of an IaaS stack such as OpenStack [58], CloudStack [59], OpenNebula [60] and an SDN-enabled Network Controller such as OpenDaylight (ODL) [61], Open Network Operating System (ONOS) [62], Ryu [63] and FloodLight [64]. The VIM provides Authentication Services, Image Management Services, Block Storage Services, Certificate Management Services, Scheduling Services and a Networking Management Service. The Networking Management Service ensures that the network is not a bottleneck or a limiting factor in a virtual infrastructure deployment, and gives users self-service capabilities over network configurations. Multi-tenancy support for virtual infrastructures is one of the main capabilities achieved by the management plane of the virtual infrastructures. It enables tenants to share the same physical infrastructure in a completely isolated way where all virtual resources are isolated and protected between tenants. All components described above are able to provide multi-tenancy capabilities ranging from the GUI and APIs offering information only to the relevant tenant, to complete isolation of the virtual networks (shared in the physical plane) by different networking technologies, e.g., Virtual Local Area Network (VLAN) [65], Virtual Extensible Local Area Network (VXLAN) [66][67] and Generic Routing Encapsulation (GRE) [68][69], as well as other encapsulation technologies. This multi-tenancy should now be extended to the edges of the networks.

SDN Controller. The SDN infrastructure proposed employs an SDN Controller centric paradigm to link the business-specific SDN-Apps and the underlying virtualised network elements. Key technologies investigated include, amongst others, Virtual Tenant Network (VTN) [70], Service Function Chaining (SFC) [71], Network Slicing [4] and multi-tenancy support for SDN-Apps. An approach being primarily considered nowadays is the use of an SDN Controller as a logically centralised controller, where the control of the network functions and the governance of the different network elements managed in the infrastructure can be implemented. A logically centralised approach for management purposes enables the composition of holistic views of the network, and thus it enables the creation of advanced network intelligent protocols to control the traffic in the network, having wider information about the current status of the network provided by all network elements controlled. The main idea is to configure all network elements to be controlled by such SDN Controllers. Then, when the traffic is passing through such network elements, the SDN Controller will enforce the rules being configured in these network elements. The initial configuration and periodical updates of configuration will be conducted by the SDN Controller dynamically, according to the protocols running in the SDN Controller. Below the reader can find a list of key functionalities considered in the proposed architecture that may be provided by SDN-Apps for the 5G mobile edge infrastructure:

- **Virtual Tenant Network (VTN)** isolation manages virtual networks (Layer 2 or L2 domains) and virtual ports connected to such virtual networks in order to provide true L2 slicing and traffic isolation of the traffic associated to each of the tenants. The practical implementation of this functionality could be based on tunnelling and tagging protocols such as Multi-protocol Label Switching (MPLS) [72][73], VLAN, VxLAN, GRE, etc. The aim is to achieve the manipulation of logical maps of the network and create multiple co-existing virtual networks for each tenant regardless the underlying transport technology and network protocols.

- **Network slicing**, or more precisely infrastructure slicing in the context of this document, enables end-to-end virtual infrastructure sharing for multi-tenants. 5G network infrastructures will become increasingly multi-tenant, hosting heterogeneous types of tenants with different requirements, with the need of adopting a service model where each tenant is provided with its own virtual infrastructure. In this context, network virtualization becomes crucial and, when combined with IT virtualization, allows service providers to create multiple co-existing isolated end-to-end virtual infrastructures for their tenants.
- **Service Function Chaining (SFC)** [71][74][75] defines a specifically ordered list of network services/functions (e.g., load balancers, firewalls etc.) based on use scenario requirements, and manages traffic redirection functionalities to ensure that the traffic in the infrastructures is flowing through certain services.
- **Security Group Control (SGC)** [76][77] manages firewall-filtering functionalities in order to provide control of traffic being produced or consumed in the infrastructure. It includes filtering of traffic based on MAC and IP addresses, port number, packet type, etc.

Figure 3 shows an overview of the intersection between the data and control planes of the network. The bottom part presents the data plane. The introduction of a central switch logically connecting all edges has been considered (especially for the prototype testbed implementation). It will enable the SDN Controller to have an enforcement point where other overlay services and capabilities can be added into the network in order to control traffic flows. These functionalities have been identified in SELFNET Deliverable 2.1 [78] as a WAN (Wide-Area Network) Controller. It has been decided to consider this option in order not to limit the architecture vision. The inclusion of this type of SDN-App for controlling the WAN traffic would be decided later according to the requirements of the project towards the implementation of the use cases.

Figure 3 depicts an overview of the architecture deployed for the management of the control plane in Mobile Edge Computing infrastructures in 5G architectures. In the control plane shown in the upper part of Figure 3, the controller is logically centralised in the data centre. However, it is important to emphasise that a logically centralised SDN Controller does not mean that it would be, architecturally speaking, a single point of failure or a potential bottleneck in the network. The SDN Controller can be implemented in a highly distributed way yet offer a logically centralised abstraction for the control of the network. The SDN Controller has functional elements for such a distribution allocated in each of the edges and the data centre. Most SDN solutions are based on powerful network abstraction approaches that allow controlling and provisioning heterogeneous technologies in a unified way.

The extensions to the ETSI NFV standard proposed in this architecture are defined as follows:

- The extension of the management plane to include the new PIM architectural component in charge of the management of the life cycle of physical machines in the mobile edge computing infrastructure.
- A new interface between PIM and VIM to allow the management plane to provide homogeneous understanding of the different geographical zones available in the mobile edge computing infrastructure so that there is an alignment between the geographical zones of both physical and virtual computers.
- A new orchestrator that encloses wider responsibilities than those defined by the NFV Orchestrator existing in the ETSI MANO standard. This new orchestrator takes the responsibility to deploy services from bare metal with a

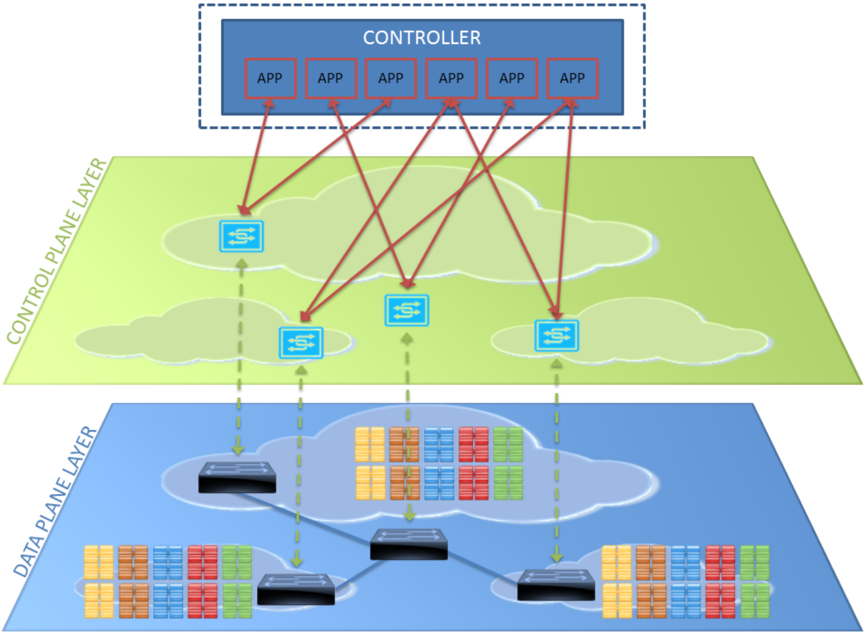


FIGURE 3 Overview of the architecture for the management of the control plane in Mobile Edge Computing infrastructures for 5G architectures

zero touch interaction with the infrastructure. This zero touch innovation allows network administrators to saved significant time in the service deployment.

It is noted that our proposal realizes an integrated multi-layer network management and orchestration architecture of 5G networks. Table 1 summarizes the technologies adopted in the validation of the proposal in this paper for the various functionalities required, across the different layers including physical, virtual and service layers, to provide a complete solution.

4 | FUNCTIONAL VALIDATION

There are a significant number of innovations indicated in this paper, and preliminary empirical validations of the essential innovative designs have been conducted. The following list summarises the prototypes that have been validated through implementation and functional demonstration to validate the technical approaches adopted in different components of the architecture presented in previous sections. These prototypes are shown in the infrastructure depicted in Figure 4.

- **Physical Infrastructure Manager (PIM).** This research has developed a customised version of Meta-as-a-Service (MaaS) [33] to enable us the deployment of a PIM suitable to control computational resources allocated in different geographical locations and to facilitate the deployment of the physical infrastructure in the data centre and the edges of the network. The extension has been focused on providing Mobile Edge Computing capabilities over MaaS.

TABLE 1 Functionalities and Technologies across Architectural Layers

	Physical Layer	**Virtual Layer**	**Service Layer**
Physical Infrastructure Management	MaaS	MaaS	-
Service Infrastructure Management	Juju	Juju	Juju
Hypervisor	QEMU	KVM	LXC/LXD
Networking	OpenvSwitch	OpenvSwitch	OpenVSwitch
SDNController	OpenDaylight	OpenDaylight SFC VTN	OpenDaylight SFC VTN
Energy Management	Open Energy Monitor	-	-
Monitoring	NFVMon	NFVMon	-
LTE	SDR Ettus x310	OpenAirInterface	-
Virtual Infrastructure Manager	-	Openstack	Openstack

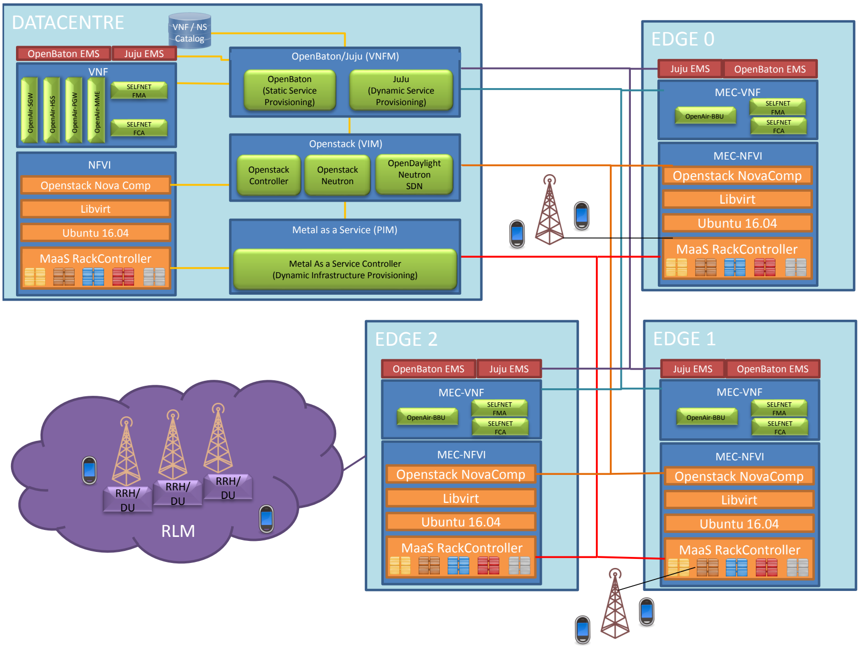


FIGURE 4 Prototyped 5G Mobile Edge Computing infrastructures

An initial support for MEC capabilities was already available in MaaS. However, MaaS has a strong requirement related to the assumption that all the nodes deployed within the same geographical zone have access to the Internet, whereas in 5G architectures, this is not normally the case and there is only connectivity to the core network segment. This limitation has been overcome in our prototype by extending MaaS. The MaaS Rack Controller in all the edges of the infrastructure and MaaS Controller in the Data Centre can be found in Figure 4.

- **Virtual Infrastructure Manager (VIM)**: The prototype employs all the services related to the cloud controller and the SDN Controller to set up multi-location multi-tenancy scenarios. OpenStack has been used as VIM and OpenDaylight as SDN Controller. It has achieved a complete integration between PIM and VIM in order to enable truly multi-location capabilities in OpenStack. This integration has required the creation of a new component that integrates the different geographical zones managed in MaaS with the different availability zones controlled by OpenStack. See OpenStack Nova Compute in all the edges of the infrastructure and OpenStack Controller and OpenStack Neutron in the Data Centre in Figure 4.
- **Networking and SDN Control Services.** Networking Gateway capabilities and the SDN Controller have been prototyped. Three different prototypes have been implemented based on Neutron, OpenDaylight+Neutron and ONOS+Neutron. Finally, OpenDaylight has been used and recommended due to its maturity with respect to the Neutron northbound interface provided with Service Function Chaining (SFC) capabilities. See OpenDaylight Neutron SDN in Figure 4.
- **Static Service Deployment.** Two different prototypes have been implemented to deploy a VNFM implementation in charge of performing static provisioning of VNFs into a VIM infrastructure with multi-location support. The prototypes are based on OpenMANO and OpenBaton; both have been successfully integrated into OpenStack. At the moment of the prototyping, none of them provides real support to deal with multiple availability zones efficiently. Finally, OpenBaton has been used and recommended due to the architectural design of the software that makes it easier to include the new capabilities into the VIM driver of the OpenBaton architecture. See OpenBaton in the Data Centre and OpenBaton EMS in each of the edges in Figure 4.
- **Dynamic Service Deployment.** The prototype deploys a VNFM implementation to perform dynamic provisioning of the VNFs over both physical and virtual infrastructures[79]. Three different prototypes have been implemented based on Juju, CHEF and Puppet, respectively. Juju offers promising capabilities due to its integration with both OpenStack and MaaS, which allows also the control of the location where the VNF is located. Thus, Juju has been used and recommended due to its natural integration with both MaaS and OpenStack. See Juju in the Data Centre and Juju EMS in each of the edges in Figure 4.
- **4G/5G services.** The prototype deploys an LTE infrastructure by means of the VNFM manager integrated with both VIM and PIM into a mobile edge computing architecture. The infrastructure is fully virtualised and it has been integrated within OpenStack using different virtualization technologies, concretely, KVM and Docker/LXC. The LTE infrastructure has been prototyped using OpenAirInterface VNFs, composed by HSS, MME and SGW/PGW running in the Data Centre. These are using KVM and BBUs operational in each of the edges running in LXCs with hardware devices USBP B210 and Mobile Phones LG directly connected to these LXC containers. See the VNFs for the 4G/5G services in Figure 4.

The prototypes have created a proof-of-concept implementation. Two different versions have been implemented. One version has only one edge location as a first step enabling the consortium to run services. The other version is an extension in which another edge is also made available. Each of the geographical locations comes with compute nodes where VMs can be allocated.

There are other works that investigate automatic process to reduce times from various perspectives. A related scenario is that the deployment time to create new operators needs to be reduced since the Mobile Virtual Network Operators (MVNOs) exist. For instance, Martinez et al. [80] addressed a Multi-Layer (Packet and Optical) Aggregation

Network, and proposed SDN/NFV orchestration to compose isolated backhaul tenants used for different MVNOs. However, no numerical results on time reduction are reported. Moreover, a more relevant aspect is to deploy new network elements/functions in the SDN/NFV context and manage to reduce the deployment time. For example, Katsalis et al. [81] proposed to explore a network slicing architecture for 5G communications, and presented some indicative results on deploying an LTE eNB as a VNF using Juju over a clean installation. However, no complete and integrated solution that is able to reduce the times in all the layers is found in the literature. Controlling and having an absolute control of the life cycle in all the layers is the solution that our work achieves.

5 | TESTBED AND EMPIRICAL RESULTS

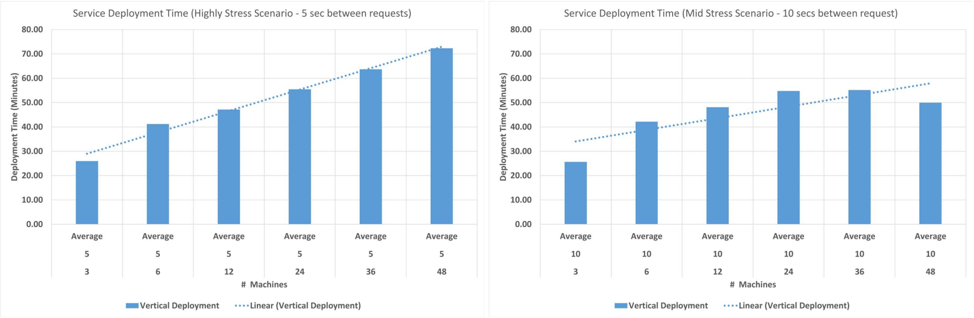
In order to validate the proposed infrastructure against the ambitious 5G KPI related to the fast deployment of 5G services, complete automated orchestration of PIM, VIM and SIM has been implemented to achieve a fully automated deployment of services into virtualised infrastructures from bare metal. To this end, when a new physical machine is attached to our edge, it is detected and automatically configured to enable the power management of this physical machine to gain control over the hardware resources. Then, MaaS is invoked to perform the installation of the operating system and later on Juju is called, in order to install OpenStack compute service on top of such the physical machine to allow VIM to acquire control of the hardware resources. Subsequently, Juju is re-invoked to act now on top of OpenStack to perform the deployment of the 5G virtual services, i.e., OpenAirInterface VNF component, on the new hardware resources controlled by OpenStack. All these steps contribute to the overall service creation (deployment) time of 5G services from bare metal.

The testbed has been created using 8 physical machines as managed computers, each one having 8 cores, 24GB of RAM and 2x1Gbps Ethernet NICs. These machines are managed by a physical machine with an Intel Xeon Processor E5-2630 v4 with 32GB and 1x10Gbps Ethernet NIC acting as a management plane. The purpose of this testbed is to empirically investigate the service deployment time consumed to perform the installation of virtual services for 5G infrastructures from bare metal. To this end, the physical machines have been virtualised to emulate a larger infrastructure with our current physical resources. Consequently, 8 virtual machines have been created on each of the 6 physical computers , resulting in 48 virtualised physical machines in total, by utilizing nested virtualization when the virtual machines are created on top of OpenStack. Since the main purpose is not to optimise the performance of the virtualised service deployed, but to demonstrate the scalability of the proposed architecture, this deployment has allowed us to perform the deployment of a larger number of hardware resources. Thus, the number of virtual machines is ranged from 1 to 48.

The 48 VMs are bare-metal VMs and, at the end, all of them should be controlled by OpenStack, and a virtualised service for monitoring 5G networks should be deployed in each of them. Between the execution of each of the scenarios to be analysed later in this section, a complete clean-up of the VMs was performed in order to allow the execution of the experiment was always from bare metal. For each of the scenarios where a given number of physical machines were provisioned, two different ramping times have been analysed.

The ramping time is defined as the time elapsed between the moment when a new VM is started into the infrastructure and the moment when the previous VM was started. A ramping time of 0 means the parallel starting of all the VMs at the same time. In addition, a realistic high-stress scenario with a ramping time of 5s and a realistic mid-stress scenario with a 10s ramping time were applied, meaning that one new VM is connected to the infrastructure per 5s or 10s, respectively. Subsequently, the VMs were deployed using a vertical orchestration strategy where all the VMs allocated in the same physical machine are deployed before starting the deployment of further VMs in the next one.

It is worth noting the significant time-consuming aspect related to the execution of the experiment, and the service creation times are presented in terms of minutes. 10 different executions of each of the scenarios have been carried out and the averaged times are those plotted in Figure 5. Once the experimentation has been designed, implemented and prototyped, it has required significant execution time (10 executions x 2 experiments x 6 scenarios x 90 minutes approximately) to gather the results presented herein.



(a) Average time for 5s ramping time

(b) Average time for 10s ramping time

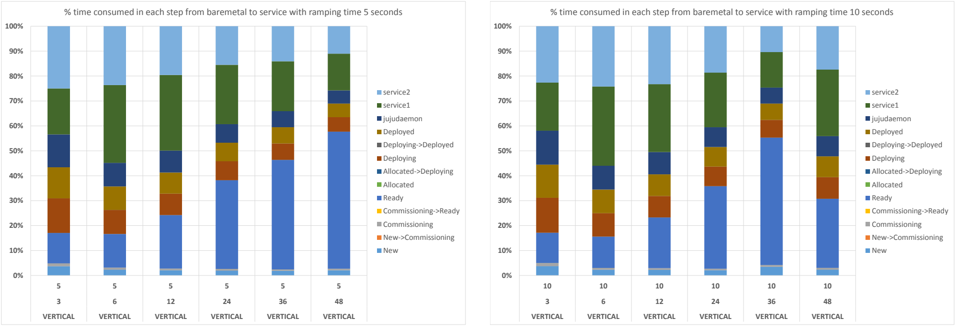
(c) Standard deviation for 5s ramping time

(d) Standard deviation for 10s ramping time

FIGURE 5 5G service creation time from bare metal

Two main experiments have been carried out, where the 5s (left side of Figure 5) and 10s (right side of the figure) ramping times have been tested and analysed, respectively, to investigate the behaviour of the proposed architecture at different stress levels. Figure 5(a) and Figure 5(b) show the average execution time in each experiment, while Figure 5(c) and Figure 5(d) show the standard deviations associated with each experiment. The Y-axis values of the plots represent the minutes to perform the deployment. The starting time is measured when the first network packet is detected in the experiment, usually related to the packet exchange involved in the Preboot Execution Environment (PXE) booting protocol. The ending time is defined as the time when the deployment of the last service of the last VM is completed and the service becomes up and running. The X-axis values of the plots show the different scenarios that were executed in the testbed, specifically, the number of VMs involved in each of the scenarios. The results shown are the average of all the times gathered per physical machine. Both X and Y axes are represented using an exponential distribution and the size of the bars in the figure follows a linear distribution, which is the base to consider the architecture scalable. Furthermore, it shows a slope with an acceptable gradient, which is a significant achievement in the factor of scalability of the proposed architecture. It should be noticed that the standard deviations along all the service deployment times achieved along all the machines involved in each of the experiment also show a very slow gradient, which is a clear sign of the stability and resilience of the proposed infrastructure in the context of scalability in terms of the number of nodes.

Figure 5 also shows that the differences in the service creation times between the different scenarios that involve different ramping times are insignificant. The results demonstrate good responsiveness against high-stress conditions where the ramping time is reduced to half, which is directly related to a double stress condition. Moreover, it should be noticed that, in all the scenarios analysed in terms of size and stress conditions, the service creation times on average and taking into account the standard deviation fulfil the ambitious KPI set by the EU 5G-PPP programme, i.e., creating a new 5G service even from bare metal in less than 90 minutes.



(a) % Percentage time consumed with 5 seconds ramping time (b) % Percentage time consumed with 10 seconds ramping time

FIGURE 6 5G service creation time breakdown

Figure 6 shows the percentage of average time consumed across the different steps involved in the deployment of a service from bare metal. Figure 6(a) and Figure 6(b) provide an analysis of the behaviour in the distribution of the average time consumed for scenarios with ramping time 5 and 10 seconds, respectively. As can be observed, the distribution in times is not affected significantly by the ramping time and moreover it is clear from the analysis of the graphs that the main contribution to the overall times comes from the “Ready” phase. This time represents the time between the moment where the machine is ready to be used and the time where the machine is selected by the Juju scheduler in order to be used as the target for the deployment of the service.

6 | CONCLUSIONS

The proposed 5G Mobile Edge Computing infrastructure has been designed and prototyped in a realistic testbed implementation. Architectural decisions have been taken, wherever appropriate, in order to align the proposed infrastructure with the latest and most innovative trends in the control, management and data planes of softwarised 5G networks. The architecture presented is flexible and extensible, which allows it to cope with the architectural evolutions foreseen from other 5G research activities. Moreover, it is noted that the proposed infrastructure is agnostic to the 5G air interface design, which is an on-going work both within the EU and globally. Comprehensive design considerations for the data, control and management planes have been presented, centred on the Mobile Edge Computing architecture.

This research has employed OpenStack for implementing the Virtual Infrastructure Manager. No existing automation tool has been able to provide a complete deployment of OpenStack integrated with OpenDaylight or any other

SDN Controller, as accomplished in this research. Furthermore, it is worth highlighting that a completely functional LTE infrastructure running in virtual infrastructure has been achieved. Both the real hardware mode using standard mobile phones and an emulation mode using software phones have been enabled to facilitate experimentation over a portable infrastructure. All the software used is based on open source implementations. It is noted that the proposed infrastructure is not constrained to a specific SDN Controller. Two promising SDN Controller candidates that offer SFC capabilities, ONOS and ODL, have been installed and analysed, and a functional demo of SFC capabilities in ODL has been achieved.

It is noted that all the major design aspects proposed in this document have been implemented in the prototype testbed. This research has experimentally achieved noticeable innovation towards a novel 5G infrastructure design and implementation. In particular, significant achievements have been made and empirically tested in the prototype testbed to reduce the services creation time in physical and virtual infrastructures, motivated by meeting the ambitious KPI in substantially reducing service creation time envisioned by the 5G-PPP association and the European Commission. An empirical validation of the achievement of reducing service creation time from 90 hours to 90 minutes has been conducted. The scalability of the architecture and the resilience against the size of the infrastructure has been empirically validated, tested and analysed by means of intensive testing and in all the executions to demonstrate that the concerned 5G-PPP KPI has been achieved through fully automated service deployment introduced by this research.

ACKNOWLEDGMENTS

This work is in part supported by the European Commission Horizon 2020 Programme under grant agreement number H2020-ICT-2014-2/671672 - SELFNET (Framework for Self-Organized Network Management in Virtualized and Software Defined Networks). This work is also supported by the UWS VP Fund for 5G Video Lab.

REFERENCES

[1] Consortium S, 5G-PPP SELFNET Project;. `https://5g-ppp.eu/selfnet/`.

[2] Association GP, 5G-PPP Key Performance Indicators (KPIs);. `https://5g-ppp.eu/kpis/`.

[3] 5G Vision;. `https://5g-ppp.eu/wpcontent/uploads/2015/02/5G-Vision-Brochure-v1.pdf`.

[4] 5G White Paper;. `https://www.ngmn.org/uploads/media/NGMN_5G_White_Paper_V1_0.pdf`.

[5] Mobile Edge Computing;. `http://www.etsi.org/technologiesclusters/technologies/mobile-edge-computing`.

[6] Mobile Edge Computing;. `https://portal.etsi.org/portals/0/tbpages/mec/docs/mobileedge_computing_-_introductory_technical_white_paper_v1%2018-09-14.pdf`.

[7] ETSI NFV;. `http://www.etsi.org/technologiesclusters/technologies/nfv`.

[8] C-RAN - Wikipedia, the free encyclopedia;. `https://en.wikipedia.org/wiki/C-RAN`.

[9] C-RAN - Wikipedia, the free encyclopedia;. `https://5g-ppp.eu/5g-ppp-phase-1-projects/`.

[10] Global 5G Actions;. `https://5g-ppp.eu/global-5g-actions/`.

[11] Garg VK. Wireless Network Evolution: 2G to 3G. 1st ed. Rappaport TS, editor, Upper Saddle River, NJ, USA: Prentice Hall PTR; 2001.

[12] Dahlman E, Parkvall S, Skold J. 4G: LTE/LTE-advanced for mobile broadband. Academic press; 2013.

[13] Amit N, Gordon A, Har'El N, Ben-Yehuda M, Landau A, Schuster A, et al. Bare-metal performance for virtual machines with exitless interrupts. Communications of the ACM 2015;59(1):108–116.

[14] KVM;. `http://www.linux-kvm.org`.

[15] Kivity A, Kamay Y, Laor D, Lublin U, Liguori A. kvm: the Linux virtual machine monitor. In: Proceedings of the Linux symposium, vol. 1; 2007. p. 225–230.

[16] QEMU;. `http://www.qemu.org`.

[17] Bartholomew D. Qemu: a multihost, multitarget emulator. Linux Journal 2006;2006(145):3.

[18] VMWare;. `http://www.vmware.com`.

[19] Rosenblum M. Vmwares virtual platform. In: Proceedings of hot chips, vol. 1999; 1999. p. 185–196.

[20] Linux Containers(LXC);. `https://linuxcontainers.org`.

[21] Virtual Box;. `https://www.virtualbox.org`.

[22] XEN;. `http://www.xenproject.org`.

[23] Alaez RM, Chirivella-Perez E, Calero JMA, Wang Q. New Topology Management Scheme in LTE and 5G Networks. In: 2018 IEEE 87th Vehicular Technology Conference (VTC Spring) IEEE; 2018. p. 1–5.

[24] Sesia S, Baker M, Toufik I. LTE-the UMTS long term evolution: from theory to practice. John Wiley & Sons; 2011.

[25] ElNashar A, El-Saidny MA, Sherif M. Design, deployment and performance of 4G-LTE networks: A practical approach. John Wiley & Sons; 2014.

[26] Baldo N, The ns-3 LTE module by the LENA project; 2011.

[27] Henderson TR, Lacage M, Riley GF, Dowell C, Kopena J. Network simulations with the ns-3 simulator. SIGCOMM demonstration 2008;14(14):527.

[28] Nikaein N, Marina MK, Manickam S, Dawson A, Knopp R, Bonnet C. OpenAirInterface: A flexible platform for 5G research. ACM SIGCOMM Computer Communication Review 2014;44(5):33–38.

[29] SP-030388 Network Sharing Cover Sheet - 3GPP;. `ftp://www.3gpp.org/tsg_sa/TSG_SA/TSGS_21/Docs/PDF/SP-030388.pdf`.

[30] 3GPPP;. `http://www.3gpp.org/`.

[31] Network Functions Virtualisation(NFV) Management and Orchestration;. `http://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf`.

[32] Mijumbi R, Serrat J, Gorricho JL, Latré S, Charalambides M, Lopez D. Management and orchestration challenges in network functions virtualization. IEEE Communications Magazine 2016;54(1):98–105.

[33] Metal As A Service;. `http://maas.io`.

[34] Rocks;. `http://www.rocksclusters.org/rocks-documentation/4.1/getting-started.html`.

[35] HP Insight Cluster Management Utility| Hewlett Packard;. `http://www8.hp.com/us/en/products/server-software/product-detail.html?oid=3296361`.

[36] Chirivella-Perez E, Alaez RM, Calero JMA, Wang Q, Gutierrez-Aguado J. UWSIO: Towards automatic orchestration for the deployment of 5G monitoring services from bare metal. In: Wireless Communications and Networking Conference (WCNC), 2018 IEEE IEEE; 2018. p. 1–6.

[37] Open Energy Monitor;. `https://openenergymonitor.org/emon/`.

[38] Arduino;. `https://www.arduino.cc/`.

[39] Smart Energy Groups | energy awareness;. `https://smartenergygroups.com/`.

[40] eNOS - A Cloud Based, Open Source Energy Management;. `http://www.openenos.com/`.

[41] Ghost;. `https://www.symantec.com/en/uk/ghost-solution-suite/`.

[42] Clonezilla;. `http://clonezilla.org/`.

[43] PartImage;. `https://www.partimage.org/Main_Page`.

[44] Fog;. `https://fogproject.org/`.

[45] Chapter 5. Virtual storage - VirtualBox.;. `https://www.virtualbox.org/manual/ch05.html`.

[46] McLoughlin M, The QCOW2 image format; 2008.

[47] Puppet;. `https://puppetlabs.com/`.

[48] CHEF;. `https://www.chef.io/chef/`.

[49] Marschall M. Chef infrastructure automation cookbook. Packt Publishing Ltd; 2015.

[50] SmartFrog;. `http://smartfrog.org/display/sf/SmartFrog+Home`.

[51] Goldsack P, Guijarro J, Loughran S, Coles A, Farrell A, Lain A, et al. The SmartFrog configuration management framework. ACM SIGOPS Operating Systems Review 2009;43(1):16–25.

[52] Juju;. `https://jujucharms.com/`.

[53] Ansible;. `https://www.ansible.com/application-deployment`.

[54] Hochstein L, Moser R. Ansible: Up and Running: Automating Configuration Management and Deployment the Easy Way. " O'Reilly Media, Inc."; 2017.

[55] Chirivella-Perez E, Gutiérrez-Aguado J, Alcaraz-Calero JM, Wang Q. NFVMon: Enabling Multioperator Flow Monitoring in 5G Mobile Edge Computing. Wireless Communications and Mobile Computing 2018;2018.

[56] Infrastructure As A Service;. `https://en.wikipedia.org/wiki/Cloud_computing#Infrastructure_as_a_service_.28IaaS.29`.

[57] Bhardwaj S, Jain L, Jain S. Cloud computing: A study of infrastructure as a service (IAAS). International Journal of engineering and information Technology 2010;2(1):60–63.

[58] Sefraoui O, Aissaoui M, Eleuldj M. OpenStack: toward an open-source solution for cloud computing. International Journal of Computer Applications 2012;55(3).

[59] Kumar R, Jain K, Maharwal H, Jain N, Dadhich A. Apache cloudstack: Open source infrastructure as a service cloud computing platform. Proceedings of the International Journal of advancement in Engineering technology, Management and Applied Science 2014;p. 111–116.

[60] Milojičić D, Llorente IM, Montero RS. Opennebula: A cloud management tool. IEEE Internet Computing 2011;15(2):11–14.

[61] Medved J, Varga R, Tkacik A, Gray K. Opendaylight: Towards a model-driven sdn controller architecture. In: World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2014 IEEE 15th International Symposium on a IEEE; 2014. p. 1–6.

[62] Berde P, Gerola M, Hart J, Higuchi Y, Kobayashi M, Koide T, et al. ONOS: towards an open, distributed SDN OS. In: Proceedings of the third workshop on Hot topics in software defined networking ACM; 2014. p. 1–6.

[63] Ryu S, Framework Community: Ryu SDN Framework; 2015.

[64] Wallner R, Cannistra R. An SDN approach: quality of service using big switch's floodlight open-source controller. Proceedings of the Asia-Pacific Advanced Network 2013;35:14–19.

[65] Frantz PJ, Thompson GO, VLAN frame format. Google Patents; 1999. US Patent 5,959,990.

[66] Virtual eXtensible Local Area Network;. `https://en.wikipedia.org/wiki/Virtual_Extensible_LAN`.

[67] Moy J. RFC 7348: Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. Tech Rep 2014;.

[68] Generic Routing Encapsulation;. `https://en.wikipedia.org/wiki/Generic_Routing_Encapsulation`.

[69] Hanks S, Meyer D, Farinacci D, Traina P. Generic routing encapsulation (GRE) 2000;.

[70] Virtual Tenant Networks;. `http://docs.opendaylight.org/en/stable-oxygen/user-guide/virtual-tenant-network-(vtn).html`.

[71] Service Function Chainning;. `https://wiki.opnfv.org/display/sfc/Service+Function+Chaining+Home`.

[72] Semeria C, Engineer M. Multiprotocol label switching. Enhancing Routing in the New Public Network, White paper, Juniper Networks (March 1999) 2000;.

[73] MultiProtocol Label Switching;. `https://en.wikipedia.org/wiki/Multiprotocol_Label_Switching`.

[74] Bhamare D, Jain R, Samaka M, Erbad A. A survey on service function chaining. Journal of Network and Computer Applications 2016;75:138–155.

[75] Pignataro C, Halpern J. Service Function Chaining (SFC) Architecture 2015;.

[76] Openstack Neutron;. `https://wiki.openstack.org/wiki/Neutron`.

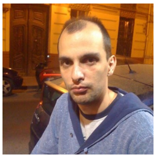
[77] Tkachova O, Salim MJ, Yahya AR. An analysis of SDN-OpenStack integration. In: Problems of Infocommunications Science and Technology (PIC S&T), 2015 Second International Scientific-Practical Conference IEEE; 2015. p. 60–62.

[78] D2.1 Use Cases Definition and Requirements of the System and its Components;. `https://bscw.selfnet-5g.eu/pub/bscw.cgi/d18783/SELFNET%20Deliverable%202.1%20-%20Final%20v12.pdf`.

[79] Pérez MG, Celdrán AH, Ippoliti F, Giardina PG, Bernini G, Alaez RM, et al. Dynamic reconfiguration in 5G mobile networks to proactively detect and mitigate botnets. IEEE Internet Computing 2017;21(5):28–36.

[80] Martínez R, Vilalta R, Requena M, Casellas R, Muñoz R, Mangues J. Experimental SDN control solutions for automatic operations and management of 5G services in a fixed mobile converged packet-optical network. In: 2018 International Conference on Optical Network Design and Modeling (ONDM) IEEE; 2018. p. 1–10.

[81] Katsalis K, Nikaein N, Schiller E, Ksentini A, Braun T. Network slices toward 5G communications: Slicing the LTE network. IEEE Communications Magazine 2017;55(8):146–154.



ENRIQUE CHIRIVELLA-PEREZ (MEng, BEng, MIEEE) is PhD Candidate in the School of Computing, Engineering and Physical Sciences at the University of the West of Scotland, United Kingdom. He is currently involved in the H2020 5G-PPP Phase 1 SELFNET project. His main interest includes infrastructure monitoring, network management, automatic deployment of services and software-defined networks in mobile edge computing and 5G networks.

RICARDO MARCO-ALAEZ (Phd, MIEEE) is a Research Assistant in the School of Computing, Engineering and Physical Sciences at the University of the West of Scotland, United Kingdom, where he's involved in the H2020 5G-PPP Phase 1 SELFNET project. His main interests include network management, mobile networks in cloud computing and 5G networks. Ricardo Marco has a PhD in computer science from the University of the West of Scotland.

ALBA HITA obtained her degree in Computer Engineering from the University of Murcia in 2017. She is currently a PhD Candidate in the School of Computing, Engineering and Physical Sciences at the University of the West of Scotland. Her main interest are: scheduling, energy monitoring system, cloud computing in 5G infrastructures.

ANA SERRANO MAMOLAR is a PhD candidate in the School of Computing, Engineering and Physical Sciences at the University of the West of Scotland, United Kingdom, where she is involved in the H2020 5G-PPP Phase I SELFNET project. Her main interests include network management, cognitive control plane, cyber-security and self-protection in mobile edge computing and 5G networks.

JOSE M. ALCARAZ CALERO (PhD, MEng, BEng, SMIEEE, FHEA), is Full Professor in Networks and Security at School of Computing, Engineering and Physical Sciences, University of the West of Scotland, UK. He is Board Member of the EU 5G-PPP Technology Board and Technical Co-manager of the H2020 5G-PPP Phase 1 – SELFNET Consortium (6.8m EUR) and the the H2020 5G-PPP Phase 2 – SLICENET Consortium (8m EUR). His main research interest include Network Control and Network Management, Service Automation, 5G, Mobile Edge Computing. He has published more than 100 peer-reviewed papers in these areas.

QI WANG is a Full Professor in the School of Computing, Engineering and Physical Sciences at the

University of the West of Scotland (UWS), UK. He is a Board Member of EU 5G PPP Technology Board. He is the Technical Co-Manager for EU Horizon 2020 5G PPP projects SELFNET and SliceNet and Principal Investigator or Co-Investigator for numerous projects sponsored by UK EPSRC etc. His research interests include new-generation mobile networks, and video networking and processing. He has published more than 120 peer-reviewed papers in these areas.

PEDRO M. NEVES Received his M.Sc. and Ph.D. degrees in Electronics and Telecommunications Engineering from the University of Aveiro, Portugal, in 2006 and 2012 respectively. After graduation, he became research fellow of the Telecommunications Institute, where he worked on European funded projects on broadband wireless access networks. In June 2006 he joined PT Inovação, working on heterogeneous wireless environments in the context of European and Eurescom funded projects. He participated in more than 10 international collaborative projects, is co-author of six international books, and has published more than 30 articles in journals and conference proceedings.

GIACOMO BERNINI received his Laurea degree in telecommunication engineering from the University of Pisa, Italy, in 2006. He is an R&D project manager at Nextworks where his research focuses on Software Defined Networking, Network Functions Virtualization,5G Networks and Services, Network Slicing Open Source initiatives. His activity spans from Nextworks industrial activities to EU projects. He is currently active in the FP7 and H2020 projects.

KONSTANTINOS KOUTSOPOULOS has been working for the last 20 years as system designer and software developer in the field of research and commercial projects around applications for personal and mobile communications, embedded platforms and residential gateways, communication protocols, M2M methodologies and software orchestration/provisioning techniques. His current interests are domain specific languages and information models processing, component/platform automation.

MANUEL GIL PÉREZ is Research Associate in the Department of Information and Communication Engineering of the University of Murcia, Spain. His research primarily focuses on cybersecurity, including intrusion detection systems, trust management, privacy-preserving data sharing, and security operations in highly dynamic scenarios. He received a Ph.D. degree (Hons) in computer science from the University of Murcia, being (co-)author of 60+ scientific publications in journals and conference papers.

GREGORIO MARTÍNEZ PÉREZ is Full Professor in the Department of Information and Communications Engineering of the University of Murcia, Spain. His scientific activity is mainly devoted to cybersecurity and networking, also working on the design and autonomic monitoring of real-time and critical applications and systems. He is working on different national (14 in the last decade) and European IST research projects (11 in the last decade) related to these topics, being Principal Investigator in most of them. He has published 160+ papers in national and international conference proceedings, magazines and journals.

MARIA JOÃO BARROS are currently coordinating the 5G PPP projects SELFNET- Framework for Self-organized Network Management in Virtualized and Software Defined Networks project (https://selfnet-5g.eu/) and SLICENET End-to-End Cognitive Network Slicing and Slice Management Framework in Virtualised Multi-Domain, Multi-Tenant 5G Networks (https://slicenet.eu). Programme Manager, responsible for International projects Coordination, management roles, project work and representation, and proposals elaboration. with expertise in communication networks. With expertise in communication networks, I've been involved in different ICT areas, like 5G, Smart cities, Internet-of-Things (IoT), Future Internet (FI), and Artificial Intelligence (AI); focusing on topics like Network Management, Network Slicing, Machine-learning, Human-Machine interfaces, and Language Technologies.

ANASTASIUS GAVRAS has more than 25 years of professional experience in academic and industry research in network and service management, security and middleware. His current interests are large scale testbeds for 5G network infrastructure and Future Internet research and experimentation. He is involved in several 5G initiatives with a focus on the requirements of the vertical industry sectors on 5G. He is editors of several books and author of several papers and articles in the area.