Vision on Software Networks and 5G

SN WG – January 2017

(final version 1.9)

Table of Contents

1 Introduction

5G PPP vision is that in a decade from now, telecom and IT will be integrated towards a common very high capacity ubiquitous infrastructure, with converging capabilities for both fixed and mobile accesses [1]. 5G will not only be an evolution of mobile broadband networks but will bring new unique network and service capabilities. 5G will integrate networking, computing and storage resources into one programmable and unified infrastructure. This unification will allow for an optimized and more dynamic usage of all distributed resources, and the convergence of fixed, mobile and broadcast services.

The 5G PPP European research programme is structured in three phases running and linking to mobile industry 3GPP specification process and other SDO like ETSI. Their proposed specifications are finally submitted to the International Telecommunications Union (ITU) that releases the final standard.

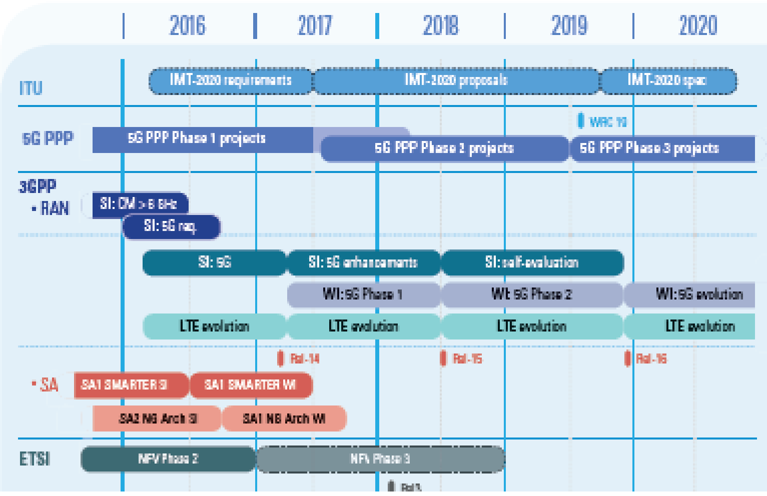

Figure 1: 5G PPP general phases and SDO timing

Nineteen projects of first phase of 5G PPP were selected in 2015 and are currently working under same collaboration umbrella agreement addressing multiple technological and business angles of 5G networks evolution [2].

In the 2nd phase of 5G PPP projects starting in 2017 there is a special focus on vertical industrial involvement for enabling 5G scenarios. This is a crucial step towards the 5G trial programme phase 3 as indicated in the 5G manifesto and the announced 5G Action Plan for Europe September 2016 [3]. This will lead to the EU trial roadmap for implementation with end-to-end 5G platforms from 2018 and beyond integrating multiple technologies that will be operated in a pre-competitive environment showcasing several vertical-sector cases (eHealth, Manufacturing, Media, Automotive, etc)

Software Network Working Group (WG) of 5G PPP aims at analysing and addressing unification and applicability of key research topics related to Software Networking including software defined concepts, infrastructures, systems and components for Wire and- Wireless Networks, including Networked Clouds, IoT and Services, integrated SDN/NFV vision as developed and promoted by the 5G PPP projects and also aligned with the Architecture WG [4]. Software Networks is a significant area of research and innovation. Software Defined Networks (SDN) and Network Function Virtualization (NFV) are essential to support many aspects of the anticipated functionality offered by next generation 5G Networks in 2020+. This is a technical paper that gathers the contribution of projects to the specific point of combining SDN and NFV technologies. The contributing projects have been; 5GEx, 5G-Crosshaul, 5G-Xhaul, CHARISMA, COHERENT, ENSURE, SELFNET, SESAME, SONATA, SUPERFLUIDITY, VIRTUWIND.

2 NFV and SDN and 5G

Telecommunication networks have become a critical infrastructure for any society enabling economic growth and social prosperity. The services supported by today's telecommunication networks are relied upon by millions of people every day. These networks and services continue to evolve, supporting ever increasing workloads and increasing traffic and diversity of supported services. This evolution is forcing the underlying network technologies to change, increasing the level of programmability, control and flexibility of configuration, while reducing the overall costs related to network operations.

5G will not only be an evolution of mobile broadband networks but the infrastructure to bring unique service capabilities. Aligned with the 5G vision, this goes beyond a simple increase in network speed or reliability, and instead it represents a substantive shift in telecommunication technology, where the network offers unique features to each of the evolving services it supports.

SDN and NFV are core technologies in the transformation of networks for 5G. SDN and NFV could be seen as different expressions of an overall transformation trend, which is deeply impacting Telecom and IT industries. This trend is transforming several Industries, in using ”softwarization" on general computing platforms to optimize operational processes and in bringing rapidly and efficiently new values in infrastructures. Their introduction aims to lower the cost of network and service operation and to reduce the time to market for new services while introducing higher flexibility. In addition virtualization of networking systems is a key enabler that offers a multitude of benefits for telecommunication and datacenter operators by decoupling network functions from proprietary hardware as well as decoupling services from propriety service platforms. This is very disruptive to today's telecom value chain and this evolution is familiar: we saw this happen in software and IT services with the rise of cloud computing. The expression “Software Networks” is often referred to a general paradigm shift in telecom architecture from “boxes” to “functions”, and from “protocols” to “APIs”. In parallel this shift is also driving a convergence between telecommunications and IT infrastructure, producing an IT solution which delivers carrier grade platform upon which 5G “Software Networks” are implemented.

In this sense, SDN and NFV are major transformational technologies for operators to split network capacity and elements into virtual slices that can be then used for multiple applications. That includes broadband video cases which may require high throughput speeds of 10 Gb/s at mobile devices as well as lower bandwidth applications but critical such to connect a huge number of Internet of Things (IoT) elements and devices that are less demanding on the network [6]. The 5G network will have to be extremely efficient in its low-bandwidth transmissions and have enhanced coverage.

The initial value proposition associated to the introduction of SDN and NFV was aimed to increase network configurability and lower the operational network costs and necessary infrastructure investments, decoupling network functions from specialized hardware so that the functions are offered as virtualized software elements. Today the SDN/NFV value proposition has shifted form ‘cost savings’ to enable features like flexible services, multi-tenancy, etc. that opens up new business opportunities and reduces the time to market for these new services.

3 Terminology

Many different terms are currently being used in the area of software networks. In order to align a bit the terminology (since it is not always the case that different actors understand the same from a given term), we provide next an agreed view of the most widely used terms:

Abstraction: a representation of an entity in terms of selected characteristics. An abstraction hides or summarizes characteristics irrelevant to the selection criteria.

Application Plane (AP): the collection of applications and services that program network behaviour.

Control Plane (CP): the collection of functions responsible for controlling one or more network devices. CP instructs network devices with respect to how to process and forward packets. The control plane interacts primarily with the forwarding plane and, to a lesser extent, with the management plane.

Forwarding Plane (FP): the collection of resources across all network devices responsible for forwarding traffic.

Management Plane (MP): the collection of functions responsible for monitoring, configuring, and maintaining one or more network devices or parts of network devices.

Network Function (NF): A functional building block within a network infrastructure, which has well-defined external interfaces and a well-defined functional behaviour. In practical terms, a Network Function is today a network node or physical appliance.

NF Forwarding Graph: a graph of logical links connecting NF nodes for the purpose of describing traffic flow between these network functions.

NF Set: a collection of NFs with unspecified connectivity between them.

NFV Infrastructure (NFVI): the totality of all hardware and software components (e.g. hypervisor, excluding VM/VNF according to ETSI) which build up the environment in which VNF are deployed. The NFV Infrastructure can span across several locations, i.e., multiple Network PoPs. The network providing connectivity between these locations is regarded to be part of the NFV Infrastructure or can be controlled together.

NFV Management and Orchestration (NFV-MANO): functions collectively provided by NFVO, VNFM, and VIM.

NFV Orchestrator (NFVO): functional block that manages the Network Service (NS) lifecycle and coordinates the management of NS lifecycle, VNF lifecycle (supported by the VNFM) and NFVI resources (supported by the VIM) to ensure an optimized allocation of the necessary resources and connectivity. The NFV Orchestrator is in charge of the network wide orchestration and management of NFV (infrastructure and software) resources, and realizing NFV service topology on the NFVI. The NFV Orchestrator manages and automates the distributed NFV Infrastructure. The NFV Orchestrator has control and visibility of all VNFs running inside the NFVI. The NFV Orchestrator provides GUI and external NFV-Interfaces to the outside world (i.e. BSS/OSS). The NFV Orchestrator makes extensive use of the NFV lower stack to perform cloud operation and automation, thus extend the basic capabilities of the NFV-Operating System and provides much richer environment.

Orchestration: describes the automated arrangement, coordination, and management of complex systems, middleware and services.

Resources: physical or virtual compute/storage/network element with some attributes (e.g. cpu#, storage size) and connection points (basically network connectivity ports: e.g. a server has 2 NICs). A Resource Node/Element is a set of resources encapsulated so that they offer externally only a limited set of resource attributes and connection points; presented as a single controllable resource entity (node), e.g., a Compute Node or a Network Element.

Service: instantiation of Service Template, will define all still undefined attribute values and connection points, and will allocate resources and deploy SW.

Service Function (SF): a function that is responsible for specific treatment of received packets. A Service Function can act at the network layer or other OSI layers. A Service Function can be a virtual instance or be embedded in a physical network element. One or multiple Service Functions can be embedded in the same network element. Multiple instances of the Service Function can be enabled in the same administrative domain.

Service Template: pre-packaged reusable service definition that a Multi-domain orchestrator advertises to other Network operators and customers. Each Service Template will have attached a descriptor that defines its functioning, architecture, SLA, pricing, the NFs (Network functions) that compose it with reference to the NFIB (Network Function Information Base), along with a forwarding graph and deployment instructions.

Slice: a network of computing, communication, (and measurement) resources capable of running resource elements (Network/Service Functions) or a network service. In the data plane a slice is defined by a set of resource containers spanning a set of network components, plus an associated set of service access points (users) that are allowed to access those resource containers. A slice can be pre allocated or created on demand. In the control plane (from an operator's perspective), slices are the primary abstraction for accounting and accountability—resources are acquired via a slice control and management interface (API).

Virtualisation: an abstraction whose selection criterion is dedicated to a particular consumer, client or application.

Virtualised Network Function (VNF): an implementation of a Network Function that can be deployed on a Network Function Virtualisation Infrastructure (NFVI).

Virtualised Infrastructure Manager (VIM): functional block that is responsible for controlling and managing the NFVI compute, storage and network resources, usually within one operator's Infrastructure Domain.

Virtualised Network Function Manager (VNFM): functional block that is responsible for the lifecycle management of VNF.

VNF Forwarding Graph (VNF FG): a NF forwarding graph where at least one node is a VNF.

4 Combining NFV and SDN: architectural trends and existing gaps

4.1 Introduction: early steps combining NFV and SDN

SDN and NFV are seen as the critical architectural building blocks to give mobile operators the necessary tools to address the massive data usage by their customers in the wave of expected 5G use cases. The software focus driving the confluence of NFV and SDN will allow mobile operators to be more responsive to customer demands either dynamically or through on-demand service provisioning via self-service portals.

NFV and SDN were developed by different standardization bodies; however, they are complementary technologies and as a result are often collectively referred as NFV/SDN. Although they have substantial value when exploited separately, in combined they offer significant additional value. The NFV technology brings the possibility of creating new network functions on-demand, placing them on the most suitable location and using the most appropriate amount of resources. It also provides the necessary management (orchestration) elements to deal with the virtualised elements. To take the most of this virtualisation, the SDN is able to adjust the network accordingly, enabling network programmability and sequencing of functions (chaining VNF with SDN flows).

Research and development of SDN and NFV technologies has been a hot topic lately (almost all 5G-PPP projects, if not all, are looking to some extent into one or the other at least). A plethora of different mechanisms have been and are being proposed. Similarly, but just lately, the integration of SDN and NFV has received some attention. We can find one example of this on the standardization side (more details on standardization activities can be found in Section 5). As part of Phase 2 of the ETSI NFV ISG, GS NFV-EVE 005 [8] compiled a "Report on SDN Usage in NFV Architectural Framework". The document provides an overview of SDN in relation to the ETSI NFV architectural framework as well as a summary of current industry work including a comparison of network controllers and PoCs relevant to NFV and SDN. Building on the foundation of ITU-T Y.3300 [9], IETF RFC 7426 [10] and ONF TR-502 [1], the ETSI report above covers:

- An overview of SDN in the NFV architectural framework (section 4 of [8]).
- Design patterns of SDN in the NFV architectural framework (section 5 of [8]).
- Functional recommendations for the SDN-NFV integration (section 6 of [8]).

As specified in the report, within the NFV architectural framework, SDN solutions might be used in the infrastructure domain, in the tenant domain, or both.

The first type of connectivity services are those supported by the NFVI to enable communication among VNFs and among their components (VNFC – it is possible to have a VNF consists of more than one component), in a single PoP or in a more general way even for a case where VNFs are instantiated in separated PoPs, reachable through a WAN connection. There is a clear understanding that SDN plays a key role to support the requirements on elasticity and virtualisation for the infrastructural network to support the VNFs. This is the role of the *infrastructure controller*, managed by the VIM.

The second deals with the network services provided at the service tenant layer, and has to deal with the operation and management of the network service's constituent VNFs, and includes whatever semantics are related to the network service and that might be controlled by means of the SDN paradigm, through a programmatic, logically centralized control of the data forwarding plane. The network services at this layer are suitable to be managed according to the SDN paradigm, and that is the proposed role of the *tenant controller*.

Out of the various locations of SDN resources, SDN controller(s) and SDN applications explored in the report, the illustration of Figure 2 consolidates and summarizes the majority of the scenarios. The particular illustration is also aligned with the SDN controller interfaces:

- Application Control Interface - interface between an SDN controller and an SDN application - provides an application programmatic control of abstracted network resources.
- Resource Control Interface - Interface between an SDN controller and SDN resources - it is used to control network resources (e.g. OpenFlow).
- Orchestration Interface - interface between an SDN controller and an NFV Orchestrator.
- Controller-Controller Interface - interface between SDN controllers.

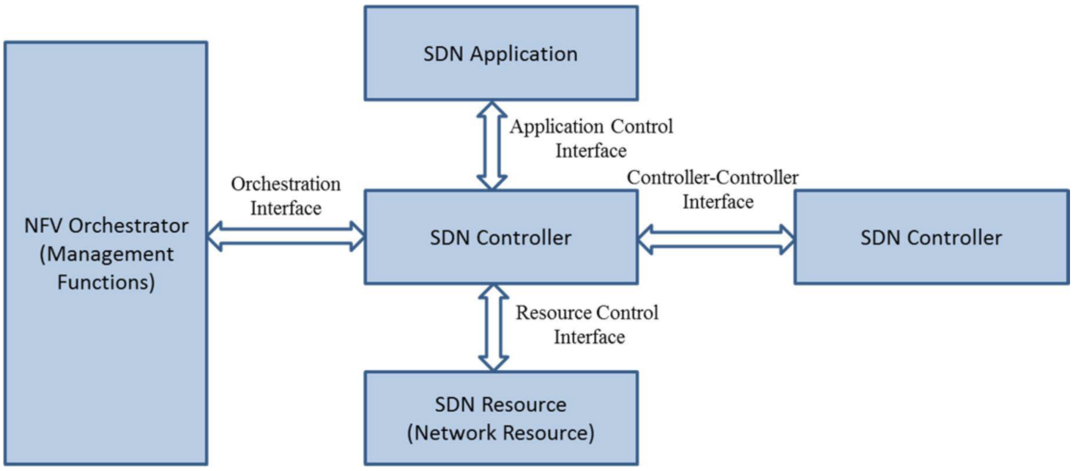
SDN Application
Application Control Interface
Orchestration Interface
Controller-Controller Interface
NFV Orchestrator (Management Functions)
SDN Controller
SDN Controller
Resource Control Interface
SDN Resource (Network Resource)

Figure 2: SDN Controller Interfaces (section 4.4.1 of [8])

In terms of design patterns, ETSI envisions that SDN can be used to fulfil broad use cases for:

- Interconnecting VNF components (VNFCs).
- Interconnecting VNFs towards implementing service chaining or load balancing.
- Interconnecting VNFs across multiple VIMs, in the same or different NFVI-PoPs.

The use cases above apparently also expand to establishing hierarchies of SDN controllers:

- for distributed performance, scalability and reliability
- for distributed, cross-service provider or cross-domain services
- for implementing NaaS within a single service provider
- for fault management in multi-layer transport network

Last but not least, ETSI recognizes the role of SDN in creating network connection topologies, (VNF) forwarding graphs and service chains (SFC).

Finally, section 6 of [8] provides an extensive categorized of functional recommendations. The combined NFV and SDN architecture that we propose needs to be validated against that list.

4.2 An architecture proposal to integrate NFV and SDN for 5G

5G mobile networks will necessarily exploit the latest advances in software networking technologies, following a standard-compliant, layered approach. As described in the 5G-PPP Architecture WG white paper released in July 2016 [11], different planes are foreseen, each having different concerns: Application and Business Service Plane, Multi-Service Management Plane, Integrated Network, Management & Operations Plane, Infrastructure Softwarisation Plane, Control Plane and Forwarding/Data Plane. Taking this into consideration, as well as the standardisation progress in related bodies such as ETSI NFV and ONF, we describe next a first conceptual architecture seamlessly and flexibly combining SDN and NFV technologies. Figure 3 illustrates the overall architecture of integrating NFV and SDN, which joins the perspectives of both NFV and SDN communities, as explained below.

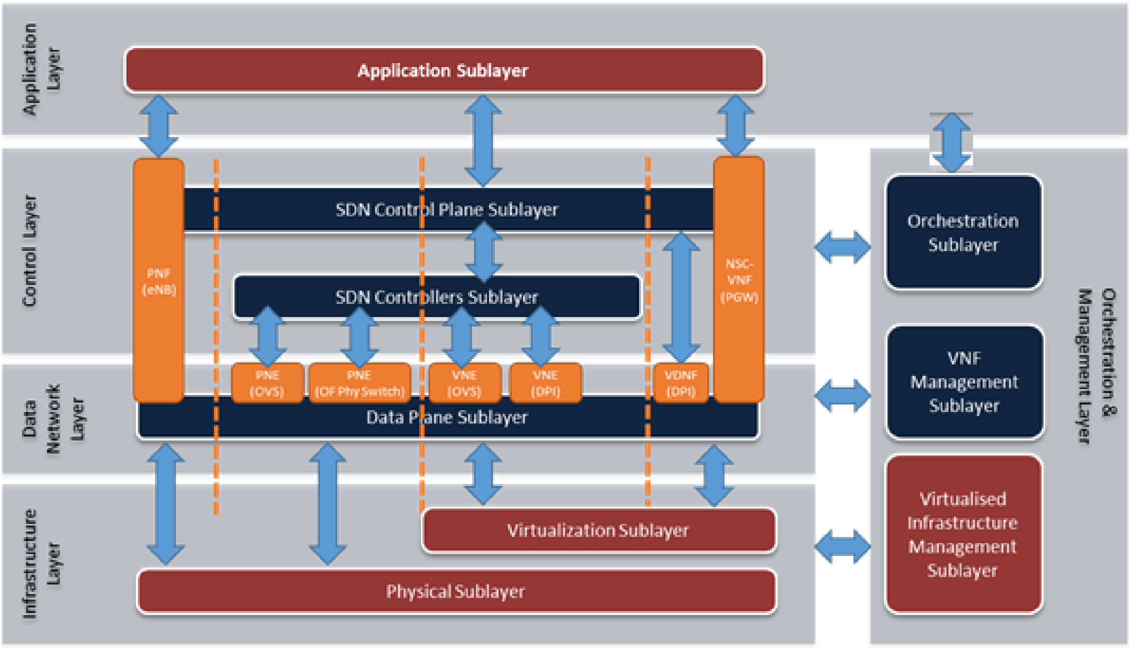

Figure 3: Overall architecture of SDN/NFV integration and management

To specify the integration and interactions between SDN and NFV, it is necessary to clarify the various types of network functions. To this end, network functions can be classified based on the different levels of their compatibility with either NFV or SDN implementations. For backwards-compatibility of legacy network functions, Physical Network Functions (PNFs, e.g., 4G/5G base stations such as eNodeBs) and Physical Network Elements (PNEs, e.g., physical switches based on Open vSwitches/OVS or OpenFlow) are those that cannot be virtualised, although the latter can be possibly controlled with SDN. Note that there are VNFs– referred to as Non-SDN-enabled Virtual Network Functions, NSC-VNFs, in the figure – that cannot be controlled with SDN, even though they are virtualized. In addition, Data Network Functions (DNFs, e.g., virtual DPIs without the control plane) refer to those VNFs that contain data plane functions only, following a separation of the control and data planes in certain SDN implementations.

Architecturally, from the bottom up, the Infrastructure Layer provisions both physical and virtualized resources and the operation environment for the various VNFs. The Control Layer is broken down to a SDN Controller sublayer and a SDN Control Plane sublayer on top of it. These two sublayers and the underlying Data Plane sublayer can be conceptually mapped to the

ONF's three-layer SDN model. Specifically, the Data Plane sublayer is largely controlled by the SDN Controllers sublayer, where a pool of SDN Controllers may be deployed for scalability and wide-area deployment considerations. The architectural organization and physical locations of these SDN Controllers should be optimized to allow high cost-efficiency and high performance of running the dynamically deployed VNFs on demand. On top of the SDN Controllers, various SDN Applications (Apps) in the SDN Control Plane provide the control logic to the SDN Controllers, based on various business application requirements from the upper Application Layer. In parallel to this stack, a vertical Orchestration and Management Layer manages the Infrastructure, Data Network and Control layers through the Virtualized Infrastructure Management, VNF Management and Orchestration sublayers. This architecture is compliant with the ETSI NFV's Management and Orchestration (MANO) framework.

This architectural proposal represents an initial step towards effectively integrating SDN/NFV. Next section goes into several examples where software networks (will) play a significant role, identifying specific SDN/NFV integration aspects that have not yet properly addressed.

4.3 Some examples of software network features and scenarios

4.3.1 Cloud RAN

The *Cloud Radio Access Network* (C-RAN) is a mobile network architecture allowing operators to meet the needs of increasingly growing user demands and 5G requirements for achieving the performance needs of future applications. C-RAN intends to move part of the access infrastructure (e.g., the eNB) to local datacentres, taking advantage of the cloud technologies, elastic scale, fault tolerance, and automation. Using this architecture, a minimum set of hardware critical functions is kept on the physical *Remote Radio Head* (RRH) components, while the remaining virtualized (CPU intensive) part, the *Base Band Unit* (BBU), is deployed and concentrated in small local datacentres for a large number of cells. Figure 4 depicts the basic concept.

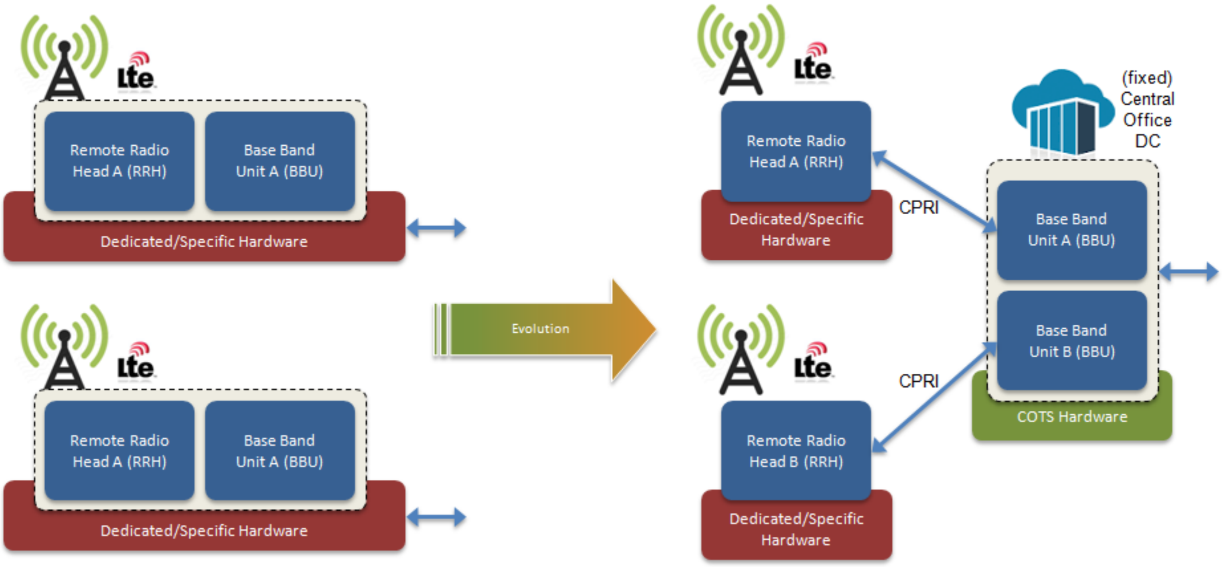
lte
Remote Radio Head A (RRH)
Base Band Unit A (BBU)
Dedicated/Specific Hardware
lte
Remote Radio Head A (RRH)
Base Band Unit A (BBU)
Dedicated/Specific Hardware
Evolution
lte
Remote Radio Head A (RRH)
Dedicated/Specific Hardware
CPRI
(fixed) Central Office DC
Base Band Unit A (BBU)
Base Band Unit B (BBU)
COTS Hardware
CPRI
lte
Remote Radio Head B (RRH)
Dedicated/Specific Hardware

Figure 4: 3GPP C-RAN basic model.

This approach has multiple advantages: 1) it brings resource isolation for C-RAN functions that are now software defined and can be deployed and scale using commodity hardware and off-the-shelf standards components; 2) it reduces the cost per deployed cell, which is an important aspect, as next mobile generations will reduce the cell coverage (and therefore increase the number of cells) as a way to increase bandwidth; 3) by centralizing BBUs in a local DC, the required amount of resources is lower than for traditional scenarios, as today cell-sites are provided with processing power for peak hours; 4) X2 interfaces (i.e., the one defined among eNBs) become internal links in top of the rack DC network, reducing latency times on mobility situations; 5) it leverages C-RAN SDN controllers [27] to improve network capacity and flexibility by providing dynamic control, fine grained resource allocation and better latency control to meet QoS requirements.

4.3.1.1 Impact on SDN/NFV integration

There are different C-RAN solutions, which propose different functional splits between the cell-site part (RRH) and the DC part (BBU). This has impacts on the interface between them and the amount of bandwidth required for that. For this reason, this issue is of extreme importance.

In terms of NFV and SDN integration, C-RAN solutions require proper interaction between the MANO layer, to roll out virtual BBUs where needed and when needed, and the SDN control layer, to ensure that these BBUs and the RRHs are connected meeting the actual requirements.

5G PPP Projects such as 5G Xhaul, CHARISMA and 5G-Crosshaul are contributing to these topics.

4.3.2 Mobile Edge Computing

Mobile Edge Computing (MEC) is a new technology platform, which is currently being standardized by an ETSI Industry Standard Group (ISG). MEC will offer application developers and content providers a cloud computing and IT service environment to deploy and run applications at the edge of a mobile network. MEC environments are characterized by:

- Proximity
- Ultra low latency
- High bandwidth
- Real-time access to radio network information
- Location awareness

MEC provides a set of services that can be consumed by applications running on the MEC platform. These services can expose information such as real time network data e.g., radio conditions, network statistics etc. and the location of connected devices to consuming applications. Exposing these types of information can be used to create context for applications and services with the potential to enable a new ecosystem of use cases, such as location services, big data, multimedia services and many more, with a significantly improved user experience.

MEC can take advantage of NFV deployments, due to the complimentary use of cloud infrastructure, virtualization layers and management and orchestration components. Operators are increasingly looking at virtualization of their mobile networks and, in particular, C-RAN. In this context, operators will have to create small DC infrastructures at the edge, in order to support RAN processing requirements. Having IT infrastructures on the edge, will enable operators to use unused resources to support 3rd parties to deploy applications in an edge computing environment, without having to invest on additional resources, especially for that purpose. As a consequence, mobile edge computing becomes cheaper, making it more appealing to network operators, content providers and developers.

The same concept can be applied to fixed networks, where virtualization is being adopted. In fact, for network providers with mobile and fixed operations, traditional central offices can be used to concentrate both mobile and fixed access, making those places good candidates to provide edge computing applications for all customers. Figure 5 outlines this vision.

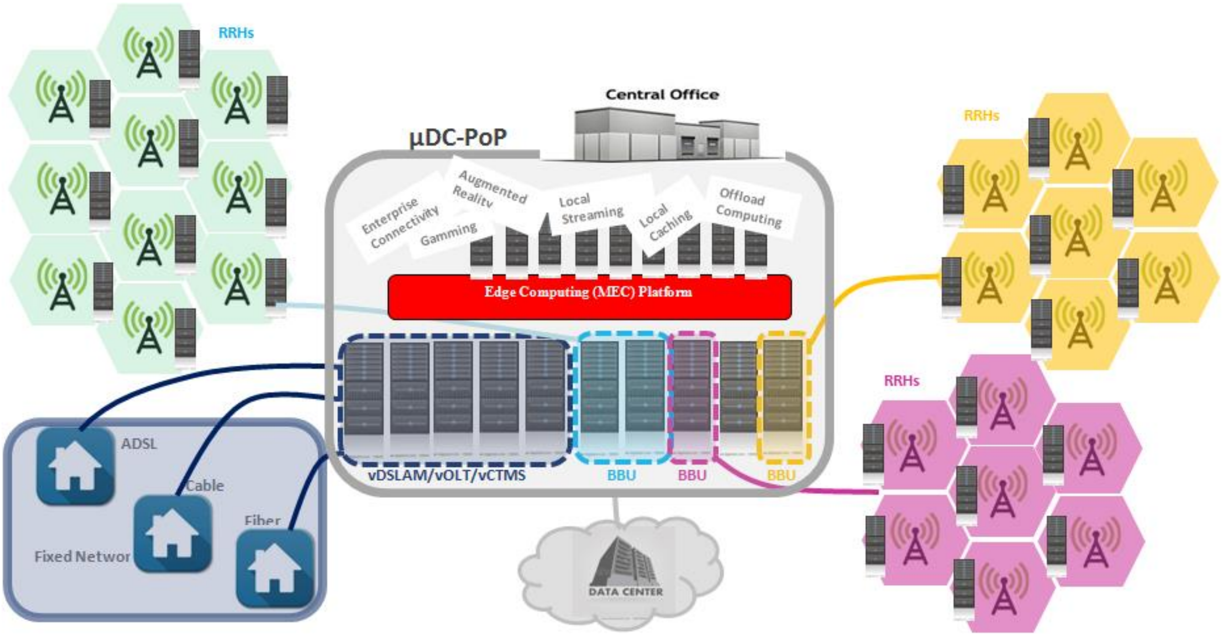

Figure 5: MEC mobile/fixed vision.

The edge computing architecture includes the MEC platform as the central point. This platform allows 3rd party applications to register and get authenticated/authorized, in order to access a set of services, available via the respective APIs Using the APIs, applications can request the platform to activate traffic offload (TOF) to serve customers from the edge, or enable access to other information provided by the operator.

The NFVI and VIM defined by the NFV are essentially reused to provide a virtual environment, this time for the MEC applications. The same applies to the management and orchestration layer, which can be used to manage the applications lifecycle and orchestrate them according to the best interests of end users and application providers. In the management layer, an additional manager is introduced, which is dedicated to managing the MEC platform itself, including its services and respective APIs. Figure 6 shows the proposed architecture for MEC.

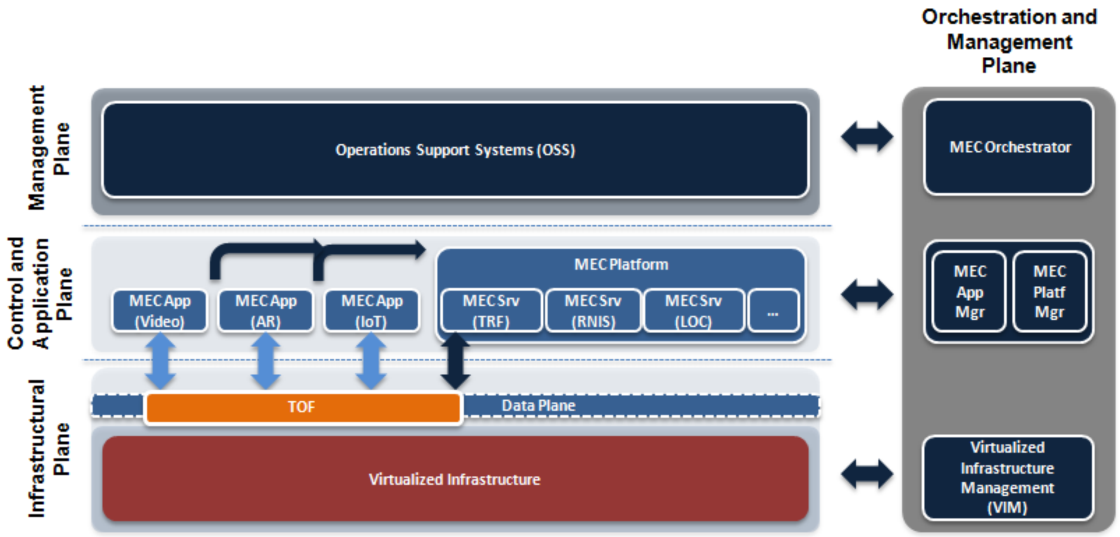

Figure 6: MEC architecture.

4.3.2.1 Impact on SDN/NFV integration

In the MEC architecture, SDN and NFV blocks are key technological components, as they complement to bring compute, storage and networking resources closer to the end user, also opening new APIs and hosting environments to both mobile operators and external players, which can also make use of RAN related information when offering a service. Proper integration of SDN and NFV is needed in order to enable the offering of virtual resources at the mobile edge, adapting to both the user demands and to the actual network load, supporting multiple tenants.

5G PPP Projects such as COHERENT, SESAME and Superfluidity are contributing to these topics.

4.3.3 Multi-domain/Multi-provider Orchestration

Market fragmentation results from having a multitude of telecommunication networks and cloud operators, each with a footprint focused to a specific region. This makes it difficult to deploy cost effective infrastructure services, such as virtual connectivity or compute resources, spanning multiple countries as no single operator has a big enough footprint. Even if operators largely aim to provide the same infrastructure services (VPN connectivity, compute resources based on virtual machines and block storage), inter-operator collaboration tools for providing a service spanning several administrative boundaries are very limited and cumbersome. This makes service development and provisioning very time consuming. For example, having a VPN with end-points in several countries, in order to connect multiple sites of a business (such as a hotel chain), requires contacting several network operators. Such an approach is possible only with significant effort and integration work from the side of the business. This is not only slow, but also inefficient and expensive, since the business also needs to employ networking specialists to do the integration instead of focusing on its core business

Technology fragmentation also represents a major bottleneck internally for an operator. Different networks and different parts of a network may be built as different domains using separate technologies, such as optical or packet switched (with different packet switching paradigms included); having equipment from different vendors; having different control paradigms, etc. Managing and integrating these separate technology domains requires substantial amount of effort, expertise, and time. The associated costs are paid by both network operators and vendors alike, who need to design equipment and develop complex integration features. In addition to technology domains, there are other reasons for having multiple domains within an operator, such as, different geographies, different performance characteristics, scalability, policy or simply historic (e.g., result of a merge or an acquisition). Multiple domains in a network are a necessary and permanent feature. However, these should not be a roadblock towards service development and provisioning, which should be fast and efficient.

A solution is needed to deal with both the multi-operator collaboration issue, and address the multi-domain problem within a single network operator. While these two problems are quite different, they also share a lot of common aspects and can benefit from having a number of common tools to solve them. In order to implement Network Service and resource orchestration across multiple administrative domains, which may belong to different infrastructure operators or service providers, hereby referred as "providers", the ETSI MANO NFV management and orchestration framework needs to be extended. One potential approach is presented next.

For multi-provider Network Service orchestration, a multi-domain orchestrator (MdO) offers Network Services by exposing an OSS/BSS – NFVO interface to other multi-domain orchestrators belonging to other providers. For multi-provider resource orchestration, a multi domain orchestrator presents a VIM-like view and exposes an extended NFVO – VIM interface to other multi domain orchestrators.

Figure 7 shows the different functional blocks responsible for service orchestration (SO) and resource orchestration (RO) as defined by the ETSI Open Source MANO (OSM). Resource orchestration is provided by the NFV Management and Orchestration, together with Network Service orchestration. On the other hand, service orchestration, which among others is responsible for configuring parameters within VNFs e.g. via Element Managers, is implemented by an OSS/BSS system.

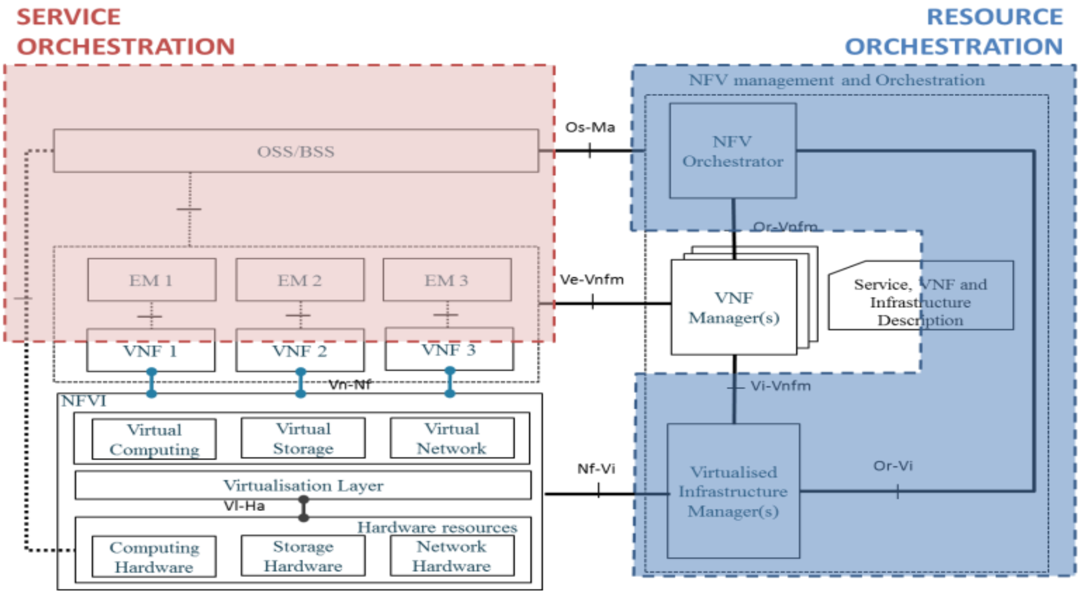

Figure 7: Service and resource orchestration

The Multi-provider Multi-domain Orchestrator (MdO) coordinates resource and/or service orchestration at multi-domain level, where multi-domain may refer to multi-technology (orchestrating resources and/or services using multiple Domain Orchestrators) or multi-operator (orchestrating resources and/or services using Domain Orchestrators belonging to multiple administrative domains). The MdO interacts with Domain Orchestrators via I3 interface APIs to orchestrate resources and services within the same administrative domains. The MdO interacts with other MdOs via I2 interface APIs (business-to-business, B2B) to request and orchestrate resources and services across administrative domains. Finally, the MdO exposes on interface I1 service specification APIs (Business-to-Customer, B2C) that allow business customers to specify their requirements for a service. Figure 8 illustrates the interfaces described above.

The framework also considers third party MdO service providers, which do not own resource domains but operate a multi-domain orchestrator level to trade resources and services from other providers (the ones actually owning such resources).

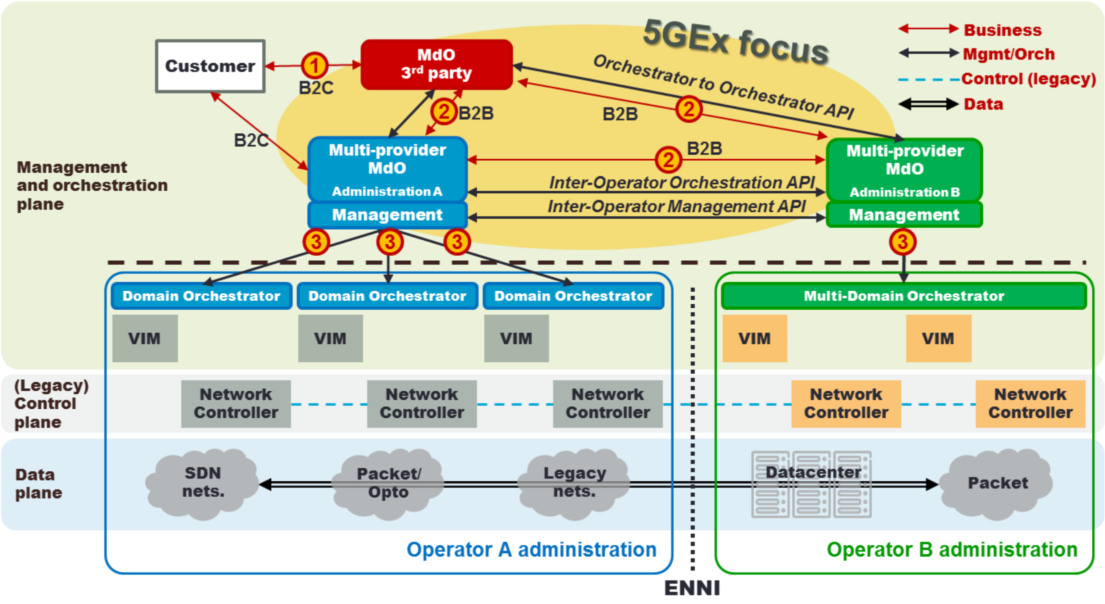

Figure 8: Multi-domain orchestration architecture

This approach allows for a clear separation between the multi-domain elements and the local elements, while still ensuring the flexibility to handle both multi-technology and keeping local infrastructure details confidential. The multi-domain orchestrator is in charge of abstracting the underlying infrastructure before it announces what utility and functions the operator is capable of to its neighbouring operators. Using such an inter-working architecture for multi-domain orchestration will make possible use-cases that are nowadays hard to tackle due to the interactions of multiple heterogeneous actors and technologies.

4.3.3.1 Impact on SDN/NFV integration

As part of the service decomposition task, the multi provider NFVO must be able to initiate potentially traffic engineered multi-provider connectivity. To be able to manage connectivity constraints, the multi provider NFVO interworks with a potentially external path computation element PCE, which has the visibility of inter provider topology. For example, the PCE may be a consumer of the topology information gathered to support the multi provider orchestration.

Typically, the connectivity data plane is configured by a control plane protocol, such as MP-BGP or PCEP. In this case, the multi-provider NFVO programs the control plane to implement resource orchestration, e.g. for allocating and distributing labels on ENNI links. For example, the multi-provider NFVO may use the PCEP protocol to ask an active stateful PCE to set up connectivity. Intermediate providers should verify that an incoming connectivity request has the corresponding orchestration state/policy that confirms that the connectivity request is accepted. Such an orchestration state may a priori exist or it may need to be established by the inter provider NFVO.

Alternatively, the inter-provider NFVO may configure directly the connectivity data plane along the principles of ONF SDN for OpenFlow controlled administrative domains. In this case coordination for ENNI data plane resource orchestration must be implemented by the inter provider NFVO.

Based on the following, it is evident that multi-provider, multi-domain orchestration requires tight integration between NFV and SDN, since data plane connectivity within each provider/domain and between providers/domains might involve SDN control coupled with VNF management and orchestration to provide a given end-to-end service.

5G PPP Projects such as 5G EX and SELFNET are contributing to these topics.

4.3.4 Network Programmability

A key goal in software networks is to increase the flexibility and programmability of 5G networks in order to bridge the gap between telecom business needs and operational management systems. An example architecture addressing this part is described next. As shown in Figure 9, the service programming and orchestration framework consists of the software development kit, the service platform, and different catalogues storing artifacts that can be produced, used, and managed by the system. Services developed and deployed by this system run on top of the underlying infrastructure accessible to the system via Virtual Infrastructure Managers (VIMs) and WAN Infrastructure Managers (WIM).

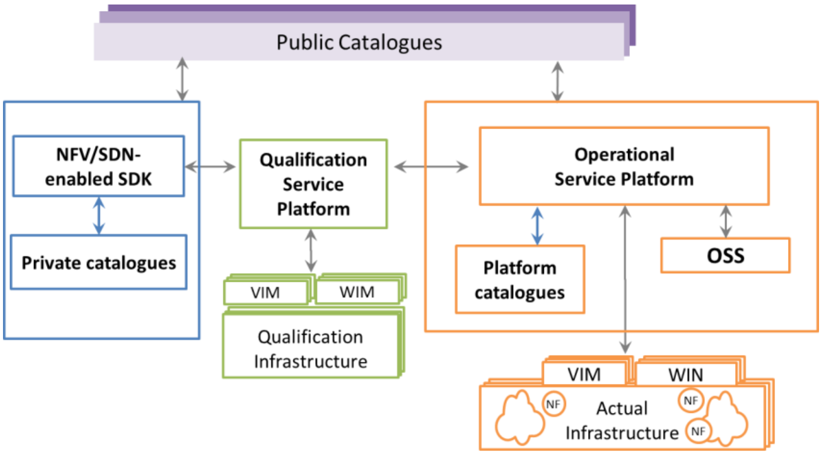

Figure 9: Network programmability architecture (SDK, Qualification, Service Platform)

We next provide some additional details for each of the functional blocks:

- **Catalogues:** for storing static information regarding network functions and services, like code, executables, configuration data, and specific management requirements and preferences. Contents, location, organization, and implementation of catalogues for different artifacts can vary considerably. However, users of these catalogues need to deal with them in a consistent fashion and the differences across different catalogues need to be harmonized and abstracted away. Different types of catalogues exist. As a high-level categorization, we foresee the following three types of catalogues:

- Private catalogues of service developers, where they can define, access, reuse, and modify services and service components.
- Service platform catalogues made available to authorized service developers for reusing existing components in their services, and used for storing services and their components that need to be deployed by the service platform.
- Public catalogues storing artifacts developed and maintained by third-party developers on arbitrary platforms accessible to service developers and service platform operators.

- **Service Development Kit (SDK):** The SDK supports service developers by providing a service programming model and a development tool-chain Figure 10 shows an overview of the foreseen SDK components. SDK design allows developers to define and test complex services consisting of multiple network functions, with tools that facilitate custom implementations of individual network functions. The implemented artifacts are stored in the developer's private catalogues. Moreover, service components can easily be obtained from external catalogues using the foreseen interfaces. The obtained artifacts can be directly used in a service or after being modified and tested using the SDK development tools. The service components and all the information necessary for deployment and execution of a service are bundled together into a package. The service package can be handed over to a service platform for actual deployment and for testing, debugging, and profiling purposes.

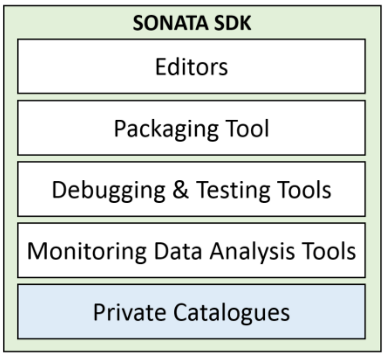

Figure 10: Example of SDK functions from SONATA

- **Qualification testing** module is a key element for SDN/NFV adoption and it is not present in existing software network architectures. This consists of a set of verification and validation (V&V) tools and exhaustive testing procedures over replicated operational infrastructures and under multiple testing conditions to assure the readiness of carrier-grade level and their direct utilization. Some examples of qualification functions can be automatic Service package integrity analysis, qualification test driver over chosen service platforms, a MANO translator can transform a service description between formats of different service platforms (since there is a coexistence of many open source and commercial MANO options).

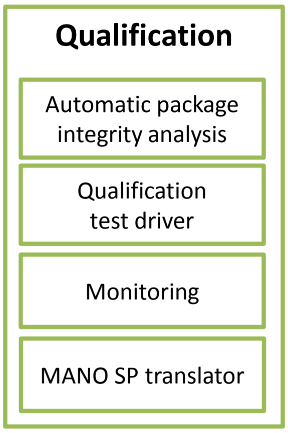
Qualification
Automatic package integrity analysis
Qualification test driver
Monitoring
MANO SP translator

Figure 11: Example of a Qualification functions

- **Service Platform:** As shown in Figure 12, an entry point (called gatekeeper module) in the service platform is responsible for processing the incoming and outgoing requests. The service platform receives the service packages implemented and created with the help of SDK part and is responsible for placing, deploying, provisioning, scaling, and managing the services on existing cloud infrastructures. It can also provide direct feedback about the deployed services to the SDK, for example, monitoring data about a service or its components. Service platforms are designed with full customization possibility, providing flexibility and control to operators and developers at the same time. The service developer can ship the service package to the service platform together with service- or function-specific lifecycle management requirements and preferences, called Service-Specific Managers (SSM) and Function-Specific Managers (FSM), respectively. SSMs and FSMs can influence the Service and VNF lifecycle management operations, e.g., by specifying desired placement or scaling behavior. By virtue of a modular design in the Management and Orchestration Framework of the service platform, the service platform operator can customize it, e.g., by replacing the conflict resolution or information management modules.

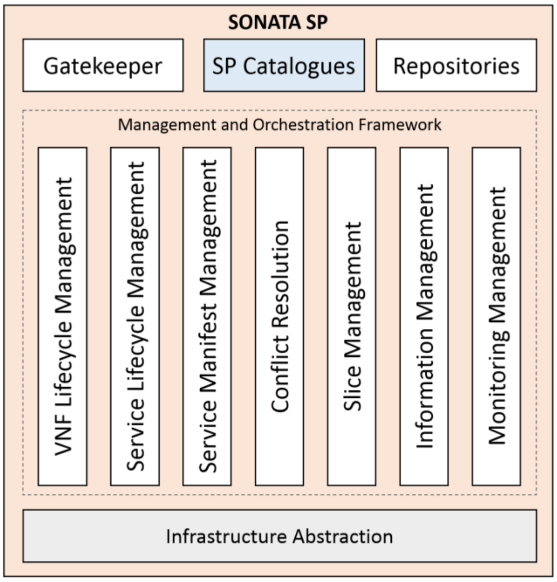

Figure 12: Example of modular Service Platform

- **Underlying Infrastructure:** The infrastructure needs to host and execute the actual network functions of a service, e.g., as a virtual machine. The service platform sends necessary information and instructions for execution and lifecycle management of services to the infrastructure. The infrastructure may belong to the service platform operator, or to a third-party infrastructure operator. The interaction between the service platform and the infrastructure is done through a VIM, e.g., OpenStack, which provides an abstract view on different infrastructure resources. This description is based on the assumption that a service platform runs directly on top of an actual NFV Infrastructure. However, the telecom system designs also enable a recursive deployment model, where a service platform can act as an abstraction to the underlying infrastructure for another service platform, creating a recursive, hierarchical service platform. While the service platform deals with service- and function-level functionality, the infrastructure deals with the resource level. It is here that service-level requirements have to be translated into resource-level requirements. The infrastructure management also takes care of infrastructure-level management functions.

4.3.4.1 Impact on SDN/NFV integration

These SDN/NFV systems are following the DevOps workflow, which is supported by the integration between the SDK and the service platform. This workflow implies continuous deployment and continuous integration during service development. The main entity exchanged between the SDK and the service platform is the service package to be deployed and runtime information like monitoring data and performance measurements regarding the service package, which is provided to the service developer during the development phase, as well as the runtime. This information can be used for optimizing, modifying, and debugging the operation and functionality of services.

5G PPP Projects such as SONATA and Virtuwind are contributing to these topics.

4.3.5 Network slicing

From the beginning of all 5G discussions in the research and industry fora, it has been agreed that 5G will have to address much more use cases than the preceding wireless generations, which first focused on voice services, and then on voice and high speed packet data services. In this case, 5G should be able to handle not only the same (or enhanced) voice and packet data services, but also new emerging services like tactile Internet and IoT. These use cases take the requirements to opposite extremes, as some of them require ultra-low latency and higher-speed, whereas some others require ultra-low power consumption and high delay tolerance.

Because of these very extreme 5G use cases, it is envisioned that different radio access networks are needed to better address the specific requirements of each one of the use cases. However, on the core network side, virtualization techniques can allow tailoring the network resources on separate (network) slices, specifically for each radio access network and use case, in an efficient manner.

From all the existing definitions and approaches, the foundation of 5G Network Slices is built around virtualization technologies, multi-tenancy and multi-service support, integrated network programmability and the adoption of the SDN/NFV design paradigms.

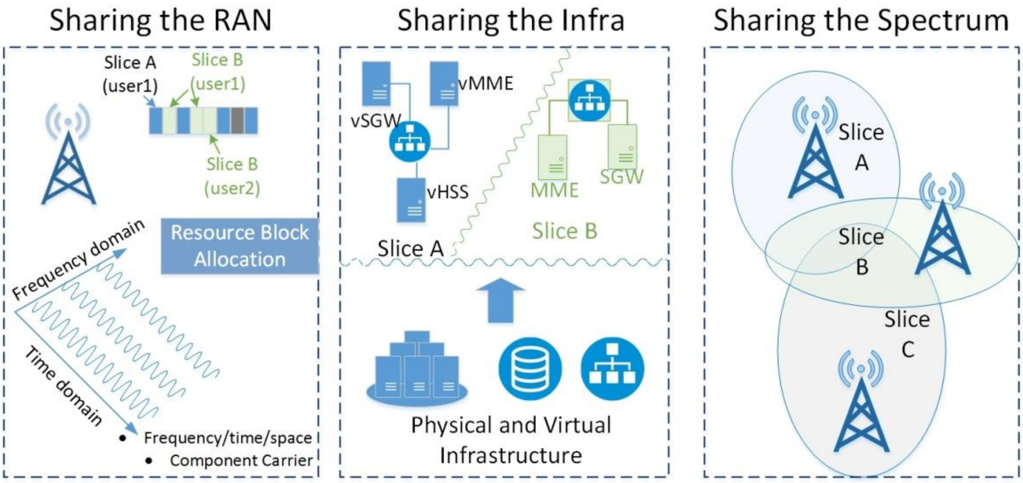

Figure 13: Network Slicing and RAN sharing

Note that Network Slicing is closely related to the concept of RAN Sharing for multi-service offering. For the evolved LTE 3GPP has defined and ratified different kinds of architectures with varying degrees of sharing (see 3GPP TS 23.251 specification). By means of mobile network resource sharing actually three dimensions of the problem exist, as also presented in Figure 13 .

• Sharing the Processing/Network/Storage Infrastructure: The switch fabric and the core network/system (cpu, storage, memory) are virtualized and shared between tenants, for the deployment of physical or virtual network elements (e.g. SP-GW, MME, HSS).

• Sharing the base station resources: Different sharing schemes and scenarios exist where the focus stays on the way the Physical Resource Blocks (PRBs) in frequency/time/space domain, are shared in the MAC layer to provide isolation, while maintaining the multiplexing gain.

• Sharing the Spectrum resources: The spectrum sharing problem is not related to MAC layer operations. It is related to the band of operation or activation of a component carrier. If multiple operators can share bands, the band of operation can be dynamically adjusted. Cognitive radio techniques fall under this category.

Network slicing techniques can also allow dedicating resources for even more specific use cases within the major 5G categories. For example, within the major IoT category, which is perhaps the most disrupting one, some autonomous IoT devices will have very low throughput, will have much longer sleep cycles (and therefore high latency), and a battery life thousands of times longer compared to smart phones or some other connected IoT devices that will have almost continuous control and data communications. Hence, it is envisioned that a single virtual core network could be used by slicing separate resources to dedicated radio access networks (RANs) that are better suited for specific use cases.

Note that there is no single design available for a unified framework that supports the Network Slicing concept and at the same time is able to:

- efficiently capture the need for integrated network programmability and SDN control,
- support service orchestration and NFV techniques in multi-domain RAN,
- provision for concepts like the Cloud-RAN and the Mobile Edge Computing (MEC).

Network slicing is also a key for introducing new actors in existing market at low cost -- by letting new players rent "blocks" of capacity, if this new market provides performance that are adequate with the application needs (e.g., broadcasting updates to many sensors with satellite broadcasting capabilities). However, more work needs to be done to define how network slicing will impact existing architectures like ETSI NFV, and to define the impacts of network slicing to guaranteeing quality-of-service.

4.3.5.1 Impact on SDN/NFV integration

Network slicing is probably one of the main examples of a need for tighter integration of SDN and NFV technologies. A slice can be defined as a “grouping of physical or virtual (network, compute, storage) resources which can act as a sub network and/or cloud and it can accommodate service components and network (virtual) functions”. This grouping of resources clearly requires NFV management and orchestration functions to handle the resources in addition to advanced SDN control for proper interconnection of these resources. Complex slicing approaches will require more advanced SDN/NFV integrated mechanisms, as currently researched by all 5G-PPP projects.

5 Relevant standardization activities

The 5G-PPP pre-standards working group is chartered with the role of identifying the SDOs (standard developing organizations) where the 5G-PPP projects may impact or get impacted with, as well as facilitating the standardization exploitation of the project results through e.g. setting up standardization roadmaps, and disseminating project results to the wider community of standardization stakeholders. As outlined in the white paper issued by the 5G-PPP pre-standards group [16], the SDN and NFV technologies are key pillars of the 5G system design in addition to radio access technologies, and hence play an important role in the overall 5G standardization roadmap. In this section we identify key aspects of the standardization activities that are most relevant to the SDN and NFV working areas, identify possible gaps, and contributions of projects, complementing other standardization summaries such as [16] and [29].

5.1 NFV

5.1.1 ETSI

The ETSI NFV ISG (Industry Specification Group) is probably the most relevant standardisation initiative so far arisen in the Network Function Virtualisation domain. Initially incepted by a group of top telecommunication operators at the end of 2012, has rapidly grown up since incorporating other operators, network vendors, ICT vendors and service providers.

The ETSI NFV specifications define the functional characteristics of each module, their respective interfaces, and the underlying data model. The data model is basically made up by static and dynamic descriptors for both VNFs (Virtual Network Functions) and Network Services. These latter are defined as compositions of individual VNFs, interconnected by a specified network forwarding graph, and wrapped inside a service.

The ETSI NFV roadmap foresaw two major phases. The first one was completed at the end of 2014, covering functional specification, data models, PoC description, etc. The second phase is closing in the current timeframe, and will release the expected final version of the ETSI NFV specification documents. The currently acting specification of the ETSI NFV architecture released in December 2014 and revisited later with security perspective. The informative reference document for the second release is available per [55]. For the forthcoming period (i.e. 2017-2018) the third release is planned. Already decided topics are:

- Information Modelling (IFA)
- End-to-end multi-site services management (IFA)
- Advances and considerations for MANO (IFA, EVE)
- Acceleration technologies (IFA)
- Charging, billing and accounting (EVE008)
- License Management (EVE)
- Security analysis and management (SEC)
- Reliability and availability considerations (REL)
- DevOps and continuous integration (TST)
- Testing (TST)
- Policy Management (IFA)

The ETSI NFV architecture supports multi-PoP configurations, where an NFVI-PoP is defined as the physical location where a network function is instantiated. A NFVI-PoP can be mapped to a datacentre or a datacentre segmentation isolated from the rest of world. Supporting multi-PoP configurations pertaining to different administrative domains is one of the gaps of the ETSI NFV architecture

Mobile Edge Computing (MEC) capabilities deployed in the edge of the mobile network can facilitate the efficient and dynamic provision of services to mobile users. The ETSI ISG MEC working group, operative from end of 2014, intends to specify an open environment for integrating MEC capabilities with service providers' networks, including also applications from 3rd parties. These distributed computing capabilities will make available IT infrastructure as in a cloud environment for the deployment of functions in mobile access networks. It can be seen then as a complement to both NFV and SDN. There are several aspects, such as MEC federation, that have not been tackled yet in MEC and can be considered as a potential gap not only in MEC.

ETSI NVF Security Working Group. The NFV SEC Working Group comprises computer, network and Cloud security experts, representing network operators, equipment vendors and law enforcement agencies. The working group's main objectives, as presented in [61], are to advice the NFV ISG on all matters of the relevant security technologies and develop a wide range of industry specifications that:

- Identify both the NFV-specific security problems, as well as the technological advantages of the NFV environment that can be harnessed to improve the security of the network operators' services
- Provide specific guidance on various aspects of the NFV security in a systematic, holistic manner, building trust from secure hardware modules to software and covering identity management, authentication, authorization and secure attestation, as well as the means of global monitoring of the whole NFV environment and decisive operational security actions in response to security breaches
- Address in detail the security of the present Open Source-based platforms (such as OpenStack)
- Contribute to solving the problem of implementing Lawful Interception (LI) in the NFV environment
- Work in close collaboration with other ETSI NFV WGs, PoCs, as well as external organisations (ETSI TC Cyber, ETSI TC LI, Trusted Computing Group and contributing members of OpenStack)

The group was initially created as an expert group in Phase I in 2012 with the objective to establish the NFV security problem statement and advise all other working groups. In Phase 2, it grown to a full working group with 18 active participants, regular meetings and a steady stream of contributions. For the NFV SEC Working Group, the NFV technology presents unique opportunities for addressing security problems. It can improve the security properties of network functions, facilitate agile provision of secure services by the carrier and provide better protection of the carrier cloud.

A list of the current ETSI NFV SEC WG work items is available here [62]. This covers a set public documents that addresses multiple aspects such as: a thorough assessment of potential new security concerns related to NFV, security problem statement (ETSI GS NFV-SEC 001), a review of security features in open source management software in NFV (with focus on open stack), security and trust guidance related to NFV development, architecture and operation (NFV-SEC 003), privacy and regulatory aspects (NFV-SEC 004 and NFV-SEC 006) and certification deployment scenarios (NFV-SEC 005) and use cases and technical approaches for multi-layer host administration (NFV-SEC 009).

5.1.2 IETF/IRTF

In the Internet Engineering Task Force (IETF) – and in the sibling organisation, the Internet Research Task Force (IRTF) – there are distinct working groups addressing topics of interest for 5GPPP project partners, both from the research perspective (IRTF) and from the normalisation perspective (IETF). Some of the activities of interest are the following:

- IRTF NFV RG. The NFV Research group explores issues around NFV. A recent contribution [15] has been released to analyse the gaps about handling of multiple administrative domains in NFV environments. 5GEx and 5G-Crosshaul are contributing to this RG.
- IRTF SDN Research Group. This group adopted a document exploring the separation between Service and Transport concerns in SDN [14]. It addresseed problems such as unclear responsibilities for service provisioning, complicated reuse of components, monolithical control architectures, and complex network diagnosis and troubleshooting by a cooperating layered architecture for SDN (CLAS). CLAS consists of two strata, a Service Stratum for service provisioning and providing capabilities to external applications, and a Transport Stratum for connecting service end nodes building the end-to-end service requested by applications on top.

The IETF/IRTF is becoming a very active and relevant venue in the area of software networks. In addition to the two IRTF research groups mentioned above, there are many WGs addressing some virtualization aspects. An overview of these efforts as well as relevant gaps is reported in a “live” document (officially adopted by the IRTF NFVRG, driven by 5GPPP project partners) [16]. For brevity, we do not include here all the analysis, but just cite a few as examples:

- Redundancy and reliability mechanisms are currently not dealt with by the Service Function Chaining (SFC) or any other WG in the IETF. While there was a main driver for forming a new WG on this topic (VNFpool), no WG is hosting this as this time.
- It would be worthwhile to see if some of the specific approaches developed in the NVO3 WG (e.g. overlays, traffic isolation, VM migration) can be applied outside the Data Center (DC) environment, and specifically if they can be applicable to network virtualization. These approaches would be most relevant to the NFVI and the VIM part of the MANO part of the ETSI NFV architecture.

5.1.3 OASIS TOSCA

OASIS TOSCA[1] (Topology and Orchestration Specification for Cloud Applications) focuses on portability and manageability of cloud applications and services across their lifecycle. It is working on a simple profile for NFV, defining data models for networks services and VNFs, and templates for VNFs, virtual links, forwarding graphs, and network services. This profile is still worked upon (Draft 03 – March 2016).

[1] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=tosca

5.1.4 TM Forum ZOOM

TM Forum is global industry association focused on for digitalization of every industry and its transformation. It has a practical way through collaboration programs and industrial communities which lead to rapid prototypes – ranging from digital business models to open APIs – and associated procedures. TM Forum provides industry best practices and standards to accelerate adoption.

ZOOM (Zero-touch Orchestration, Operations and Management) initiative is devoted to NFV and focused in providing business and operational orientation to NFV on-going research activities. This is very relevant in the 'operation' area understanding the impact of software networks in Operational Support Systems (OSS) and how to build a blue print for an end-to-end management across telecom operators. [41]

5.1.5 Small Cell Forum

Small Cell Forum (SCF) has carried out different studies analysing the introduction of virtualization technologies in small cell [57][58][59]. A small cell is split into two components: a remote small cell, where functions are non-virtualized, i.e. they are Physical Network Functions (PNFs) that are implemented via a tightly coupled software and hardware, and a central small cell, where functions are virtualized, i.e. they are Virtual Network Functions (VNFs) that are implemented by abstracting the hardware from the software, so that they are executed on a pool of shared computation, storage and networking resources. A central small cell will serve multiple remote small cells. The central and remote small cells are physically connected through the fronthaul link.

The consequences of the Small Cell concatenation to PNFs and VNFs, is under investigation in the frame of the SCF. The various solutions administered for achieving virtualisation in Small Cells are studied in [56]. The document also addressed the impact of these choices in the architecture and supporting infrastructure at the edge. The latter affects both NFV and MEC architectures and is still missing clean and optimised implementations.

5.2 SDN

5.2.1 ITU-T

ITU-T Study Group 13[2] is the lead study group of SDN in ITU-T and develops the SDN framework, including SDN terminology, as baseline of ITU-T SDN standardization activities.

ITU-T Study Group 15[3] defines technologies and architectures of long-haul optical transport networks enabling; fibre- or copper-based access networks; and home networks. The study group is working on two new recommendations on "Common control aspects" and "Architecture for SDN control of transport networks". It intends to descript concepts common to both SDN controller and ASON (automatically switched optical networks) control approaches. It investigates also potential interaction among ASON control, SDN controllers, management functions, and transport resources.

[2] http://www.itu.int/en/ITU-T/studygroups/2013-2016/13

[3] http://www.itu.int/en/ITU-T/studygroups/2013-2016/15

5.2.2 ONF

The Open Networking Foundation[4] (ONF) is a member-driven organisation promoting the adoption of SDN through the development of the OpenFlow protocol as open standard for the communication between the controller and the data forwarding network elements. Apart from OpenFlow specifications development, ONF is very active in the definition of architectures around SDN, independent of the protocol to be used for accessing and configuring the equipment (the so called South Bound Interface from the controller perspective). A number of ONF relevant initiatives are listed as follows:

- The Architecture WG releases architectural specifications for SDN. The two main specifications are [11] and [13], providing insight on SDN architectural matters, with emphasis on the controller capabilities and the interfaces with the other elements in the architecture (applications on top, devices below).
- OpenFlow definition. Two standards are defined to control the forwarding behaviour (OF-switch) and to configure and manage the switches (OF-config). 5G-Crosshaul is participating in ongoing activities to define extensions for optical switches and for microwave transmission [60]. OF-switch is actively developed further, seeking to keep a balance between increasing functionality without increasing the amount of protocol headers that can be used for matching packets.
- APIs between layers. There are a number of initiatives in ONF exploring APIs between layers in the SDN architecture. The Transport API (T-API) specifies an API for the following transport network controller services: topology, connectivity, path computation, virtual network, and notification. On the other hand, the North Bound Interface (NBI) group is working on Intent based NBIs. An Intent NBI expresses what a network service-consuming agent (i.e., an application or operator) requires from a network (i.e., it defines a requested network service) but it does not specify or constrain how that service may or should be delivered by the network.
- The Carrier Grade SDN WG is working on a framework document where a number of requirements are identified to accomplish a truly operational SDN from the perspective of network operators and service providers. Aspects like the ability to deliver managed services end-to-end, SLA compliance, Service Operations and Maintenance capabilities, etc. are highlighted as core of an SDN Carrier Design. The Carrier Grade SDN framework document also describes some of these issues considering Inter-Carrier interoperability and Interoperability/co-existence between SDN and Legacy networks. The 5GEx project is cited in such document as exemplary project analysing the multi-domain aspects.
- Slicing for 5G. The Mobile WG has started to produce a report for the application of SDN to 5G slicing. The idea is to describe how key functional aspects of the SDN architecture apply for 5G enablement, including the business-driven concept of network slicing. Ideas like the one proposed in [14] are also being considered in this group.

5.2.3 IEEE

The IEEE has created an SDN initiative[5] without creating new standards so far. The recommended practice 802.1CF specifies how terminals can connect to access routers using IEEE 802 technologies. This recommended practice defines an access network reference model relevant entities, and reference points for communication among these entities.

[4] https://www.opennetworking.org/

[5] http://sdn.ieee.org/

The IEEE P1903 WG[6] develops specifications for Next Generation Service Overlay Networks (NGSON). It is currently working on the specification of service enabling functions, which can be provided as Virtualized Network Functions (VNFs) to support NFV applications.

Relevant to consider also IEEE 1903-2011 "Functional Architecture of Next Generation Service Overlay Networks" describes a framework of Internet Protocol (IP)-based service overlay networks and specifies self-organizing networking capabilities, including routing and forwarding schemes, which are independent of underlying transport networks.

5.2.4 MEF

The Metro Ethernet Forum[7] (MEF) defined Carrier Ethernet or Ethernet connectivity as a service. Basically Ethernet services with different topologies, E-Line, E-LAN, and E-Tree, are defined among User-Network-Interfaces (UNI) with defined service level agreements (SLA). As such, these Ethernet services can be used as an abstraction of connectivity among different nodes. MEF has also defined how Ethernet services could be stitched together at Network-Network-Interfaces (NNI) and how data centers could be connected.

The MEF has started specification work on lifecycle orchestration (LSO), providing a reference architecture and framework (MEF-55). This framework defines requirements on APIs, but the definition and implementation of the APIs is still pending.

[6] http://standards.ieee.org/develop/wg/1903_WG.html

[7] https://www.mef.net

6 Relevant open source activities

In this section, the key open source efforts/platform relevant to SDN and NFV are introduced. Many of these are being adopted, used or contributed from 5G PPP projects.

6.1 SDN Controllers

To satisfy the multi-tenancy requirements in today's modern data centers, tunneling protocols such as VLAN and GRE are used. Their aim is to provide abstraction layer on the top of the physical infrastructure in order to ensure tenant isolation, expanded address space, L2 over L3 services, and VM interconnection between multiple datacenters.

The SDN technology provides a separation between network control and data plane functionality. The SDN controller is a central independent entity, responsible for the stitching the two planes in terms of providing a control logic as a software application that can enforce specific rules and policies over the physical resources. Having the global view of the network dataplane, the controller manages the switches using southbound protocols such as OpenFlow, OVSDB, Netconf etc.

Several SDN controllers exist currently among the open source community and within the industry. One of the first widely available, the NOX controller [31] was originally developed by Nicira and released as open-source software. Other controller frameworks aimed at deployment in production environments, include Beacon [32], Maestro [33] and FloodLight [35], all of which are implemented in Java. **FloodLight** is the open source basis for Big Switch's commercial OpenFlow controller. Large number of companies such as Ciena, Cisco, Ericsson, Fujitsu, Huawei, Intel, NEC are involved in the development and support of **ONOS** [34] an open source SDN OS controller aimed for service providers architected for performance, high availability, scale-out and well-defined northbound and southbound abstractions and interfaces to create apps and services. **OpenContrail** [36] led by Juniper Networks, is another SDN controller and a modular project that provides an environment for network virtualisation and published northbound APIs. In particular, the network virtualisation is provided by means of a set of building blocks and high level policies; it integrates an SDN controller to support network programmability and automation, and a well-defined data model to describe the desired state of the network. **OpenDaylight (ODL)** [37] is currently the de facto controller platform. Moreover it goes beyond the controller functionality by providing open source framework for SDN programmability. Twelve founding members have actively supported OpenDaylight, some of which include Cisco, Juniper and IBM. ODL allows the inclusion of north or southbound projects, standards and protocols due to its extensibility. It is developed in Java and based on Apache Karaf - a small OSGi based runtime that provides a lightweight container onto which various components and applications can be deployed. ODL is being more focused on the support of networking protocols for virtualization platforms such as OpenStack via ne Neutron ML2 plugin.

All three, ONOS, OpenContrail and ODL controllers are integrated in **Open Platform for NFV (OPNFV)** [51] releases and they are distributed together with Open Stack.

Another group of controllers exist on the other hand, dedicated for Dockers and container networking, such as Weave and Flannel.

Weave [49] was created by Zett.io, in order to address the Docker's networking issue. It provides a single virtual network abstraction for Docker containers deployed across multiple hosts. Weave single switch network representation allows to multiple containers, an access to, and abstracted overlay of the same physical resources (file systems, databases, etc). Similarly, **Flannel** [50] provides an overlay network view based on IP addresses and VXLAN solution. Each host running a Docker container, has assigned a unique IP addresses from a given set of subnets addresses to enable inter-container communication. Flannel uses etcd to store mappings between the virtual IP and host addresses. A flanneld daemon runs on each host and is responsible for watching information in etcd and routing the packets.

6.2 Network Service Automation Software

Where the network can be controlled with the open source software above, the software and applications that can run upon such software-controllable networks must also be managed in an automated fashion. This automation is supplied by the process of orchestration which can either be seen as a completely centralised or decentralised process. The former is known as composition in the software/IT world and the latter as, indeed, orchestration, although in the context of software or NFV orchestration the process is mainly not a decentralised one.

The Linux Foundation created the **Open Platform for NFV (OPNFV)** [51] initiative in 2014, after the creation of the OpenDaylight Project in April 2013 to boost adoption of SDN and NFV. OPNFV is an integrated, open source platform to accelerate the introduction of new NFV products and services, by essentially bringing together service and NFV providers, cloud and infrastructure vendors, developers' communities, and customers into a new NFV ecosystem. OPNFV was motivated by the European Telecommunications Standards Institute and ETSI NFV to achieve consistency among open standards in terms of performance and interoperability among virtualized network infrastructures. OPNFV promotes an open source network aimed at accelerating innovation and collaboration between the participating communities based on current technological enablers. It is largely a collection of technologies and open source projects based around such technologies as OpenStack and OpenDayLight. It looks into the additional features of Fault Management, IPv6, SFC, L3VPN, Reservation, Continuous Integration and Deployment in its application domain. Details of all projects that bring additional functionality to OPNFV can be viewed at the OPNFV wiki site [53]. OPNFV provides new releases every six months (available around March and September). Last stable version named 'Colorado' was available in fall 2016.

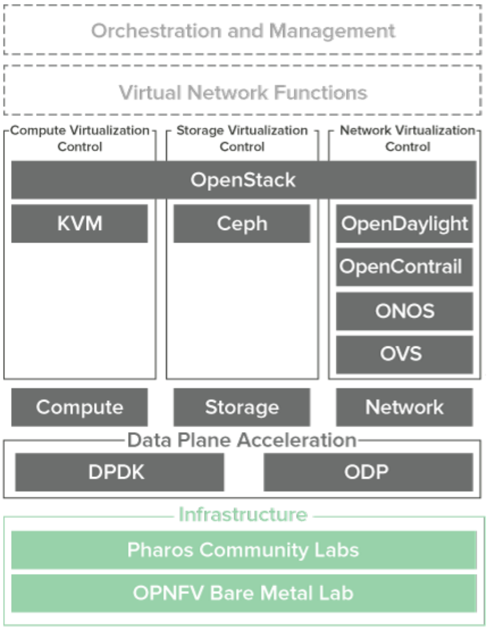

Figure 14: OPNFV architecture

The OpenStack community also is developing a NFV project under the Tacker project. **Open Stack Tacker [43]** started as a subproject of OpenStack Neutron and has since been placed into its own official project. The Tacker project takes the TOSCA Simple Profile for NFV as the basis of its VNF descriptors and then converts this into an appropriate Heat Orchestration Template (HOT) . Tacker is currently considering how to implement SFC (service Function Chain). Tacker has a general purpose monitoring framework that can be leveraged within the TOSCA-NFV descriptors. The delineation of Tacker to OPNFV is seen such that Tacker can present its VNFMs to be managed by OPNFV.

OpenMANO [42] was an open source project integrated under **OSM** hosted under ETSI NFV WG that aims to provide a practical implementation of the reference architecture for NFV management and orchestration proposed by ETSI NFV ISG, and being enhanced to address wider service orchestration functions. The project is available under the Apache 2.0 license, it was first released in early 2015 and it is currently under active development. The OpenMANO framework is essentially focused on resource orchestration for NFV and consists of three major components: openvim, openmano, and openmano-gui. The first component is essentially focused on the resource infrastructure orchestration, implementing express EPA (Enhanced Platform Awareness) requirements to provide the functionality of a Virtual Infrastructure Manager (VIM) optimized for virtual network functions and high and predictable performance. Although openvim is comparable to other VIMs, like OpenStack, it provides a set of features such as direct control over SDN controllers by means of specific plugins (currently available for Floodlight and OpenDaylight), aiming at high performance data plane connectivity. In addition, openvim supports a northbound API available to the functional resource orchestration component openmano to allocate resources from the underlying infrastructure, by direct requests for the creation, deletion and management of images, flavours, instances and networks. Openvim comes with a lightweight design that does not require additional agents to be installed on the managed infrastructural nodes. The functional resource orchestration component itself is controlled by a northbound API, which are currently suitable to be used directly by network administrators via a web-based interface (openmano-gui) or by a command line interface (CLI) that eases integration in heterogeneous infrastructures and with legacy network management systems. The functional resource orchestrator is able to manage entire function chains that are called network scenarios and that correspond to what ETSI NFV calls network services. These network scenarios consist of several interconnected VNFs and are specified by the function/service developer by means of easy-to-manage YAML/JSON descriptors. It currently supports a basic life-cycle for VNF or scenarios (supporting the following events: define/start/stop/undefine). The OpenMANO framework includes catalogues for both predefined VNFs and entire network scenarios, and infrastructure descriptions carrying Enhanced Platform Awareness (EPA) information.

OpenMANO along with Canonical's Juju and Rift.io, figures as key technical components in ETSI's announced **Open Source MANO (OSM)**. Release ONE announcement took place in fall 2016. The approach taken in OSM is to split responsibility of resource and service orchestration, carried out by OpenMANO and Rift.io, respectively. OSM has its founding members as BT, Canonical, Intel, Mirantis, Rift.io, Telefonica, Telekom Austria and Telenor. OSM's motivation is for to become a reference implementation for the specification work happening through ETSI and aims at making 6-monthly releases of their ETSI compliant NFV-MANO framework. The architecture from a layered perspective has a GUI at the top which interacts with the Network Service Orchestrator (implemented by Rift.io) that oversees the management of requests to the Resource Orchestrator (provided by openmano) and VNF Configuration (provided by Juju). This all eventually results in virtual infrastructure instances running on the Virtual Infrastructure Manager (OpenMANO's openvim or OpenStack).

Open-O [54] is a project under Linux Foundation to develop an open source software framework and orchestrator to enable agile software-defined networking (SDN) and network function virtualization (NFV) operations. Open-O is supported by multiple organization such as Brocade, China Mobile, China telecom, GigaSpaces, Huawei, Intel, Red Hat or ZTE. Open-O first release called "SUN," has been released in November 2016. With an end-to-end service approach, the open-o project aims to accelerate multi-vendor integration, service innovation and improve agility across network operations. It provides an open orchestrator to bridge the gap between virtualized functions and connectivity services.

It is also relevant to mention here the **ECOMP** [63] (Enhanced Control, Orchestration, Management & Policy) platform pushed by US AT&T since from July 2016 announced the intention to create an open source group to work with the community under Linux foundation in 2017 [64]. ECOMP includes SDN controller part, NFV orchestrator, service perspective and associated procedures impacting the OSS. AT&T has established partnerships with Bell Canada and Orange Europe to test and deploy ECOMP framework in order to reach more mature operational level. However Linux Foundation has not yet announced ECOMP as an official project probably because there is already the Open-O project hosted under its umbrella. In this sense, both ECOMP and Open-O encompass more than MANO with the global service vision and we may see a future collaborations and contributions under Open-O.

Hurtle [44] is an ETSI NFV compatible Network Function Orchestrator. Hurtle was part of the Mobile Cloud Networking FP7 project [47]. It's been developed over the past three and a half years of the project and is also used in other projects. The key goal of hurtle is to deliver any software as a service, on-demand and given the trend of bringing software-based approaches seen in cloud computing within the networking sphere, hurtle has shown to be able to deliver networking software (for example OpenEPC [49], OpenAirInterface [50]) as a service specific to a tenant's needs. One of the key features of hurtle is the ability to deliver multi-domain, and multi-service composed services. Hurtle supports a number of VIMs including OpenStack, CloudStack and Joyent's Triton. Given Hurtle's service focus, placement logic is focused and specific to those characteristics of services, typically cost and latency. Hurtle can deploy a composed service across a set of data centres (sometime referred to as PoPs) and also data centres where no peering-like agreements exist. All components within Hurtle can be part of continuous integration and deployment processes. Hurtle is very extensible, allowing for integration to other NFVOs, for example there existing integration between it and OpenBaton. Hurtle also has billing services available to it from Cyclops [48]. Hurtle delivers tenant's service instances through the Service Manager and creates these service instances through the Service Orchestrator. The SO has the capability to automatically monitor and react to events (scaling- or fault-related) generated by the resources that provide the service instance.

OpenBaton [40] is an ETSI NFV compliant Network Function Virtualization Orchestrator (NFVO). OpenBaton was part of the OpenSDNCore project [40] started almost three years ago by Fraunhofer FOKUS with the objective of providing a compliant implementation of the ETSI NFV specification. OpenBaton is easily extensible. It integrates with OpenStack, and provides a plugin mechanism for supporting additional VIM types. It supports Network Service management either using a generic VNFM or interoperating with VNF-specific VNFM. It uses different mechanisms (REST or PUB/SUB) for interoperating with the VNFMs. The orchestrator implements the key functionalities of the MANO architecture. Specifically, it: i) currently uses the OpenStack as first integrated NFV PoP VIM; ii) maintains an overview on the infrastructure, supporting dynamic registration of NFV PoPs; iii) receives virtual network function packages from the different users including VNF images and virtual network functions descriptors (VNFDs); iv) deploys on-demand the VNFs on top of an infrastructure consisting of multiple data center instances (NFV PoPs); and v)deploys in parallel multiple slices one for each tenant, consisting of one or multiple VNFs. OpenBaton's initial focus was to provide the main functionalities for provisioning and managing Network Services. However, today, a number of advanced features is supported, such as mechanisms for increasing the automation in NS management, including auto-scaling (in/out), monitoring, fault management, TOSCA, etc.

FP7 T-NOVA [45] designed and implemented a management/orchestration platform named **TeNOR** [46] for the automated provisioning, configuration, monitoring and optimization of Network Functions-as-a-Service (NFaaS) over virtualized Network/IT infrastructures. TeNOR is one of the core components of the T- NOVA architecture and being used in some 5G PPP projects. It is responsible for network services and VNFs lifecycle management operations over distributed and virtualized network/IT infrastructures. In essence, TeNOR is focused on addressing two of the most critical issues related to NFV operational environments: (i) automated deployment and configuration of services composed of virtualized functions; and (ii) management and optimization of networking and IT resources for VNF hosting. Additionally, it is responsible for the identification and allocation of resources to network services, Service Level Agreement (SLA) fulfilment and resource usage optimization. The TeNOR orchestration platform includes the implementation of an algorithm to map services jointly into the network and cloud infrastructure, choosing the optimal resources to deploy the service and scales and migrate according to the monitoring data received. Both external and internal interfaces of TeNOR are REST/JSON, in a micro-service oriented architecture, which was selected to ensure a lean and modular implementation and operation of the system. TeNOR provides a "generic" VNF Manager, while allowing 'specific' VNF Managers to be provided together with a (set of) VNFs (with an authentication/authorization scheme to control which functions should or shouldn't be delegated to those 'specific' VNF Managers. VIM implementation is based on an integration of OpenStack and ODL for the realization of the NFVI-PoPs, in which OpenStack Cloud Controller and ODL comprise the VIM for the management of the infrastructure.

6.3 Remarks associated to SDN and NFV frameworks usage

There is much competition in the area of open source activities related to NFV and SDN. This level of activity is encouraging, however makes the decision of choosing an open source-based solution difficult especially when long term viability and maintenance of on particular project is considered. It is now not clear of what should be the reference implementation or even there will be one, given the competition and the existence of many open NFV solutions (OSM, Open-O, recently open ECOMP, OS Tacker, etc) taking a high-level approach of ETSI NFV/MANO. Initially OPNFV was seen to be this lower level reference implementation. OPNFV has matured and expanded integrating ODL, Open Contrail and ONOS controllers. OSM and Open-O have a own interpretation of some ETSI NFV/MANO elements when not completely specified such as descriptors parameters, packaged chains to on-board, etc or on the scope.

All initiatives' code is available under OS licensing (mainly Apache 2.0) and all support OpenStack as the baseline VIM. A small number support more than just OpenStack and this approach should be encouraged so as not to encourage mono-cultural thinking and be subject to the effects of that. An example is OSM which is now natively supporting VMWare environment. Others are expanding towards containers orchestration support (i.e. kubernetes) which is a R&D topic to explore understanding how to properly support network deployments.

There are frameworks that usually target the IT domain that have been shown to provide capabilities described by ETSI NFV and although they do not quite fit the bill a suitable mapping and relevance can be established to the ETSI NFV model. This is very well demonstrated by the Rift.io, Cloudify and Hurtle projects, which although can be considered as service delivery enablers can also be used effectively in the NFV sphere. Should those considering OSS to enable their NFV deployments, considering projects that also target IT service delivery is a wise move.

7 Conclusions

This whitepaper is the first outcome of the 5G-PPP Software Network WG, which aims at analyzing and addressing unification and applicability of key research topics related to Software Networking including SDN and NFV, integrating the vision as developed and promoted by the 5G PPP projects and also aligned with the 5G-PPP Architecture WG.

SDN and NFV are core technologies in the transformation of networks towards 5G. This transformation trend is quite visible now in several Industries, through the use of ”softwarization" on general computing platforms to optimize operational processes and in bringing rapidly and efficiently new values in infrastructures.

This document conveys a first converged vision on the challenging research issue of combining SDN and NFV technologies. It firstly introduces early attempts and steps to achieve this combination by main standardization organizations, to then provide a first architectural proposal to integrate NFV and SDN for 5G. This architectural proposal represents an initial step towards effectively integrating SDN/NFV, which we plan to further evolve (by including more results from the work conducted within the framework of the contributing projects) in a future white paper. After introducing this conceptual architecture, we enumerate several examples where software networks (will) play a significant role, identifying specific SDN/NFV integration aspects that have not yet properly addressed. Last, but not least, we provide a survey of existing relevant standardization and open source activities, with a two-folded goal: (i) get familiar with existing efforts, and (ii) identify open gaps that need to be tackled.

There is a high competition concerning open source SDN / NFV frameworks as seen in previous section. This should not be necessarily considered as a negative fragmentation but potential implementation alternatives. The OS initiatives and 5G projects should work on exploring advance functionalities (i.e. multi-domain slicing, recursiveness, multi-provider, etc) to enable future scenarios ensuring the consistency in the information/data models, compatibility of network service descriptors and open APIs.

Reviewing on-going 5G PPP 1st phase projects, the SN working group has gathered the OS frameworks that are being used. There is a wide adoption of openstack (nova, neutron) as baseline VIM, being docker/kubernetes the second most prominent option. The most popular SDN/NFV OS frameworks are: OpenDayLight as SDN Controller (with ONOS in 2nd place) and TeNOR as NFV orchestrator (used in at least four projects) and also OSM framework if we consider not only the adoption but the established links and expected impact. It should be noted that these 5G PPP projects are running in parallel to the OS community developments which were not available when the projects started (first stable releases of OSM or Open-O date from fall 2016 while 5G PPP projects were already in implementation phase). Another interesting find is that Open-O or ECOMP do not appear in the adoption radar of EU R&D 1st phase 5G PPP which is should be corrected in 2nd phase. The first wave of 5G PPP projects is delivering pieces to OS initiatives since there is not an OS R&D EU community established. Thus the projects are influencing OS developments or creating complementary solutions while researching advance functionalities. Some new MANO platforms, SDN controller extensions, developer toolsets (SDK) and Service Platforms are being released by 5G PPP projects.

We believe this whitepaper represents a first solid step towards getting a harmonized SDN/NFV combined architecture from the main 5G-PPP phase I projects working in the area of SDN and NFV. This will serve as basis for subsequent work on further specification of this architecture and its interfaces.

8 References

[1] 5G PPP vision document - https://5g-ppp.eu/wp-content/uploads/2015/02/5G-Vision-Brochure-v1.pdf

[2] https://5g-ppp.eu/5g-ppp-phase-1-projects/

[3] 5G Action Plan for Europe https://ec.europa.eu/transparency/regdoc/rep/1/2016/EN/1-2016-588-EN-F1-1.PDF

[4] https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-July-2016.pdf

[5] Open Networking Foundation, TR-502, "SDN Architecture", Issue 1.0, June 2014, available at https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf

[6] https://www.sdxcentral.com/5g/definitions/what-is-5g/

[7] Open Networking Foundation, TR-521, "SDN Architecture", Issue 1.1, January 2016, available at https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-521_SDN_Architecture_issue_1.1.pdf

[8] ETSI, GS NFV-EVE 005 (V1.1.1) - (12-2015), "Network Functions Virtualisation (NFV); Ecosystem; Report on SDN Usage in NFV Architectural Framework", available at http://www.etsi.org/deliver/etsi_gs/NFV-EVE/001_099/005/01.01.01_60/gs_NFV-EVE005v010101p.pdf

[9] ITU, Recommendation ITU-T Y.3300 (06-2014), "Framework of software-defined networking".

[10] IETF, RFC 7426 (January 2015), "Software-Defined Networking (SDN): Layers and Architecture Terminology".

[11] View on 5G Architecture, 5G PPP Architecture Working Group, Version 1.0, July 2016, available at https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-5G-Architecture-WP-July-2016.pdf

[12] Open Networking Foundation, "SDN Architecture – Issue 1" TR-502, June 2014, available at https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf

[13] Open Networking Foundation, "SDN Architecture – Issue 1.1" TR-521, January 2016, available at https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-521_SDN_Architecture_issue_1.1.pdf

[14] Luis M. Contreras, Carlos J. Bernardos, Diego López, M. Boucadair, P. Iovanna, "Cooperating Layered Architecture for SDN", draft-irtf-sdnrg-layered-sdn-00, work-in-progress, March 2016.

[15] C.J. Bernardos, L.M. Contreras, "Multi-domain Network Virtualization", draft-bernardos-nfvrg-multidomain-00 (work in progress), March 2016.

[16] 5G-PPP Pre-standards-WG Issues Paper, https://5g-ppp.eu/wp-content/uploads/2014/02/5G-PPP-Pre-standards-WG-Issues-Paper-for-20-Oct-15-WS_final-edited.pdf

[17] C.J. Bernardos, A. Rahman, JC. Zúñiga, L.M. Contreras, P. Aranda, "Gap Analysis on Network Virtualization Activities", draft-irtf-nfvrg-gaps-network-virtualization-00 (work in progress), March 2016.

[18] NOXREPO.org. NOX. NOXREPO.org. [Online] 2016. www.noxrepo.org.

[19] Beacon Openflow. [Online] 2016. https://openflow.stanford.edu/display/Beacon/Home.

[20] Maestro Platform -A scalable control platform writen in Java which supports OpenFlow switches. code.google.com. [Online] 2016. https://code.google.com/p/maestro-platform/

[21] Project Floodlight. Project Floodlight - Open Source Software for Building Software-Defined Networks Project Floodlight - Open Source Software for Building Software Defined Networks. Project Floodlight. [Online] 2016. www.projectfloodlight.org/floodlight.

[22] Juniper Networks. OpenContrail. OpenContrail. [Online] 2016. http://opencontrail.org/.

[23] OpenDaylight.org. OpenDaylight. OpenDaylight. [Online] 2016. www.opendaylight.org.

[24] Weave. [Online] 2016. http://www.infoworld.com/article/2608941/application-virtualization/weave-project-knits-together-networks-for-docker-containers.html

[25] Flannel. [Online] 2016. https://coreos.com/blog/introducing-rudder/

[26] OpenBaton. [Online] 2016. http://openbaton.github.io/

[27] OpenSDNCore. Online 2016. http://www.opensdncore.org/

[28] OpenMANO. Online 2016. https://github.com/nfvlabs/openmano

[29] Hurtle, Online 2016, http://hurtle.it

[30] http://www.itu.int/en/ITU-T/jca/sdn/Documents/deliverable/Free-download-sdn-roadmap.docx

[31] NOXREPO.org. NOX. *NOXREPO.org.* [Online] 2016. www.noxrepo.org

[32] Beacon Openflow. [Online] 2016. https://openflow.stanford.edu/display/Beacon/Home.

[33] Maestro Platform -A scalable control platform written in Java which supports OpenFlow switches. code.google.com. [Online] 2016. https://code.google.com/p/maestro-platform/

[34] http://onosproject.org/ https://code.google.com/p/maestro-platform/

[35] Project Floodlight. Project Floodlight - Open Source Software for Building Software-Defined Networks Project Floodlight - Open Source Software for Building Software Defined Networks. Project Floodlight. [Online] 2016. www.projectfloodlight.org/floodlight.

[36] Juniper Networks. OpenContrail. *OpenContrail.* [Online] 2016. http://opencontrail.org/.

[37] OpenDaylight.org. OpenDaylight. *OpenDaylight.* [Online] 2016. www.opendaylight.org.

[38] Weave. [Online] 2016. http://www.infoworld.com/article/2608941/application-virtualization/weave-project-knits-together-networks-for-docker-containers.htmlhttp://www.infoworld.com/article/2608941/application-virtualization/weave-project-knits-together-networks-for-docker-containers.html

[39] Flannel. [Online] 2016. https://coreos.com/blog/introducing-rudder/

[40] OpenBaton. [Online] 2016. http://openbaton.github.io/

[41] https://www.tmforum.org/zoom/

[42] OpenMANO. Online 2016. https://github.com/nfvlabs/openmano

[43] Tacker project https://wiki.openstack.org/wiki/Tacker

[44] Hurtle, Online 2016, http://hurtle.it

[45] FP7 T-NOVA project. [Online] 2016. http://www.t-nova.eu/

[46] TeNOR Orchestrator. [Online] 2016. https://github.com/T-NOVA/TeNOR

[47] FP7 Mobile Cloud Networking Project. [Online] 2016. http://mobile-cloud-networking.eu

[48] Cyclops: Rating-Charging-Billing Solution for Cloud providers. [Online] 2016

http://icclab.github.io/cyclops/

[49] OpenEPC: Open Evolved Packet Core. [Online] http://www.openepc.com

[50] OpenAirInterface. [Online] http://www.openairinterface.org

[51] OPNFV: Open Platform for NFV. [Online] http://opnfv.org

[52] OPNFV: An Open Platform to Accelerate NFV, Linux Foundation Whitepaper, https://www.opnfv.org/sites/opnfv/files/pages/files/opnfv_whitepaper_092914.pdf

[53] OPNFV Wiki. [Online] https://wiki.opnfv.org/display/PROJ/Approved+Projects

[54] https://www.open-o.org/news/announcement/2016/11/open-o-bridges-gap-between-sdn-and-nfv-release-10

[55] NFV Release 2 Description. [Online] https://docbox.etsi.org/ISG/NFV/Open/Other/NFV(16)000274r3_NFV%20Release%202%20Description%20v10.pdf

[56] SCF 159.06.02 "Small Cell Virtualization: Functional Splits and Use Cases", January, 2016.

[57] SCF 083.05.01 "SON API for small cells", March, 2015.

[58] SCF 106.06.01 "Virtualization for Small Cells: Overview", June, 2015.

[59] SCF 154.05.01.02 "Virtualization in small cell networks. Translating NFV Concepts to SCN Functions", June, 2015.

[60] Microwave Information Model (December 2016 | TR-532), https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR-532-Microwave-Information-Model-V1.pdf

[61] Igor Faynberg, WG SEC Chairman - Presentation of the ETSI NFV SEC Working Group - Online: https://www.youtube.com/watch?v=uuwnovW92vI

[62] ETSI NFV SEC Working Group - ETSI Portal - Online: https://portal.etsi.org/tb.aspx?tbid=799&SubTB=799

[63] http://about.att.com/content/dam/snrdocs/ecomp.pdf

[64] https://www.sdxcentral.com/articles/news/att-will-launch-ecomp-open-source-2017/2016/10/