5G-PPP Software Network Working Group

# Cloud Native and 5G Verticals' services

February 2020

## 1 CONTEXT AND CHALLENGES

Network softwarization is focused on virtualization, decoupling the HW from the SW to lower the cost of network and service operation and reducing the time to market for new services while introducing higher flexibility and cost efficiency. In addition, virtualization of networking systems 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.

At the same time, cloud computing has disrupted the established order in many sectors. Using cloud resources, available on demand and billed for use, companies have been able to reduce the investment on their internal data centers. From now on, the competitiveness of a company depends directly on its capacity to quickly realize new ideas. Start-ups understand this well, so they rely on Cloud Native approaches to disrupt traditional sectors. It becomes obvious that to be innovative, the telco sector should follow the suit and facilitate migration to the Cloud Native technologies.

Cloud Native is an approach to build and run applications that fully exploit the benefits of the cloud computing model. The Cloud Native approach refers to the way applications are created and deployed, not where they are executed. It includes things like service architectures, infrastructure as a code, automation, continuous integration/delivery pipelines, observability/monitoring tools, etc.

Towards that direction, a research ecosystem of horizontal and vertical R&D actions is being formed, which examines the transformation from Cloud Ready to Cloud Native in the telco world. A carrier-grade enhancement to the Telco Cloud Native platform is required to provide better-than-IT performance in par or better than what is possible with classical telco platforms. Features like five-nine reliability, stateless microservices, etc. should be supported in the Telco Cloud Native. Fast virtualization and programmability across all networking domains: from core to edge and access, from wired/optical to wireless, cellular and satellite, is also considered.

As part of the 5G-PPP Initiative, the Software Network Working Group published a white paper, [1], in August 2019 analyzing the Cloud Native transformation in the telco ecosystem. The Software Network Working Group conducted a survey to collect technical inputs from 5G-PPP Phase 2 [7] and Phase 3 [8] projects on their: 1) supported Vertical use-cases, 2) adopted Cloud Native technologies, and 3) their architecture patterns. Here, we present a summary of [1].

### 1.1 Importance of Verticals in 5G

5G is often boasted for its ability to meet the development needs of Verticals, reflecting different sectors of the economy, that often use specific solutions for their communications. These sectors are extremely diverse: transport, media, smart city, agriculture, industry of the future, augmented reality, security or medical applications. It is both a question of taking advantage (i) of the significant economies of scale that technologies such as 5G allow, when they are launched, compared to specific developments in a given sector, as well as (ii) of 5G functionalities capable of guaranteeing performance in terms of throughput, latency, reliability, etc. through "network slicing".

### 1.2 Benefits of Cloud to Verticals

More than ever before, Verticals are competing at an accelerated pace, each trying to get the biggest market share in its domain. There are currently Verticals industries that rely on 4G systems and that will continue to exist in 5G and beyond. Those Verticals that built their software-based systems on "legacy" technologies will need to adapt or migrate towards newer technologies as 5G systems mature. On the other hand, newcomers to appear after 5G becomes available, must envision building their industry in the most flexible and adaptive way leveraging 5G capabilities. Features like short time-to-market thanks to CI/CD, mix and match proprietary code with open-source, "as a service" usage of all capabilities of the network and many other benefits that Cloud Native principles offer will keep Verticals in the competition and allow them to deliver valuable quality services in a short time.

### 1.3 Are Verticals ready?

Generally, Verticals are the main beneficiary of 5G systems, even currently enhanced Mobile Broadband (eMBB) use-cases are mainly targeted. They are intended to adhere to an architecture based on Cloud Native principles to leverage the ubiquity of Cloud Computing. It is then important to ask where do the Verticals stand with respect to Cloud Native? How far have they gotten in using this technology? Where is the potential? What are the challenges or impediments? And what is missing to bring them closer to embracing Cloud technologies and Cloud Native services?

### 1.4 What technologies do Verticals use today?

The 5G-PPP Software Network Group has conducted a survey to collect technical inputs from the different participating projects about their use cases, the virtualization technologies they use, and the architecture patterns they follow. The method consists of finding commonalities between the Verticals /use cases and then cluster the Verticals accordingly. This helps to position the Verticals with respect to Cloud Native adoption. The use-cases are grouped per industry to find similarities or differences per industry. Once this was completed, a technological analysis was done to cluster Verticals according to the technologies they use in the implementation of their use cases. Then Verticals partners will be involved, in a follow-up survey in sping 2020 to validate technologies and pattern choices.

## 2 EVOLVING FROM VIRTUAL NETWORK FUNCTION TO CLOUD NATIVE NETWORK FUNCTION

Physical Network Functions (PNF) and Virtual Network Functions (VNF) are likely to be with us for at least another decade. The only feasible approach for Cloud Native telecom is to offer an evolution of PNFs and VNFs to become CNFs (Cloud Native Network Function). This mirrors how enterprises are moving their monoliths to Kubernetes and then (often slowly) refactoring them into microservices. For this to be economic, there need to be incremental gains in resiliency, scheduling, networking, and development velocity as more network functions become Cloud Native.

Moving network functionality from physical hardware to encapsulating the software in a virtual machine is generally easier than containerizing the software. In fact, many network function virtualization VMs rely on kernel hacks or otherwise do not restrict themselves to Linux kernel user-space. They also often need to use Data Plane Development Kit (DPDK) or Single root I/O virtualization (SR-IOV) to achieve adequate performance. Containers provide nearly direct access to the hardware with little or no virtualization overhead. However, they expect containerized applications to use the stable user-space Linux kernel, not to bypass it.

There's been a similar journey on the software stack deployed currently in telco eco-system and at 5G-PPP level. Most of the prototypes and the project realization moved from a pure OpenStack ecosystem, derived by ETSI MANO, to include the capability to run Kubernetes on top of either bare metal or any cloud where the intelligence is still centralized in VNF Manager (VNFM) like box. Whereas in the future version, most functions will be CNFs running on Kubernetes (denoted in short K8s). If some VNFs haven't been ported to CNFs yet, they can use technologies like KubeVirt and Virtlet to be managed by Kubernetes. This transformation has been captured in Figure 1 below.

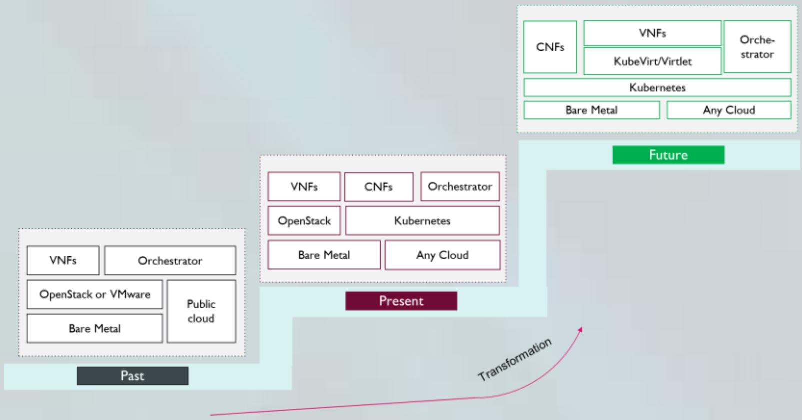


**Figure 1: Evolution of the VNFs toward container and Cloud Native network functions (a revised version of the one published in CNCF)**

## 2.1 Past: MANO Centric

The NFV MANO is an architectural framework described in [1]. Different functional blocks are identified: 1) Virtualized Infrastructure Manager (VIM), 2) NFV Orchestrator (NFVO), and 3) VNF Manager (VNFM).

Referring to a recent analysis published in [5], some concerns around MANO are pointed out. These concerns are less about the architecture, and more about the actual implementation:

- *Multi-domain orchestration*: In many cases, multi-VIM or multi-cloud features are needed to support diverse use-case case e.g. data-plane in one edge cloud and control-plane in public cloud. Having an OpenStack in a multi-cloud environment is very hard. However, Kubernetes can be run across different clouds having consistent orchestration between clouds.
- *Manage multiple vendors' virtualized network functions (VNFs) in a consistent way*: While multi-vendor orchestration is supported, most VNFMs are offered by the same compagnies that supplies the VNF recreating the hard dependency 1:1 between hardware and Element Managed System that exist in physical world.
- *Build-to-order MANO*: At architecture level, it is possible to combine the different blocks from different suppliers. But, at implementation level, it depends on the use-case and on the orchestration system, making hard to combine different blocks from different vendors.

## 2.2 MANO enhancements towards "Cloud Native" and "PaaS"

As Cloud Native transformation unfolds, it becomes evident that at least for the transition period (and perhaps beyond that), both traditional PNFs, VM based VNFs and container based VNFs will have to co-exist seamlessly within the context of the same network services. Other types of VIM like Kubernetes are added in the architecture via VIM type plugins while keeping the orchestration intelligence centralized in a VNFManager [4][2].

These enhancements towards "Cloud Native" and "PaaS" are discussed in ETSI IFA029 [4] draft where the concept of VNF *common* and *dedicated services* has been introduced. These VNFs are instantiated inside the PaaS and expose capabilities that are consumed by the Network Services (composed by consumer VNFs) that run over the PaaS:

- *VNF Common Service*: common services or functions for multiple consumers. Instantiated independently of any consumer. For example, a generic monitoring service framework.
- *VNF Dedicated Service*: required by a limited set of consumers with a specific scope. Instantiated dependently of their consumers (when required by a consumer) and destroyed when no relation exists with any consumer.

The concept of PaaS service catalogue is introduced to enable their re-use for instantiating multiple VNFs. This catalogue is a set of PaaS service descriptors for individual PaaS services. Before a consumer VNF is instantiated, the VNFM request the discovery of the PaaS service descriptor inside the catalogue and creates a temporary VNFD. The latter is composed by the original consumer VNFD and the PaaS service descriptor allowing the instantiation of the services that compose the PaaS.

## 2.3 (Possible) future: Kubernetes-centric

### 2.3.1 Looking into the future is challenging!

Going forward, a legitimate question to ask is whether a mature container orchestrator engine (e.g., Kubernetes) itself can become a fully-fledge VNF orchestrator? Obviously, if a VNF is implemented using a CRI-O (Container Runtime Interface) and deployed on Kubernetes, an intent-driven orchestrator such as Kubernetes is a natural choice for VNFM. In principle, Kubernetes can manage resources across the board via its innate Custom Resource Definition (CDR) mechanism. We point that at this stage it is difficult to predict the exact adoption path of the Cloud Native technology. It is reasonable to assume that Cloud Native capabilities will be *explored* and *leveraged* via technologies such as Kubernetes while legacy will continue playing an important role. Hereunder are several topics that need to be explored.

### 2.3.2 Kubernetes as 5G default orchestrator?

To ensure virtualization technology neutrality and portability across different use cases and future environments, Kubernetes is used as an industry de facto standard container orchestrator. Kubernetes can be either deployed on the bare metal servers or on top of some virtualization technology. Some predict Kubernetes can become the operating system for 5G networks. Indeed, thanks to its flexibility, such an orchestration engine will accelerate the adoption of container network functions (CNF) by operators.

### 2.3.3 Carrier-grade capabilities

Kubernetes already offers a myriad of options for networking and with the adoption of Network Service Mesh as a CNCF sandbox project, is poised to support nearly all other use cases:

- The CNCF-hosted project Container Network Interface (CNI) supports over a dozen networking technologies, including Multus and DANM, which has seen use in operator applications
- Network Service Mesh (a new sandbox project) flexibly creates layer 2 or 3 network endpoints ("virtual tunnels") similarly to how Envoy/Istio work with TCP and HTTP
- Kubernetes implements cluster functionality as a Custom Resource Definition (CRD)
- Kubernetes supports several carrier-grades features such as bridging VPNs, high performance vSwitch, and connecting to PNFs
- Kubernetes supports IPv6 with dual-stack

# 3 CONCLUSIONS AND LESSONS LEARNT

1) Projects clustering and patterns:
   a. Most project prototypes evolved from ETSI MANO relying on an Openstack VIM exclusively to include Kubernetes - on bare metal and public cloud - as a new VIM in parallel to Openstack.
   b. Meanwhile, they kept orchestration intelligence centralized in a VNFM-like box.
   c. Therefore, only a few of them fully exploited Kubernetes as a complete autonomous platform with its own orchestration intelligence able to host both Containerized Network Functions and classical VM-based VNFs.
2) Projects evolution to Cloud Native:
   a. We acknowledge a *reluctance* for using fully Cloud Native design provided by e.g. Kubernetes. This reluctance has been analyzed to extract the underlying reasons motivating projects to select this intermediate step where Kubernetes is considered only as a VIM. These reasons are presented as the barriers to adopt the Cloud Native patterns.
   b. These barriers are essentially *the lack of standard* and *technological maturity*, implying *human adaptation resistance*. The shift to the Cloud Native mindset, especially for the telecom sector, will follow a phased approach (re-architecting the current network function, tools for automation and orchestration at function level, etc.).

# 4 REFERENCES


[1] 5G-PPP White Paper, "Cloud Native and Verticals' services, 5G-PPP projects analysis", August 2019, Available: https://5g-ppp.eu/wp-content/uploads/2019/09/5GPPP-Software-Network-WG-White-Paper-2019_FINAL.pdf

[2] ETSI NFV, http://www.etsi.org/technologies-clusters/technologies/nfv

[3] ETSI GS NVF-EVE 011 V0.0.11, Specification of the Classification of Cloud Native VNF Implementations, work in progress, 2018.

[4] ETSI GS NFV-IFA 005, Network Functions Virtualization (NFV) Release 3; Management and Orchestration; Or-Vi reference point – Interface and Information Model Specification, 2019.

[5] ETSI GR NFV-IFA 029 V0.8.0, Report on the Enhancements of the NFV architecture towards "Cloud Native" and "PaaS", work in progress, 2018

[6] Roz Roseboro, "MANO & the Future of CSP Automation", Heavy Reading Reports, August 2017 (https://www.giiresearch.com/report/heav549643-mano-future-csp-automation.html)

[7] 5GPPP Phase 2 Projects - https://5g-ppp.eu/5g-ppp-phase-2-projects/

[8] 5GPPP Phase 3 Projects - https://5g-ppp.eu/5g-ppp-phase-3-projects


**Editor**: Dr Bessem Sayadi, NOKIA Bell-Labs France, bessem.sayadi@nokia-bell-labs.com