Network Function Virtualization: Take On the Challenge Minimizing Risks

White Paper

28 December 2016

By Angelo Baccarani (Product Manager – NFV Service Assurance)

1 Executive Summary

Telecom industry has recently started a journey toward a deep transformation that is affecting the architecture of all the networks providing any type of service, in both fixed and mobile domains.

The Telecom community (Communication Service Providers, Network Equipment Manufacturers, System Integrators, OSS/BSS vendors…) refers to such revolution with the term "virtualization" that, in its very simplified meaning, can be translated as *"running any network function in software on industry-standard hardware*". The reason why it is a *revolution* is that such concept is very different from the current approach where all the main network equipment (routers, central offices switches, firewalls, gateways, PABX…) requires a dedicated hardware.

It is interesting to underline that the concept of virtualization itself is nothing new: it is widely adopted in Data Centers (and, more widely, in IT domain) since more than 10 years based on the following standard definition: *in computing, virtualization is a broad term that refers to the abstraction of computer resources. Virtualization hides the physical characteristics of computing resources from their users, be they applications, or end users. This includes making a single physical resource (such as a server, an operating system, an application, or storage device) appear to function as multiple virtual resources; it can also include making multiple physical resources (such as storage devices or servers) appear as a single virtual resource..."*

However, the news is in adopting such concept to build the communication infrastructures, i.e. the so-called Access and Core networks that are the heart of the fixed and mobile services each of us use every day.

Problem is that although the concept is clear, its adoption for Telecom applications poses various challenges, as I will explain this paper.

1.1 The importance of Virtualization in Telecom

The first question is why today the telecom industry thinks it must go toward that direction. Besides the reasons related to an allegeable cost savings (due by the adoption of

commercial hardware vs. expensive proprietary devices) and limiting the so-called *vendor lock-in*, the main driver is the extreme flexibility (a.k.a. *agility*) the virtualized architectures promise in creating new services and offering the capacity where (and when) you need.
Having a network whose functions can be defined by means of software components running on a set of standard devices, instead of having a tight matching between a set of functions and its hardware, allows introducing new services in a more dynamic way than today. Additionally, it is also possible to allocate computing resources dynamically while today the network capacity is normally *over-provisioned* to manage the peak traffic hours, with the result that, if a network portion is under-loaded, its hardware resources cannot be re-allocated to support other areas that can be under-loaded at the same time.

Try to think how the provisioning of the legacy voice service worked in the past, in a fixed network. The network was split in areas, each with a specific capacity (in terms of simultaneous calls that can be supported) defined by the number of *voice circuits.* If, in one area, the users exhausted all the voice circuits, it was then no possible to make any further call even if other circuits were probably available in other areas.
The example above refers to a very legacy service that is the voice call on a fixed network: but then mobile networks came and the problem moved on the capacity of the radio channels managed by a single tower (a.k.a. *cell)* where the complexity of the so-called capacity planning increased due to the mobility concept. Things became more complicated with the introduction of the mobile data services where network must provide a dynamic capacity for users that make voice calls but also access an incredible amount of data services while in mobility.

Therefore, here is what operators are facing today: a continuously increasing demand of capacity for accessing data services that CSP cannot satisfy just over provisioning the network due to its extreme (and unpredictable) dynamicity. Try to think to the impact that new video services, enabled in the near future by new 5G technologies, can have on the current networks, as well as the introduction of the Internet of Things where millions of devices will ask for reliable data connectivity like the so-called connected cars.

Network virtualization promises to help in building a network infrastructure whose Quality of Service can be dynamically granted depending on the services accessed by users at a specific time, without forcing (or, at least, limiting) the CSP to continuously increase the capacity upfront like today (i.e. over-provisioning).

2 Terminology: Virtualization, NFV and SDN

Strictly related to network virtualization there are some very important concepts used very often as synonyms while they are very different each from the other. It is therefore important correctly defining them.

- **Virtualization:** it means run in software (i.e. into a Virtual Machine) a function that is traditionally executed on dedicated hardware. Each function is implemented into a single *Virtual Network Function* (VNF), running separately (e.g. DNS Server) one from the other. Hardware (the so-called *server farm*) is shared across multiple applications.
- **Network Function Virtualization (NFV):** it combines the functions of multiple VNFs to provide network services (e.g. vEPC, vIMS…) under the control of a single software called Orchestrator. Such component is the key differentiator between *simple virtualization* and *full NFV*.
- **SDN:** it introduces the possibility to change the rules used by the switches to route traffic, through a specific software component (*SDN Controller*) and a protocol (*OpenFlow* is the de-facto standard) that physical (or virtual) switches must support.

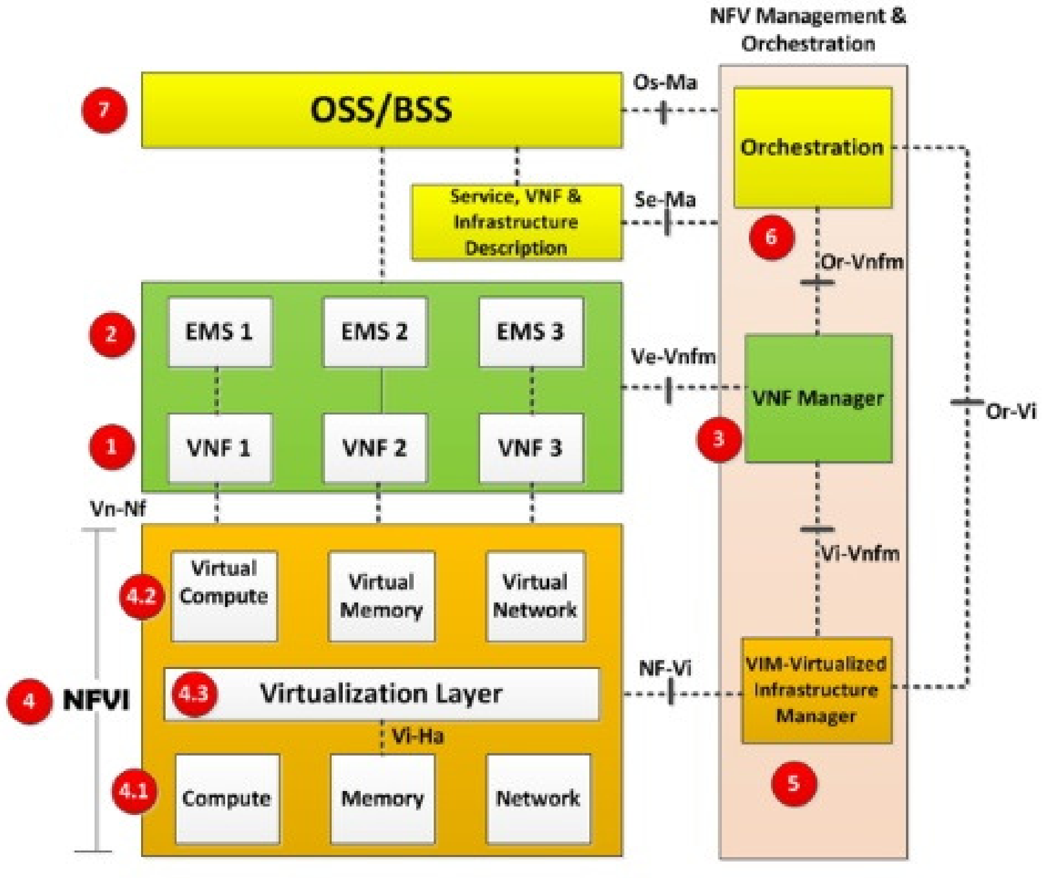
Each of the concepts above requires a different level of complexity and we can say that NFV requires virtualization, but it is possible to implement virtualization without NFV. Furthermore, NFV is highly complementary to SDN but not dependent on it (or vice-versa). NFV can be implemented without SDN, although the two concepts and solutions can be combined to potentially get greater value.

While today many CSP has already started to adopt virtualization to move some specific functions onto commercial hardware (and, consequently, from Central Offices / Switch Rooms into Data Centers), the road to NFV and SDN is much more complex due to the high number of interoperable components that must be selected and tested before going into live production.

It is also worth to underline that NFV is a very young concept: although there are many vendor consortiums mainly trying to certify the interoperability among the various components, ETSI is the only international organization releasing the standards, the first ("Phase 1") being available in January 2015 followed by a "Phase 2" available since October 2016.

3 The ETSI NFV Framework

In order to understand the complexity of the NFV technology, it is necessary to start from the definition of the overall architecture and its components as defined by ETSI in the following diagram:



Next paragraphs provide a short explanation of each functional block.

3.1 VNF (Virtual Network Function)

VNF is the basic block in the NFV architecture: we can view it as the *virtualized network element*. For example when a router is virtualized, we call it “Router VNF”. Even when one sub-function of a network element is virtualized, it is called VNF. For example in router case, various sub-functions of the router can be separated VNFs (called VNF-C, VNF components) which together function as virtual router. Other examples of VNF include firewalls, IPS, GGSN, SGSN, RNC, PCRF, S/Gw etc.

3.2 EMS (Element Management System)

The physical networks already include such component since many years but in this case ETSI applies it to the specific VNF. It is responsible for the management of VNF operation, in the same way as EMS manage their respective physical network elements, and provides FCAPS (Fault, Configuration, Accounting, Performance and Security) functions.

EMS may manage its VNFs through proprietary interfaces: there may be one EMS per VNF or an EMS can manage multiple VNFs. EMS can be also a VNF. It is very common that the same VNF vendor provides also the EMS.

3.3 VNFM (VNF Manager)

A VNF Manager (a.k.a. as *VNF Orchestrator*) manages a single VNF or a set of multiple VNFs, providing the so-called *life cycle management* of VNF instances: it setups, configures, maintains and tears down the VNFs.

A VNF manager can do the same functions as EMS but through open interface/reference points proposed in NFV architecture named *Ve-Vnfm*. Such interface is further split in twos: *Ve-Vnfm-EM* if VNFM is logically connected to an EMS or *Ve-Vnfm-vnf* if VNFM “speaks” directly with the VNF.

A key function of VNFM is to implement one of the most interesting promises of NFV: the so-called *network elasticity.* Today the capacity of a network is almost statically defined during the deployment, according to the *capacity planning* design phase. If more capacity is required, normally it means to add additional hardware. NFV changes such approach introducing the *automatic scaling*: the single VNF can increase or decrease their capacity depending on the needs in a specific time. Such scaling can be achieved through *scale up/down* (assigning more computational resources to a specific VNF) or *scale out/in* (respectively, deploying more VNF or remove them) within the limits provided by the available hardware infrastructure.

Problem is that VNFM is still under definition from a standardization standpoint because ETSI has just described but not specified its functions in the recent Phase 2 specifications, without providing details about the protocol to carry such information.
Additionally it is also still controversial the split of functions between VNFM, EMS and the Service Orchestration (the component showed as no. 6 in the diagram above).

Consequence is that, in absence of clear standards, majority of VNF vendors are providing their own VNFM while some NFVI vendors claim to integrate multiple VNFMs, leaving to the CSP the complexity of deciding to implement one VNFM for the whole NFV or integrating multiple ones.

3.4 NFVI (Network Function Virtualization Infrastructure)

It is the environment hosting all the VNFs, including physical and virtual resources as well as the virtualization layers as described below.
Note that NFVI are a set of components that are in common with the virtualization approach, i.e. you need it also if you want just virtualize some functions even if without implementing the full NFV paradigm.

3.4.1 Compute/Memory, Storage and Networking

It is the physical part of NFVI: they are the hardware resources available to run the VNF so all the virtual applications are instantiated on them, normally by means of COTS (commercial-off-the-shelf) servers.
The above realizes the deep meaning of virtualization, fully decoupling the applications from the physical environment hosting them. Therefore, we can define NFVI describing the underlying hardware as a set of compute, storage and networking components.

All the major vendor of hardware are now providing specific products designed for virtualized environments like Blade Servers and SAN (Storage Area Networks). Although focus now is on software, hardware is still very important when dealing with performances and reliability that, for telecom applications (differently from IT) must guarantee the famous "*five 9s*" (99.999% of availability).

3.4.2 Virtual Compute/Memory, Storage and Networking

It is the virtual part of NFVI, where the physical resources are abstracted into virtual resources that are ultimately utilized by VNFs. Very often this layer is indicated as Virtual Switch, being it its main component.

A Virtual Switch allows the VNF to communicate together: it does for VNF what a physical switch does for physical servers. Examples are Open vSwitch (OVS), Wind River Accelerated Virtual Switch (AVS), Cisco Nexus 1000v and 6Wind OVS: they are all software implementation of a switch that, as like the other VNF, can run on commodity hardware.

Another component belonging to this section is the Virtual NIC (vNIC) through which each VNF communicates with the Virtual Switch. Normally, all the major Operating Systems provides a concept of vNIC.

3.4.3 Virtualization Layer (a.k.a. Hypervisor)

This layer is responsible for abstracting physical resources into virtual resources. The common industry term adopted is *"hypervisor"*. It decouples software from hardware, enabling the software to progress independently from hardware.

Each guest (i.e. the VNF) can run its own operating system, to which it appears the virtual machine has its own CPU and RAM, i.e. it seems it has its own physical machine even though it does not. To do this efficiently, the hypervisor requires support from the underlying processor (a feature called VT-x on Intel, and AMD-V on AMD).

Note that if there is no virtualization layer, one may think that VNFs can run on physical resources directly: in such case, we cannot speak about virtualization nor NFV but we would call such function as PNFs (Physical Network Functions). Consequence of that if that the hypervisor is a key mandatory component for both virtualization and NFV.

There are two types of hypervisors:

- Type 1 (Native Bare Metal)
  A Type 1 hypervisor (sometimes called a 'Bare Metal' hypervisor) runs directly on top of the physical hardware. Each guest operating system runs atop the hypervisor. VMware ESX/ESXi is an example (so no O.S. is required).

- Type 2 (Hosted):
  A Type 2 hypervisor (sometimes called a 'Hosted' hypervisor) runs inside an operating system, which in turn runs on the physical hardware. Each guest operating system then runs atop the hypervisor. KVM is an example and requires a Linux distribution installed.

As listed above, examples of hypervisors for NFV are Linux KVM and VMware ESXi that are now de-facto standards. There are also other hypervisors (like Xen and Microsoft Hyper-V) but adopted in pure IT virtualized applications (i.e. Data Centers) rather than for network functions.

An alternative way to virtualize physical resources to multiple VNF is *Containers* (a.k.a. *Dockers)* technology: differently from hypervisors that require each VNF to contain the image of its own Operating System, a container can share a single O.S. among multiple VNF that are consequently much smaller. Because such technology is not yet common for NFV, being still in experimental stages, I will no further describe it in this document.

3.5 VIM (Virtualized Infrastructure Manager)

We can view it as the *management system* for NFVI: it is responsible for controlling and managing the NFVI compute, network and storage resources within one operator's infrastructure domain.

It is also responsible for collection of performance measurements and events related to the hardware infrastructure.

Example of VIMs are Openstack (and its commercial distributions mainly RedHat, Mirantis and Canonical), nowadays a de-facto standard with almost 50% of market share, and VMware.

3.6 NFVO (NFV Orchestrator)

Probably currently the most controversial component in NFV, it is also referred as *Service Orchestrator*. Its main function is *to combine multiple VNF to provide a service*.

To perform such tasks it issues commands to the VNFM(s) to generate, maintain and tear down network services composed of VNF themselves. If there are multiple VNFs, orchestrator will enable creation of end-to-end service over multiple VNFs.

NFV Orchestrator is also responsible for global resource management of NFVI resources. For example managing the NFVI resources i.e. compute, storage and networking devices among multiple VIMs (if present) in network.

Note that the Orchestrator performs its functions by <u>not</u> talking directly to VNFs but through VNFM and VIM: let me assume there are multiple VNFs that need to be chained to create an end-to-end service. One example of such case is a virtual Base Station and a Virtual EPC: they can be from same or different vendors so there will be a need to create the complete service using both VNFs. This would require a Service Orchestrator to talk to both VNFs and create the final end-to-end service.

The controversy about such component is due to lack of standards that have forced the vendors to provide their own version of NFVO that should integrate multiple VNFM even in absence of a standard way to define a service.

Currently there are on the market about 20 different Orchestrators and only few seems capable to integrate VNF of multiple vendors…not a good situation for CSP if they want to avoid the famous *vendor lock-in.*

The only initiative coming from an international standard organization is currently Open Source MANO (OSM) but first specifications just came in October 2016. It is project hosted by ETSI to develop an open source NFV MANO software stack, aligned to its proposed architecture. The project was first demonstrated as an operator use-case at the Mobile World Congress 2016. Interestingly, OSM makes use of some other open-source projects – OpenMANO and RIFT.io – along with OpenStack and Ubuntu JuJu. Considering the reuse of these projects, it is not surprising that both Telcos (such as Telefónica, British Telecom, Telekom Austria Group, Korea Telecom, and Telenor) and vendors (such as Intel, Mirantis, RIFT.io, Brocade, Dell, RADware and others) support OSM.

Anyway, there are vendors (mainly Network Equipment Manufacturer and system Integrators) that, because they cannot wait the long times required by organizations like ETSI for fully publishing their standards, are pushing their own solutions (just in alphabetical order, list does not reflect their respective market share):

- Alcatel-Lucent CloudBand Management System
- Amdocs Network Cloud Service Orchestrator
- Cisco Network Services Orchestrator
- Ericsson Cloud Manager (ECM)
- HPE NFV Director
- Huawei CloudOpera Orchestrator
- Netcracker (part of NEC) RT MANO Network Orchestrator
- Nokia CloudBand Network Director
- Oracle Network Service Orchestration
- ZTE vManager

Problem of such solutions is that there is no guarantee that a VNF working under one of such MANO will work also with another because each requires a different *VNF Descriptor* (by the way, ETSI has recently standardized the TOSCA format for it but its adoption is just at the beginning) . Furthermore, many of the vendors listed above are providing not only the MANO components but also the complete NFVI. Consequently, vendors should certify their VNF for the full NFV hosting environment and this can be very time and resources consuming.

Finally, there are initiatives coming from the Open source community: however, they do not look products ready for production environments but rather just *frameworks* that vendors can adopt for building their own product still granting a certain level of interoperability.

- Open-O: under Linux Foundation, China Mobile is driving this initiative to develop an Open Orchestrator for NFV global management and automatic deployment. It focuses on the VNFM and Orchestrator components of ETSI NFV. The project is still in very nascent stage, and not much information is available.
- OpenStack Tacker: is an OpenStack project focusing on building an Open NFV Orchestrator and a general purpose VNF Manager to deploy and operate Virtual Network Functions (VNFs) in an 'OpenStack-managed virtual infrastructure.'
- OpenMANO: a project released by Telefonica. As of today, OpenMANO is a very basic implementation and not suitable for commercial deployment but it has been included into the ETSI OSM initiative.
- RIFT.ware: RIFT.io introduced such tool and claims the release 4.0 a *complete solution for NFV management and orchestration*. They released it also to the open source community by the end of 2015.
- Open Baton: a ETSI NFV compliant MANO framework which can be used by researchers around the globe to build their own 5G/SDN/NFV/MEC testbeds, as well as to create the knowhow required for emerging 5G standards with initial Proof of Concepts (PoC). The Fraunhofer Institute for Open Communication Systems, or FOKUS, designed such tool so it comes from a public Research Institute in Germany.

3.7 OSS / BSS

We cannot consider such components as strictly related to NFV, because the Telecom industry defined and adopted it since many years. Just as reminder, OSS (Operation Support Systems) deals with network management, fault management, configuration management and service management (a.k.a. FCAPS that stands for Fault, Configuration, Accounting, Performance and Security). BSS (Business Support Systems) deals with all the software for customer management, product management, order management, service fulfillment i.e. everything directly affecting the revenues.

Problem in NFV is that current OSS/BSS requires a deep upgrade to manage both physical and virtualized network functions and to integrate with the NFV Management and Orchestration (a.k.a. MANO components showed as no. 3, 5 and 6 in the diagram above) through interfaces that are not yet fully standardized.

4 Communication Service Providers Challenges toward NFV

For a CSP, adopting (or also only evaluating) an NFV solution can be a very long and difficult process due to the disruptive nature of such solution in regards to the architectures deployed today. I therefore provide here a list of steps that CSP should carefully evaluate in order to define their plans, knowing since the beginning that this will be a journey with a mix of wins and failures before reaching an acceptable level of performances.

Note that the next paragraphs are related to the adoption of NFV, not of simply virtualization: referring to the ETSI diagram above, this means to adopt not only a virtualized environment where network functions will run as software components, but also the MANO tools that will allow to automatize as much as possible the VNF management (a.k.a. *VNF lifecycle management*).

4.1 Define your Key Business Objectives

First step is to define both your *reasons* and the *expectations*: in other words, CSP must have a clear driver (the *trigger)* to move toward such direction and the respective benefits that must be both *defined* and *measurable* (e.g. through KBO/KPI/KQI). "*Because everyone is talking about it*" is not an option.

Trigger can be the introduction of a new technology that would require a hardware upgrade like VoLTE or a platform to provide IoT services. Alternatively, it can be the replacement of old hardware becoming obsoleted or too expensive to maintain. For sure, it should be limited to a specific portion of the network: given the status of the NFV technology, thinking to upgrade the entire network to NFV all at once is not sustainable because it will be too disruptive on the service continuity.

4.2 Define What to Virtualize Through NFV

Immediate consequence of the step above is the definition of the network functions to virtualize. Examples can be the deployment of VoLTE and/or IMS nodes, vEPC or virtual CPEs to provide services to Business Users.

During this step important is to define how the new NFV infrastructure will connect to the legacy network. Remember: your network will be *hybrid* (i.e. a mix of physical and virtual) for many years ahead…

Another topic CSP must carefully evaluate is to virtualize Control Plane functions only or also User Plane. The latter introduces an additional level of complexity due to the performances that in a software-only-virtualized-solution can be challenging to keep at the same level of their physical nodes based on dedicated hardware. As example, industry is still debating about virtualizing Media Gateways functions requiring high-performance voice transcoding that today dedicated processors (DSP) provide.

4.3 Plan the Service Assurance Tools

It is extremely important CSP assign to the Service Assurance tools the same level of importance as like as the NFV infrastructure and components, but I dedicate a full section of this document to this topic.

4.4 Evaluate the Impact on Organization

CSP must be conscious that NFV is disruptive not only from a technical standpoint but mainly from an organizational point of view. Today in CSPs' organizations there is still a clear demarcation line between what is "IT" and what is "Network Management", although such line is already blurring since the advent of NGN networks that brought *IP into Telcos.*

However, NFV will force the complete convergence between IT and the Network: we have already underlined that virtualization is a concept coming from the technology adopted in Data Centers since many years. NFV will add the "network dimension" to virtualization but the big dilemma for the CSP organization is: who will manage the NFV infrastructure ?

It can be the IT or the Network (such as OSS, Operations, Engineering) departments. Problem is that neither of them has the complete set of competencies as required: the IT can be expert of the hardware and virtual infrastructure but lack the experience in Network Management. The others are exactly in the opposite situations.

It is therefore evident that CSP requires a new organizational model, with competencies coming from both the domains. Do not under-evaluate this problem: soon or later, you can have in place the best technical solution for NFV to find out that…no one can manage it !

4.5 Define the Approach

Looking to the ETSI NFV framework diagram it is clear that such technology is very wide and complex for most service providers to implement alone. Each block in the diagram requires the selection of a solution and everything must be managed (*orchestrated*, in NFV terminology) by MANO. Remember that one of the promises of NFV is *to avoid the vendor lock-in* so thinking to get everything from one vendor is not the best approach, although the safer one.

I can identify two possible approaches to follow: I define the first as "*full homemade*" and the other as "*system integrator based*".

4.5.1 Full homemade NFV

In the *full homemade* approach, CSP is fully responsible to first define the full framework, then select the participants' vendors that must comply with the defined framework and finally test the individual functional blocks to check their interoperability.

Below are the three main examples of such approach:

- AT&T released the structure and philosophy behind ECOMP (Enhanced Control, Orchestration, Management and Policy) within its "Domain 2.0" software platform for cloud computing and network virtualization, involving internal development resources and vendor partners

- Verizon released a *SDN–NFV reference architecture manifesto* co-authored with Cisco, Ericsson, Hewlett Packard Enterprise, Intel, Nokia, Red Hat and Samsung. It includes their End-to-End Orchestration (EEO) control function and is a credible alternative to the platforms provided by AT&T and Telefónica + ETSI.

- Telefónica presented its OpenMANO open source project at the 2015 Layer123 event, but early in 2016 OpenMANO morphed into Open Source MANO (OSM), a broader ETSI initiative. OSM's goal is to "*accelerate [industry] convergence on a telco-ready, production-quality, [virtualized infrastructure manager (VIM)]-independent*" MANO stack. The stack comprises an NFV orchestrator and a virtual network functions manager, together with 'service orchestration', and should minimize barriers to VNF developers and reduce the time and expense of integration. Note that OSM today has the backing of ETSI and close to 30 members and participants.

I am very well convinced that the reasons of all such approaches is that they started very well ahead of the availability of any standard. Another advantage is that they can keep the full control of their NFV platform. Risk on the long term is that the vendors will not follow such design principles, if the standards will divert too much.

4.5.2 System Integrator-based NFV

This approach requires for sure much less resources than the *full homemade* and, because based on selecting a System Integrator (S.I.) as single point of contact for the whole NFV project, also Tier 2/3 operators can afford it.

In reality, there is no need to ask a S.I. to take care of the whole project: CSP can still select independently the hardware infrastructure and the virtualization layer based on the competencies already available internally while leaving to the S.I. all the selection of the VNF and the MANO components.

Advantage of such approach is that CSP can rely on the so-called *ecosystem* already pre-certified by the single S.I. that will provide the set of VNFs and respective management functions whose interoperability have been already tested.

Cons is that CSP is not fully free in the components selection, being limited to the ones listed in the S.I.'s ecosystem. Furthermore, many big S.I. are not 100% independent because also vendor of NVFI and VNF so risk of the *vendor lock-in* can show up.

4.6 Laboratory PoC

CSP can conduct into its laboratory the first testing of the possible solution. In this phase CSP can execute both functional (*does it work ?*) and performance (*how much is it good ?*) testing in simulated scenarios, to verify the interoperability of the various components including the VNF lifecycle management and the automatic scaling.

Note the importance of Service Assurance applications in this phase (at minimum, troubleshooting tools) because it can help also in the verification process, i.e. verifying everything is running correctly.

4.7 Field PoC

The real validation of the solution should come only after at least a small-scale deployment: if CSP reaches this point, it means the solution is in a very advanced evaluation phase. However, surprises can come because traffic is now real and no more simulated. In this phase, CSP can compare the performances of the NFV deployment with similar functions still available in physical architecture (if available).

CSP can also now further expand the Service Assurance tools tested in the laboratory: ideally, CSP should integrate the monitoring of the NFV network into the existing S.A. applications to provide an end-to-end view across both physical and virtual components.

5 Service Assurance Challenge in NFV

A key challenge that Service Assurance (S.A.) poses to CSP when moving to NFV is its extreme dynamicity. In traditional physical networks, the nodes are static: although the traffic usage can vary a lot from time to time, the devices to monitor are pre-defined. Best example is the mobile network, where the locations of the users and the type of accessed services (voice, video, data…) determine the impact of their traffic on the network infrastructure.

Current OSS and Monitoring Systems know *how* and *what* to monitor. In other words, they know *how* to connect to the network, regardless we are speaking about getting performance/fault indicators from nodes' EMS (e.g. OSS tools) or sniffing live traffic (e.g. passive probes). They also know *what* information to get depending on the so-called monitor points: OSS periodically collect a specific set of indicators/faults events while probes look to a list of protocols. Then, a Service Quality Monitor (SQM) tool could correlate the info coming from the network (i.e. QoS) with the ones coming from the observation of live user traffic (i.e. QoE) to provide the full picture to the CSP about how its customers are perceiving the services provided and how network performances are affecting it.

5.1 Network will Become Dynamic

However, NFV (if fully implemented so I am not speaking about simply *virtualization)* will completely change this approach.

First of all the *network nodes* are not physical but virtual so there is no more a 1:1 mapping between them and the hardware where they run. Then, they can be dynamically instantiated and removed depending on the traffic conditions: VNFM (or the VNF-EMS, but this is still under debate…) can decide at any time to instantiate more VNF or remove them without any human intervention (i.e. *network elasticity* or *automatic scaling*). Summarizing, the network itself will become dynamic.

5.2 Services will Become Dynamic

Additionally, the concept of the *service* will become more dynamic than today. In order to optimize the network resources, CSP could sell services to their customers only when they need it. One example of such concept is the connectivity between branch offices: today the business users plan a specific capacity that is design to absorb the possible traffic peaks but probably they do not always utilize such capacity although they pay for it. With NFV, a CSP can provide a Web Portal to their business users where they can order on the fly more capacity for specific needs and only for a specific amount of time. Pushing that concept, the CSP can allow to activate a service (e.g. videoconference) only when needed. Or a business user can decide to enable an Hosted PABX or modifying the configuration of an existing one to include new branch offices.

All the examples below are just a taste of the dynamicity NFV can introduce in regards to service and network capacity. By the way, such use cases are already real scenarios for what the industry defined the Virtual CPE (Customer Premises Equipment).

5.3 Service Assurance and Service Provisioning

Although very appealing from CSP's revenue standpoint, the *network* and *service elasticity* are a nightmare for Service Assurance: to simplify, the Network Operation teams need to monitor something that can change at any time and without any manual intervention.

The only solution is that Service Assurance must become part of the so-called Service Fulfillment process. Problem is that today S.A. is part of OSS applications while separate applications, falling under the BSS category, manage everything that deals with the service ordering and provisioning. Service Orchestrators and VNFMs will play a key role in such integration but again, no standards are fully available yet.

5.4 Root-Cause Identification

Another big problem to address is the root-cause identification of the problems: such activity is already challenging in the current physical network due to the high number of components that interact to provide services, but with introduction of NFV, it will become worse.

Try to imagine that your OSS application detects a problem in the quality of a service that can affect many users in a dense area. Ok, now you are aware that there is a problem but what about its cause ? It can be in the virtual-to-physical interworking, into a VNF, or in the interaction of the VNF with the physical hardware (example: overload situation of a physical resources that is affecting all the VNF running on it).

5.5 Real-Time Analytics

A further new role of S.A. is in providing real-time feedbacks to the Service (or Network) Orchestrators about the QoS and QoE: such information, obtained monitoring the live traffic and the activity of each users, is very important to let the Orchestrator to take real-time decisions about resources to allocate.

As example, the S.A. tool, in this case probably a Real-time Analytics application, can forward real-time information about degrading QoE: the Orchestrator, based on pre-configured rules, can automatically *scale-in* more capacity to manage the increased traffic.

It is easy to understand how all the topics covered above implies a complete revolution in the OSS / BSS interaction from both technical and organizational standpoint. If CSP does not carefully plan such aspects within the projects for NFV, risk is that it will deploy a network impossible to manage (or that the S.A. will be not capable to stay synchronized with the network topology and the services to monitor).

6 Conclusions and Takeaways

This paper has provided some explanations about the importance of the adoption of the NFV concepts from CSP, an overview of the technology, a set of guidelines to drive the

CSP in its evaluation process and, finally, the role and importance of Service Assurance as part of the overall project.

Summarizing, NFV looks very promising but if CSP wants fully leveraging such investment Service Assurance must have the same importance of the components providing the service and not, as very often happen nowadays, an afterthought.