EAI Endorsed Transactions
on Future Internet
Research Article ICST.ORG

Network Function Virtualization: A Primer

Akshay Gadre and Krishna Sivalingam[1,*]

[1]Department of Computer Science and Engineering, Indian Institute of Technology Madras, India

Abstract

Network function virtualization (NFV) and **Software Defined Networking** (SDN) are widely acknowledged as potential disruptive technologies of next generation computer networks. The network operators are pushing the frontiers of NFV to reduce operational expenditure (OPEX) and capital expenditure (CAPEX). Network vendors and academic researchers look upon NFV as an opportunity to speed up innovation in modern networks. NFV also opens the space for rapid innovation in network services by reducing the cost of deployment and maturation cycle duration. This article describes the origins of NFV, its current state-of-the-art architecture, development tools, use cases, and its relationship with SDN. This article also brings forth the challenges this field holds for the academia and industry.

Keywords: NFV, SDN, VNFs, ETSI ISG, OpenFlow

Received on 25 February, 2016; accepted on 14 May, 2016; published on 18 May, 2016

doi:10.4108/eai.18-5-2016.151253

1. Introduction

The concept of virtualizing network functions or **Network Function Virtualization** (NFV) has been gaining much attention in the telecommunications and networking industry [1]. NFV involves exploiting standard computing virtualization technology to consolidate many network equipment types onto industry standard high volume servers, switches and storage. These could be located at data centers, network nodes and in the end user premises. It enables a transformation in the way that operators architect and design networks to implement their networking infrastructure. In other words, NFV refers to the development of software-based implementations of hardware network functions and running them on a virtualized set of resources on carrier-grade servers to mimic the behavior of the corresponding hardware middle-boxes.

Some of the common network functions that can be realized based on NFV concepts include traditional functionality such as packet routing, firewalls, load balancers, security and so on. The traditional method of using customized middle-boxes is not very cost effective. It is also tedious to upgrade and consists of proprietary solutions. With NFV, many of these middle-boxes can be replaced by software entities running on commodity computing server hardware using virtualization.

We first discuss the benefits of NFV from the industry perspective. NFV has originated from use-case driven requirements of the industry. It aims to reduce the cost of operation of on-the-line network services and improve the capacity of operation using the power of scalable virtualization of hardware resources for different software implemented network functions. It speeds up innovation in networking services by lowering the maturity cycle duration and cost of deployment and speeding up infrastructure upgrade processes. It also enables them to scale various services to cater to different levels of traffic, in turn, opening up various research areas related to dynamic scalability and network element placement for optimal use of network infrastructural resources.

From the academic perspective, NFV introduces several new opportunities such as design of efficient NFV related algorithms, development of open source tools to model the elements of an NFV architecture, and hypervisor innovation for virtualization of the hardware resources. It also leads to the need for mathematical models for obtaining performance and reliability measures.

Another emerging next-generation network technology is that of Software Defined Networking (SDN) [2]. This aims to separate the data plane and control plane of routers and uses open standardized interfaces to

[*] Corresponding author. krishna.sivalingam@gmail.com

communicate between them. This helps in lowering the innovation barrier that existed due to the proliferation of proprietary hardware-software router and switch interfaces. The OpenFlow protocol is one such interface which originated from academia and is widely used as the standardized interface. SDN also decentralizes control by using a controller responsible for instantiating flows in OpenFlow switches as flow rules and dynamically changing them depending on the incoming traffic flows. Contrary to common belief, SDN is complementary to NFV and vice-versa. Both of them can be implemented individually or together, with each having its own specific use case and technology barrier.

In this article, we discuss the fundamental concepts of of NFV technology along with the motivation for its need. Then, we describe a standard architecture proposed by the ETSI standards body, followed by the current open source implementation status of the NFV standard. We then motivate the real world use cases of this technology and its relationship with SDN. We conclude the paper by discussing the prevailing challenges and emerging topics of research in this field.

2. Genesis and History of NFV

This section briefly describes how the NFV concepts came into existence and its development over the past few years.

2.1. Origin

The concept of NFV was first formally proposed in the white paper submitted by ETSI NFV - Introductory White Paper[1]. Some of the world's leading telecom network operators, including AT&T, BT, Deutsche Telekom, Orange, Telecom Italia, Telefonica and Verizon, convened with the common rationale of inducing accelerated development and deployment of network functions on high volume state-of-the-art industrial servers in November 2012. They then requested ETSI (European Telecommunications Standards Institute) to initiate an ISG (Industry Specification Group) for NFV.

This ISG was charged with the tasks of research and development of requirements and architecture for virtualizing functions within Telecom networks while maintaining interoperability and without affecting the user experience. This is required since all the networks have proprietary hardware devices. Launching new services directly corresponds to deployment of new hardware resources. This increases space and power consumption to accommodate new hardware. This leads to shorter life cycles for hardware-based devices, lower return of investments and reluctance towards adopting innovative ideas in network-centric solutions.

The first phase of the ISG concluded in October 2013 with the publication of 11 ETSI group specifications [3]. This included details regarding the infrastructure, management and orchestration (MANO), security, resilience, architecture, description of the domains of infrastructure (in terms of compute, hypervisor and network) and Quality-of-Service (QoS) metrics. The second phase is currently underway with more focus on adoption and reorganization of NFV working groups.

The current working and expert groups (WGs and EGs) include: Architecture for Virtualization Infrastructure (INF), Management and Orchestration (MANO), Software Architecture (SWA), Reliability, Availability, Fault Tolerance and Resilience (REL), Security (SEC), Evolution and Ecosystem (EVE), Interfaces and Architecture (IFA), and Performance and Portability (PER).

2.2. Motivation

The main motivation behind NFV is that it reduces the requirement of dedicated specialized hardware resources for the deployment of network functions. It offloads these functions onto software running on carrier-grade generic hardware. This is managed from anywhere in the network. This is beneficial for the network operators by catalyzing innovation, testing and deployment of new services without waiting for an upgrade.

Some of the main benefits for the network operators are:

- Reduced costs and more control over power consumption by using generic equipment and leveraging standard IT virtualization technologies.
- Reducing maturity cycle duration for innovation by significantly reducing the cost of deployment and enabling quick prototyping of innovative services.
- Dynamic scaling of resources across network appliances on a single platform to cater to various applications, users, and tenants.
- Enable introduction of services to specific customers by realizing a chain of virtualized network functions (VNF) or service chains.

It also benefits network industry entrants and academia by opening the virtual appliance market to them, inducing innovation to bring new services and, in turn, shortens the time to market.

2.3. Technical Challenges

To achieve the above motive, the following challenges must be addressed before expecting a smooth deployment and operation of NFV:

- Developing high performance virtual network appliances interoperable with existing network equipment.

- Achieving backward compatibility and providing efficient migration path to virtual network functions which depend on existing operations support systems (OSS)/ business support systems (BSS).
- Managing and Orchestrating (MANO) the virtual appliances while ensuring security of the network and by mitigating risks of manual misconfiguration.
- Guaranteeing the quality of service (QoS) and ensuring resilience to failures.

3. ETSI NFV Architecture

The ETSI NFV INF WG is responsible for describing the architecture for virtualization infrastructure [4]. The most recent architecture prescription is shown in Figure 1 and is described in the following subsections.

3.1. Virtual Network Function (VNF)

This is the most important component of the NFV architecture. This is the software-based implementation of the network function of a hardware-based middle-box. A single hardware middle-box is sometimes even decomposed into multiple VNFs to mimic its actions. Examples include Firewall, NAT traversal, routing services, load balancer, etc. Having a software-based implementation enables bundling of related network features (say video) on the same instance.

3.2. Element Management System (EMS)

This is a management system for the elements, i.e. the VNFs, responsible for various operations like fault and performance management. The interface between the VNF and EMS can be proprietary and can multiplex multiple VNFs or even act as VNFs themselves.

3.3. VNF Manager

The VNF manager is responsible for life cycle management of all VNF instances including setting up, maintaining and tearing down VNFs. This is different from an EMS since it acts through an open interface in the architecture and is fundamental to rapid innovation in network functions.

3.4. Network Function Virtualization Infrastructure (NFVI)

NFVI is the scalable computational environment which is responsible for hosting the VNFs along with EMSs on it. It is responsible for exposing hardware resources as virtualized environments and gather statistics about fault and performance of the carrier-grade servers on which these services are running.

3.4.1. Hardware Resources. These are the actual computational, memory and networking resources available at the server to be exposed via the virtualization layer. Depending on the usage and requirement of the overlaying VNFs, these resources are dynamically allocated to the scale of use for an VNF. These are the maximum possible resources that can be allocated in a server.

3.4.2. Virtualized Resources. These are the abstracted resources from the hardware ones through the virtualization layer. These are made available to the VNFs to be used for their working.

3.4.3. Virtualization Layer. This layer is called the hypervisor in generic IT virtualization paradigm and is responsible for decoupling the software resources from hardware resources. This is an abstraction layer which only exposes the required hardware resources dynamically to the software implemented VNFs.

3.5. Virtualized Infrastructure Manager (VIM)

This is the component of MANO responsible for NFVI. It controls and manages the resources and infrastructure available for all servers of the network operator. It collects all performance metrics and anomalous events and uses them for future decision making.

3.6. NFV Orchestrator

This is a crucial element of the MANO part of the NFVI architecture. It chains multiple VNFs together to enable creation of an end-to-end service. It is also responsible for management of resources across various VIMs in the network. It uses VNF Manager and VIM to manage the resources.

3.7. OSS/BSS

This is the operator specific component of the architecture. OSS includes fault management, configuration, services and networking while BSS revolves around customer billing and service/product portfolio management. It interacts with the network to perform the above operations and to receive feedback.

4. Implementation status

NFV technology implementation is still in its infancy with many new technologies and optimizations being proposed and integrated day-by-day. In this section, we will briefly describe two front-runners in accelerating introduction of NFV products and services. One is an open-source platform, called *opNFV* [5], focused on building the VIM and NFVI using current state-of-the-art projects. The other is a testbed project called *Pharos* [8] that aims to develop a cloud-based testing infrastructure to test and deploy NFV technologies.

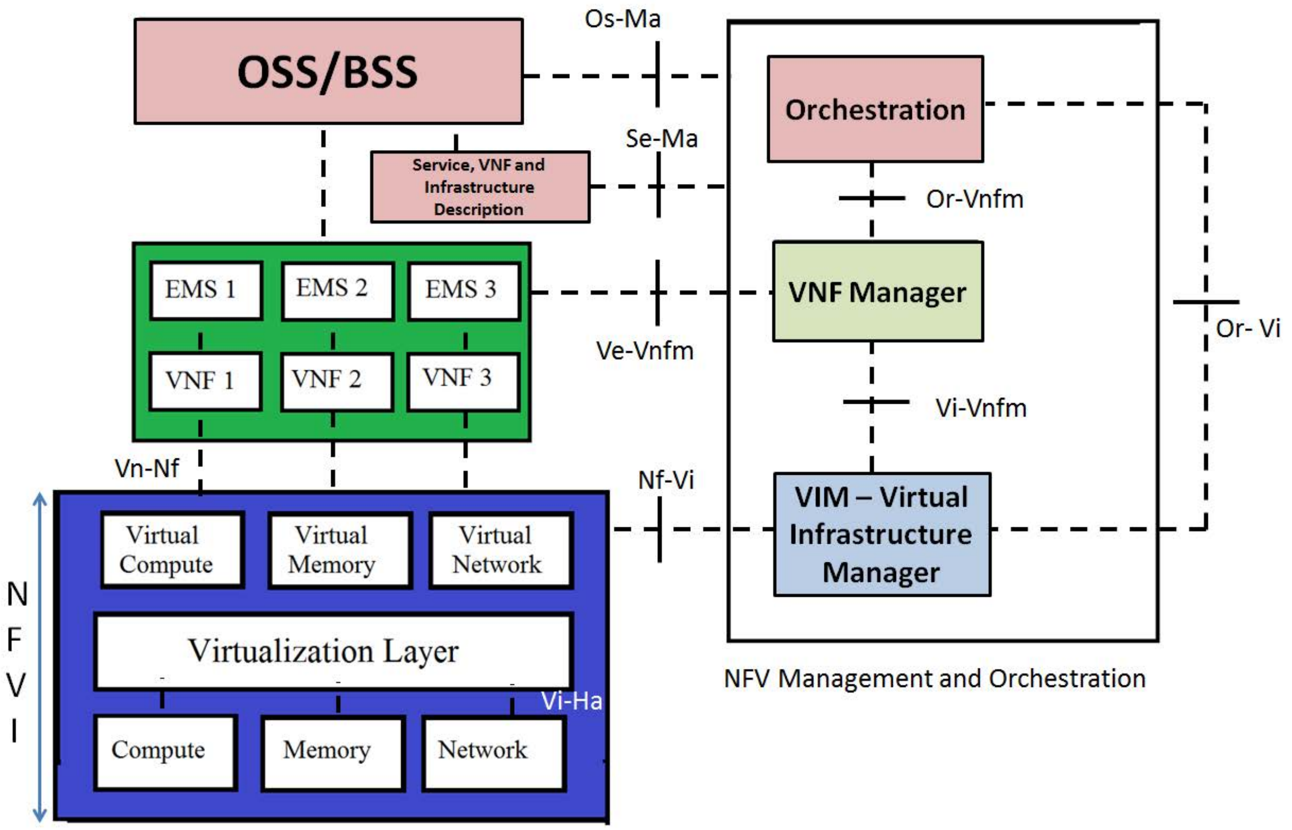

Figure 1. ETSI ISG NFV Architecture

4.1. opNFV

The *opNFV* system, an open-source, integrated platform, has been developed in order to deliver a state-of-the-art standard open-source NFV platform for catalyzing research and development in NFV. It plans on using the current virtualization, SDN, kernel and cloud infrastructure projects like OpenStack, OpenDaylight, Open vSwitch, Linux, and KVM to create the NFVI and VIM part of the ETSI ISG NFV Architecture.

The first release (Arno) happened in June 2015[6] and the most recent release is the Brahmaputra which was released in March 2016[7]. Brahmaputra boasts of a larger number of testing scenarios with more SDN controllers, installers, deployment options, and carrier-grade features. In the NFVI, it supports KVM for compute virtualization, Ceph for storage virtualization, and OpenDaylight, ONOS, OVS and Open Contrail for network virtualization with OpenStack on top of it. It supports the OPNFV Bare Metal Lab and Pharos Community Labs [8] as its cloud infrastructure beneath the virtualization layer.

The project then plans to develop standardised application programmable interfaces (APIs) for interaction with the above to form the basic infrastructure of VNFs and MANO components. The long term objectives of the project are to provide efficiency, reliability, availability, and serviceability, aiming to become the preferred platform for open source NFV.

In addition to the above, it includes fault management, support for IPv6 and L3VPN and service function chaining features. Using the above infrastructure, detailed simulation of many state-of-the-art use case scenarios of NFV like life-cycle management of VNFs, VNF forwarding graphs, fault detection and recovery, traffic generation, and abstracting NFVI as a service (NFVIaaS) can be carried out.

4.2. Pharos Project

Pharos is a opNFV test-bed project aiming to develop a collaborative virtualization environment to provide NFVIaaS by creating a testing infrastructure of community labs world-wide. It is developed along the lines of Global Environment for Network Innovations (GENI)[9] project, which allows access to slices of laboratory infrastructure worldwide across universities and cloud platforms for networking and distributed systems research as per demand.

Pharos aims to provide bare-metal compute resources for development, deployment and testing while optimizing resource usage and providing secure access. It

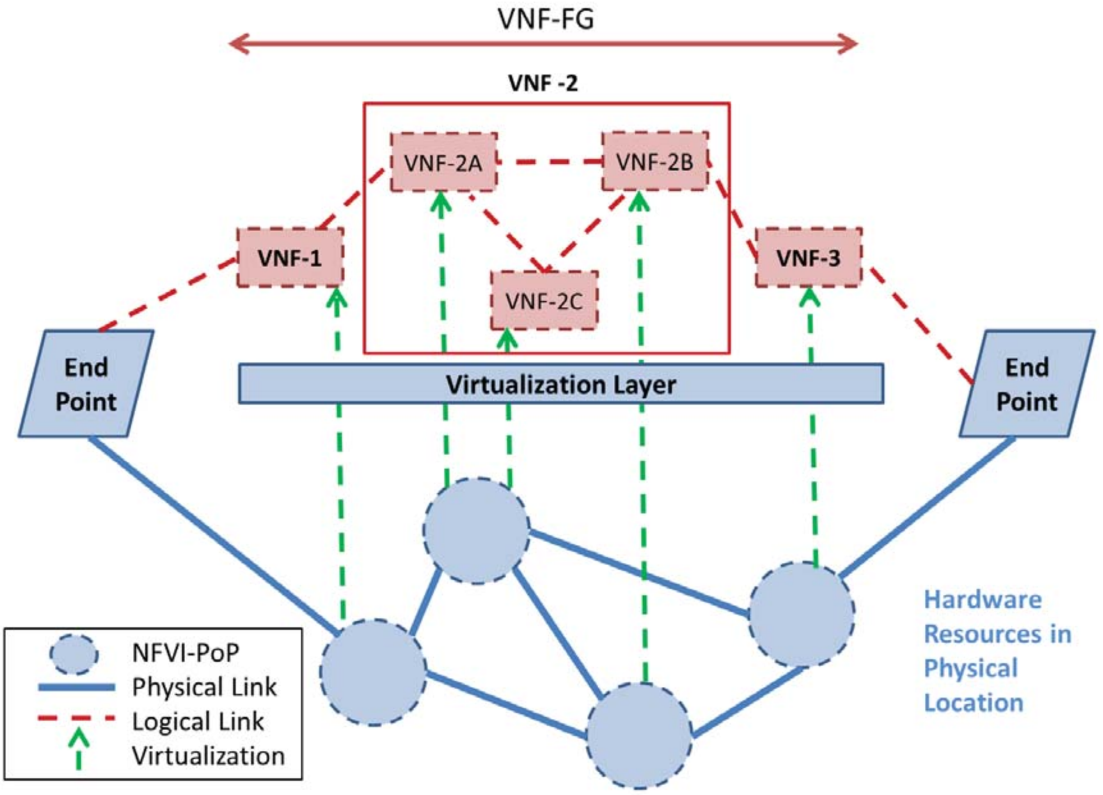

Figure 2. VNF Forwarding Graph.

emphasizes on interoperability across versions, collaborative testing across scenarios and provides a realistic and a reliable test environment.

5. NFV use cases

As mentioned above, the concept of NFV originated to cater to the various use cases[10] proposed by the network operators for the use of this technology to reduce resource cost and speed up the maturity period of innovation in various network components. In this section, we look at some of the use cases along with the motivation behind them.

5.1. VNF Forwarding Graphs

A VNF Forwarding Graph (VNF FG) is a sequential set of VNFs that a packet traverses while inside a NFV system. In an NFV scenario, VNF FG is responsible for mimicking the response of a middle-box by forwarding a packet through various VNFs.

To realize the VNF FG (shown in Figure 2), we need to develop network services to identify the VNFs involved and the interconnection topology amongst them. Note that some of the functions need not even be virtualized and can interoperate with hardware-based middle-boxes in real world scenarios.

The target is to develop an information model that allows the Network Service Providers to process the packets using multiple VNF chains working in parallel.

One of the possible solutions is that we use the NFV architecture's network resources, points of presence (POPs), to forward the packets from the respective physical switches at which the VNF forwarding chains start and traverse the VNF chains modifying the packet as required and then forward it back to the physical switch after the traversal for further forwarding in the data flow path.

5.2. Virtualization of Mobile Core Network and IP Multimedia Subsystem(IMS)

NFV architecture's consolidation of hardware is expected to reduce the total cost of ownership (TCO). Flexible and optimal allocation of network functions on hardware resources pool is pivotal to serving increased demand for service with the same infrastructure without relying on call restriction control mechanisms. It will also be important to provide the option of scaling (both up and down) resources for a VNF as required and this is provided by the VIM and Virtualization layer.

The IMS is a session control architecture to provide end-to-end multimedia services. A VNF will be able to scale by requesting extra resources from the virtualization layer. We can thus choose to actually virtualize the whole Mobile Core Network or progressively virtualize one layer at a time based on the requirements.

Important problems faced are that of resource scaling, inter-operability and transparency across layers in the virtualization hierarchy. MANO is critical to the control of the scalability problem while the other problems relate to development of interfaces for layers to interact with each other based on the application requirements.

5.3. Network Function Virtualization Infrastructure as a Service(NFVIaaS)

Many service providers offer cloud computing services with virtualized access to their physical compute, network and storage resources. Many of these services are offered as Infrastructure as a Service (IaaS) where users can run customised applications using their resources.

These resources can be considered as the NFVI resources in NFV domain with each element able to host VNFs on top of it. We will also need to provide specific interfaces for the MANO of the NFV architecture to interact with the NFVI.

Some of the key problems to be resolved in this are the atomicity of each VNF when multiple tenants share the resources, dynamic orchestration of VNF FG depending on the resources allocated, and accurate monitoring and management of allocated resources.

5.4. Virtualization of the Content Delivery Networks(CDNs)

Content Delivery Networks (CDNs) are integrated into operators' networks to deliver video and audio services. Integration of these CDN nodes into the network nodes (by way of caching) is pivotal for an effective and cost-efficient way to tackle the growth and quality of video delivery. Doing so will conserve network link resources, confine the data streams nearer to end customers, and provide better bandwidth and reliability.

Having virtualized CDNs provide multiple benefits allowing the resources to be diverted to other uses during low usage and getting high resources during peak hours of service. Same resources can be shared with multiple CDNs thus reducing the need of infrastructure for each of the CDN. The CDNs can use NFVIaaS to add their service on a carrier-grade server which will then take care of its distribution.

This also will allow CDNs to upgrade their software and provide innovative services at a much faster rate with lower costs and thus will be beneficial for them. However, a fully-functional CDN software is required for this to work and interoperability with hardware-based CDN nodes where needed will also be required.

6. Relationship to SDN

As explained above, SDN is a paradigm of networking which relies on separation of data and control plane in routers pushing against the frontiers of open-source interfaces and using centralised controllers for directing the data flow using standardised interfaces like OpenFlow.

SDN originated from aversion to proprietary control over the data flow and control flow inside a switch in any state-of-the-art router, and sought a solution which will break this transparency barrier to break monopoly in innovation in these planes by developing standardised interfaces and OpenFlow switches which follow this interface to interact with OpenFlow controllers.

The aim of both ideas is complementary and is best summarised by Figure 3 which shows that each has different objectives. NFV's objectives align with that of SDN's by hosting network functions on commodity servers instead of proprietary hardware. Using SDN also simplifies things for implementation of NFV by providing a standardised north-bound interface for redirecting flow to its VNFs.

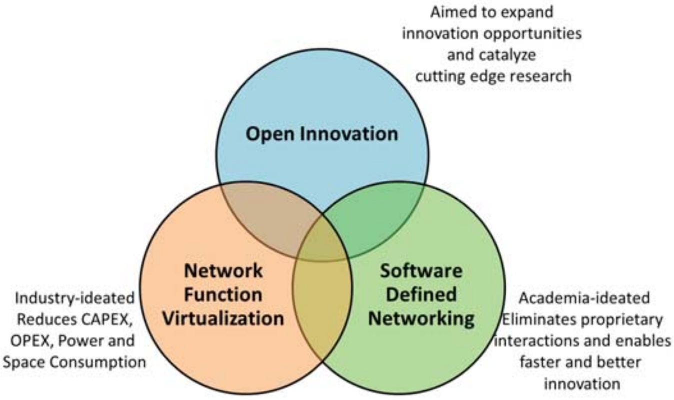
Aimed to expand innovation opportunities and catalyze cutting edge research
Open Innovation
Network Function Virtualization
Software Defined Networking
Industry-ideated Reduces CAPEX, OPEX, Power and Space Consumption
Academia-ideated Eliminates proprietary interactions and enables faster and better innovation

Figure 3. SDN and NFV.

We discuss the following example to justify the complementary existence of NFV and SDN.

6.1. Using NFV and SDN to provide router service

A virtual router provides access to multiple customers simultaneously. It also isolates their networks. To realize this it uses an L2 Network Interface Device (NID). This is typically implemented on an aggregation router at a central office. Customers requesting this service are given a slice in the virtual router. They can processes their traffic in a specific way without impacting other customers' traffic as shown in Figure 4.

The routing function is implemented as a NFV and is run on top of a VM at the NFV PoP. This NFV, along with its MANO, will route the packets destined to the customer. Here, the need of internal routing within the central office is removed and the NID is required only at the customer level as evident in Figure 5. Thus, the routing function has been virtualized as a VNF.

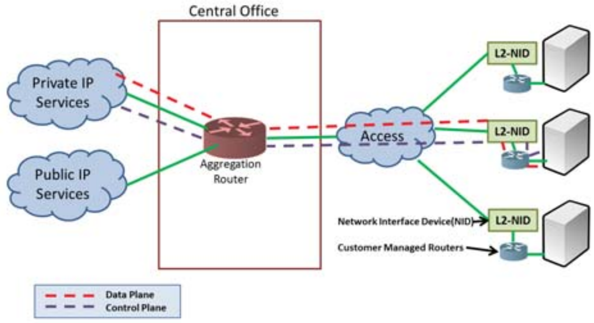

Figure 4. Managed Router Service Infrastructure

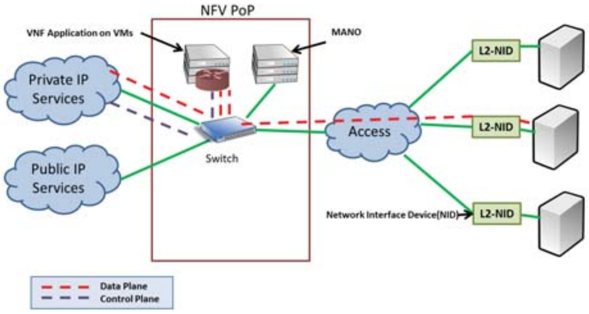

Figure 5. Managed Router Service with NFV

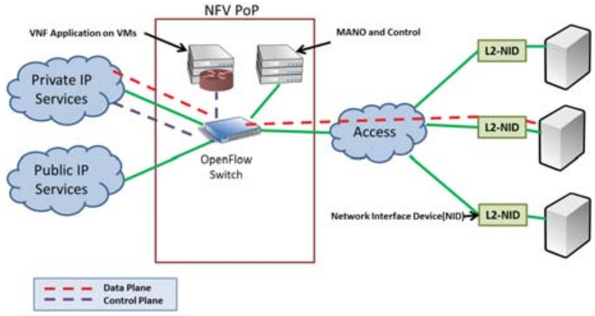

Figure 6. Managed Router Service with NFV and SDN

As we have already discussed, SDN deals with separation of the control and data plane. Thus, the data packets need not be sent to the router's distributed forwarding function but instead can be forwarded through an optimized path configured by the control plane running as a VNF on a server. Figure 6 shows the data plane and control plane for this case.

As we have shown above, the objectives of SDN and NFV are complementary instead of contradictory and both are critical for supporting innovation in modern networks.

7. Conclusions

This paper has described the emerging technology of Network Functions Virtualization, with its state-of-the-art architecture, history, requirements, use cases and its relationship with other emerging networking technologies like SDN. As mentioned earlier, the infancy of this technology makes it a rich open area for research and innovation. The recent white paper [11] by ETSI ISG lists the fundamental challenges this field still faces for overall and long-term success.

The first class of challenges are related to algorithms. They include service chaining algorithms, NFV orchestration algorithms, energy-efficient NFV architectures and parallelization using compositional patterns. These problems are associated with optimizing the usage time and infrastructure requirement of the NFV architecture by using more optimal algorithms to carry out the tasks.

The next class of challenges are related to interactions. They include the abstractions for networks and services, serviceability and reliability of network services and service fault management. These problems deal with specific use-cases of the NFV architecture, or to guarantee reliable network services to the consumers. Their solutions involve change in architecture, proof of concepts and research in the MANO part of the NFV architecture.

The last class of challenges are related to evaluation and conformance. They include performance studies, simulation platforms, backward and forward compatibility. These challenges are for verification and maturation of NFV as a emerging technology, providing field study and usage of the technology in real world and virtual scenarios. These invoke the need to develop new tools and technologies to simulate NFV in each use case and be in sync with the rapidly growing networking technologies.

Of course, there are many other challenges which are fundamental to networks, which apply also for NFV. These include developing optimal topologies and high-performance architectures and reducing the resource footprint for every use case.

Thus, NFV is seen to be one of the foremost emerging networking technologies with backing from the industry and a fertile ground of research for academia. NFV, along with SDN and other upcoming technologies, seems fundamental to the development of advanced next-generation networks.

Acknowledgments. The authors are pleased to acknowledge the help of Mr. C. S. Ganesh and Mr. Anix Anbiah, Ph.D. students in the CSE Department of IIT Madras, for their assistance in reviewing this manuscript.

References

[1] ETSI White Paper, Network Functions Virtualization: An Introduction, Benefits, Enablers, Challenges & Call for Action. https://portal.etsi.org/nfv/nfv_white_paper.pdf, October 2012.

[2] N. Feamster, Jennifer Rexford and Ellen Zegura, The Road to SDN: An Intellectual History of Programmable Networks, ACM SIGCOMM Computer Communication Review, Vol. 44, Issue 2, pp. 87–98, April 2014.

[3] ETSI Press Release for end of Phase 1, http://www.etsi.org/news-events/news/, January 2015.

[4] ETSI GS NFV-INF 001 Network Functions Virtualisation (NFV): Infrastructure Overview, January 2015.

[5] opNFV, https://www.opnfv.org/, May 2016.

[6] opNFV: Release Arno, https://www.opnfv.org/arno, June 2015.

[7] opNFV: Release Brahmaputra, https://www.opnfv.org/brahmaputra, March 2016.

[8] Pharos Project: The OPNFV Community Test Labs, https://www.opnfv.org/developers/pharos, May 2016.

[9] Global Environment for Network Innovations(GENI), https://www.geni.net/, May 2016.

[10] ETSI GS NFV-INF 009 Network Functions Virtualization (NFV); Use Cases - ETSI, October 2013.

[11] ETSI ISG Network Functions Virtualization - White Paper #3, https://portal.etsi.org/Portals/0/TBpages/NFV/Docs/NFV_White_Paper3.pdf, October 2014.

Author Biography. **Akshay Gadre** is pursuing his B. Tech. and M. Tech. degrees in Computer Science and Engineering at IIT Madras. His research interests include computer networks.

Krishna Sivalingam is a Professor in the CSE Department, IIT Madras, Chennai, INDIA. He received his Ph.D. and M.S. in Computer Science from SUNY Buffalo; and his B.E. degree from Anna University's College of Engineering Guindy, India. His research interests include wireless and optical networks. He is an IEEE Fellow, INAE Fellow and ACM Distinguished Scientist. He is presently serving as Editor-in-Chief of Springer Photonic Network Communications Journal and EAI Transactions on Future Internet.