NETWORK FUNCTIONS VIRTUALIZATION (NFV) with A TOUCH OF SDN

RAJENDRA CHAYAPATHI | SYED FARRUKH HASSAN | PARESH SHAH

Network Functions Virtualization (NFV) with a Touch of SDN

This page intentionally left blank

Network Functions Virtualization (NFV) with a Touch of SDN

Rajendra Chayapathi CCIE® No. 4991,
Syed Farrukh Hassan CCIE® No. 21617,
Paresh Shah

Addison-Wesley

Boston • Columbus • Indianapolis • New York • San Francisco
Amsterdam • Cape Town • Dubai • London • Madrid • Milan
Munich • Paris • Montreal • Toronto • Delhi • Mexico City
São Paulo • Sidney • Hong Kong • Seoul • Singapore • Taipei • Tokyo

Visit us on the Web: informit.com/aw

Library of Congress Control Number: 2016952060

ISBN-13: 978-0-13-446305-6
ISBN-10: 0-13-446305-6

First printing: November 2016

Editor-in-Chief
Mark Taub

Product Line Manager
Brett Bartow

Development Editor
Marianne Bartow

Managing Editor
Sandra Schroeder

Project Editor
Mandie Frank

Copy Editor
Warren Hapke

Indexer
Cheryl Lenser

Proofreader
Sasirekha

Technical Reviewers
Mark Cohen
Krishna Doddapaneni

Editorial Assistant
Vanessa Evans

Designer
Chuti Prasertsith

Compositor
codeMantra

I would like to dedicate this book to all the curious minds out there whose interest in NFV is the inspiration for this book. I would also like to dedicate this book to my ever supportive and loving family, for without their encouragement, support, and help, this endeavor wouldn't have been possible. Thanks to all who have supported me to write this book.

– Rajendra Chayapathi

I would like to dedicate this book to my wife and children for their love, patience, and especially for their strong support to me while writing this book. Thanks to them for letting me sacrifice many weekends and weeknights of family time. I would also like to thank my parents for their continued support, encouragement, and for guiding me through their wisdom.

– Syed F. Hassan

I would like to dedicate this book to my family, friends, and parents. I especially thank my wife, who filled in for me by taking care of the kids and keeping them busy to not slow me down. I also give a big thank you to my two adorable daughters, who excused me from spending time with them. Lastly I would like to acknowledge the support from my parents for motivating me to think different and share the knowledge with others, which translated to writing this book. Without their support, this book wouldn't have been possible.

– Paresh Shah

And most importantly, all of us would like to thank God for all His blessings in our lives!

This page intentionally left blank

Contents

This page intentionally left blank

Preface

Register your copy of *Network Functions Virtualization (NFV) with a Touch of SDN* at informit.com for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to informit.com/register and log in or create an account. Enter the product ISBN 9780134463056 and click Submit. Once the process is complete, you will find any available bonus content under "Registered Products."

This page intentionally left blank

Acknowledgments

We would like to say a very special thanks to the technical reviewers, Nicolas Fevrier and Alexander Orel. They took up the challenge of reviewing and correcting the technical inaccuracies and shared their expert opinions by providing us with helpful recommendations. Their expertise, suggestions, and guidance helped us to navigate presenting the content in the right direction and keep it at an appropriate depth.

We would like to thank Brett and Marianne Bartow at Addison-Wesley Professional for bearing with us throughout the process of putting this book together and guiding us through each step. Finally, we would like to thank our colleagues and peers for giving us the inspiration and encouragement to share our knowledge through this book.

This page intentionally left blank

About the Authors

Rajendra Chayapathi is a Senior Solution Architect in Cisco's professional and consulting services organization. His most recent work has been on emerging technologies such as NFV, SDN, programmability and network orchestration and its adoption in the industry. He has over twenty years of experience in networking technologies, customer interaction, and networking products; his focus is on network design and architecture. He has previously worked in Cisco's engineering teams where he was involved on various network operating systems and product development. Before his employment at Cisco, he provided consultancy services to AT&T and financial institutions for the design and deployment of IP core network technologies. He has been a regular speaker at multiple technology conferences such as Cisco Live, Cisco Connect and NANOG. Rajendra has a CCIE (#4991) in Routing and Switching and also holds a Bachelor's degree in Electronics and Communication from University of Mysore, India and a Masters' degree in Business Administration with a focus on technology Management from University of Phoenix, USA.

Syed Farrukh Hassan has been in the networking industry for fifteen years, and is currently a Senior Solutions Architect in Cisco's professional and consulting services organization. He has worked with various Internet and cloud service providers, helping them in adoption of innovative network technologies and supporting them in design and deployment of new architectures. In his current role, Syed is involved in SDN and NFV adoption, providing guidance, future strategy, and planning to service provider, enterprise, and data center customers. Syed has previously been part of engineering teams within Cisco and has been an active contributor towards design and innovation of network products and solutions. Syed has been a regular speaker in public forums and conferences and is recognized as a Cisco Live Distinguished Speaker. Syed is a double CCIE in Service Provider and Data Center technologies (#21617) and also a VMware Certified Network Virtualization Professional (VCP-NV). He holds a Bachelors' degree in Engineering from NED University, Pakistan, and a Masters' degree in Engineering from the University of Florida, Gainesville, USA.

Paresh Shah has been in the network industry for more than twenty years and currently working as a Director in Cisco's professional and consulting services organization. He is responsible for bringing to market new disruptive services based

on cutting-edge technologies and solutions to achieve successful deployment in customer networks. Paresh has led various global engineering and customer-facing groups in the service provider market and is a veteran of the high-end routing, service provider, enterprise, and cloud segments. He started his career as an engineer in 1996 building one of the first high-speed multi-services routers in the industry and was responsible for adoption of new technologies then, like MPLS, BGP, and L2/L3 VPN and new operating systems like IOS-XR. Paresh is leading the adoption of NFV, SDN, and segment routing consultancy services and driving the solutions for cloud-providers, traditional service providers, and enterprises that are looking to adopt these new technologies. Paresh is a regular speaker at industry conferences such as Cisco Live, NANOG, and SANOG, with a pulse on the latest trends in the industry. He has a Bachelors' degree in Electrical Engineering from University of Pune, India and a Masters' degree focusing in Networking and Telecommunications from University of Missouri-Kansas City, USA.

About the Technical Reviewers

Nicolas Fevrier is a Technical Leader in Cisco's Service Provider team. He is a networking veteran and during his twelve years at Cisco he has taken up positions for technology validation, consultancy services, and most recently Technical Marketing. He has traveled around the world to deploy, support and promote various IOS XR routing platforms. Currently Nicolas focus is towards driving and facilitating the adoption of the cutting-edge technologies focused on IOS XR products. He is also heavily engaged in providing guidance on network services such as services for securing network, mitigation of distributed Denial of Service, and services for network transformation using Carrier Grate NAT. Nicolas has strong interest in NFV and SDN areas, and has been part of technical marketing team for Cisco's service provider business. He is a regular speaker at technical conferences and a distinguished Cisco Live speaker. Nicolas holds a CCIE in Routing and Switching (#8966).

Alexander Orel has more than fifteen years of experience in networking field. He has worked in multi-vendor environments for various Internet service providers and network consulting companies. Currently Alexander works as a Solutions Architect in Cisco Professional and consulting services team, where he works with global service providers and enterprises, ratifying their requirements, and helping them plan and supporting the deployment of Next-Generation network products and technologies. His specialization is IOS XR-based platforms and NFV technologies. Alexander has a Master's degree in applied physics from Moscow Institute of Physics and Technology and currently holds CCIE certification #10391 in R&S and DC. Alexander has been a frequent presenter at various technology conferences such as Cisco Live and Cisco Connect. Alexander lives and works in Ottawa, Canada.

This page intentionally left blank

Introduction

Network functions virtualization (NFV) is significantly influencing the world of networking and changing how networks are designed, deployed, and managed.

NFV gives network service providers freedom of choice and allows separating the networking software from the hardware. This decoupling brings advantages such as cost savings in deploying and operating the network, rapid on-demand provisioning of new network functions, increased efficiency, and agile network scalability. These advantages open the door for new business opportunities, bring new services to market quicker and have been attracting tremendous interest from cloud and Internet service providers, mobile operators, and enterprise market segments.

Who Should Read This Book?

The book is targeted towards network engineers, architects, planners, and operators with any level of experience in networking technologies who are ready to enter the world of network functions virtualization. It assumes basic networking knowledge but is meant to be an entry-level book when it comes to understanding NFV architecture, deployment, management, and associated technologies.

Goals and Methods—How This Book Is Organized

It is critical to understand NFV (like any other disruptive technology) to maximize the benefits that it offers as well as to use it effectively and efficiently. This understanding of NFV requires learning new concepts and technologies and involves a learning curve for the engineers, architects, planners, designers, operators, and managers of today's networks. The motivation to write this book comes from the desire to facilitate learning about NFV technologies.

The goal of the book is to enable the reader to get a firm grasp on the NFV technologies and its building blocks. With the adaption of NFV, the roles in the networking industry will evolve significantly. This book gets readers ready to enter the NFV era, arming them with the knowledge to design, deploy, monetize, and make informed decisions about adopting NFV solutions in their networks.

The book takes the approach of building the concepts bottom-up, starting with the basic NFV concepts and discussing the advantages and design principles in depth, based on its applications. It gets the reader familiar with NFV orchestration, management, and use cases, then follow this with a discussion on the related technology of software-defined networking (SDN). It finishes with a discussion of the advanced NFV topics that glue everything together to complete the NFV canvas. The discussion is split into six chapters, each with its own goals.

Chapter 1: The Journey to Network Functions Virtualization (NFV) Era

The goal of this chapter is to understand the benefits of NFV and the market drivers that are enabling its adaption. The chapter starts the journey towards NFV by analyzing the network evolution over the past decades. This chapter also focuses on building the foundation knowledge of NFV by introducing the architectural framework and its components.

Chapter 2: Virtualization Concepts

This chapter focuses on the key technology that makes NFV possible—virtualization. The goal of this chapter is to get the reader very well acquainted with virtualization technologies and how they relate to NFV.

Chapter 3: Virtualization of Network Functions

This chapter takes a closer look at the design and deployment considerations for an NFV based network. The chapters also discuss the technical challenges that are expected when transforming today's networks to adapt NFV. The chapter closes with a discussion of network functions and services that are adapting or can adapt NFV.

By the end of the first three chapter, the reader should be familiar with planning an NFV deployment, foreseeing the challenges and design issues that will need to be considered, and evaluating the advantages that this transformation will bring and how those advantages can be maximized.

Chapter 4: NFV Deployment in the Cloud

With the foundations and design challenges already laid out and discussed in the previous chapters, this chapter takes those concepts and applies them towards orchestrating, building, and deploying NFV networks and services. The chapter also visits the management and orchestration solutions available, both through vendors and the open source community.

By the end of this chapter, the reader should have a through understanding of the tools and techniques that can be used to deploy and manage an NFV network.

Chapter 5: Software Defined Networking (SDN)

This chapter shifts to new topic and touches upon the concepts of SDN. The chapter covers the fundamentals of SDN and describes its correlation with NFV.

Chapter 6: Stitching It All Together

This chapter consolidates the knowledge gained from the previous chapters. Important considerations in an NFV network, such as security, programmability, performance, and function chaining, are discussed in this chapter. It also gives insight into the evolving NFV concepts that will shape the future of this technology.

This page intentionally left blank

Chapter 1

The Journey to Network Functions Virtualization (NFV) Era

Network functions virtualization (NFV) is a fast-emerging technology area that is heavily influencing the world of networking. It is changing the way networks are designed, deployed, and managed, transforming the networking industry towards a virtualization approach and moving away from customized hardware with prepackaged software.

This chapter walks you through the NFV journey and the market drivers behind it. It allows you to get acquainted with the concepts of NFV and examines the ongoing efforts towards standardization. It lays the foundation which is instrumental in understanding networking industry transition to NFV. It explains how the industry is evolving from a hardware centric approach to a virtualized and software—based network approach in the effort to meet the need and feed of cloud-based services which demand open, scalable, elastic and agile networks.

The main topics covered in this chapter are:

- Evolution from traditional network architecture to NFV
- NFV standardization efforts and an overview of the NFV architectural framework
- Benefits and market drivers behind NFV

The Evolution of Network Architecture

To appreciate the motivation and need behind the networking industry's fast adoption of NFV, it's helpful to take a look at the history of networking and the challenges that it faces today. Data communication networks and devices have evolved

and improved over time. But while networks have become faster and more resilient with higher capacity, they still struggle to cope with the demands of the changing market. The networking industry is being driven by a new set of requirements and challenges brought forward by cloud-based services such as infrastructure to support those services and demands to make them work more efficient. Mega-scale data centers hosting computing and storage, a factorial increase in data-enabled devices, and Internet of Things (IoT) applications are just some of examples of areas that need to be addressed for improved throughput and latency in existing networks.

This section examines traditional networks and networking devices and identifies the reasons they have been unable to cope with the new types of demands. It also takes a look at the way NFV brings a fresh perspective and different solution to these market-driven needs.

Traditional Network Architecture

The traditional phone network and perhaps even telegram networks are examples of the earliest data transport networks. Early on, the design criteria and quality benchmark by which networks were judged were latency, availability, throughput, and the capacity to carry data with minimal loss.

These factors directly influenced the development and requirements for the hardware and equipment to transport the data (text and voice, in this case). Additionally, hardware systems were built for very specific use cases and targeted functions, ran tightly coupled proprietary operating systems on them, and were meant to perform only specific functions. With the advent of data transport networks, the requirements and factors that influence the network's design and the devices' efficiency stayed unchanged (for example, the network design should achieve highest throughput with minimum latency and jitter over extended distances with minimal loss).

All the traditional networking devices were made for specific functions, and the data networks built were tailored and customized to meet these efficiency criteria effectively. The software or code running on these custom-designed hardware systems was tightly coupled to it, closely integrated with the silicon Field Programmable and Customized Integrated Circuits and focused exclusively on performing the specific functions of the device.

Figure 1-1 illustrates some of the characteristics of traditional network devices deployed today.

With the exponential increase in bandwidth demand, heavily driven by video, mobile, and IoT applications, service providers are constantly looking for ways to expand and scale their network services, preferably without significant increase in costs. The characteristics of traditional devices present a bottleneck to this requirement and create many constraints that limit the scalability, deployment costs, and

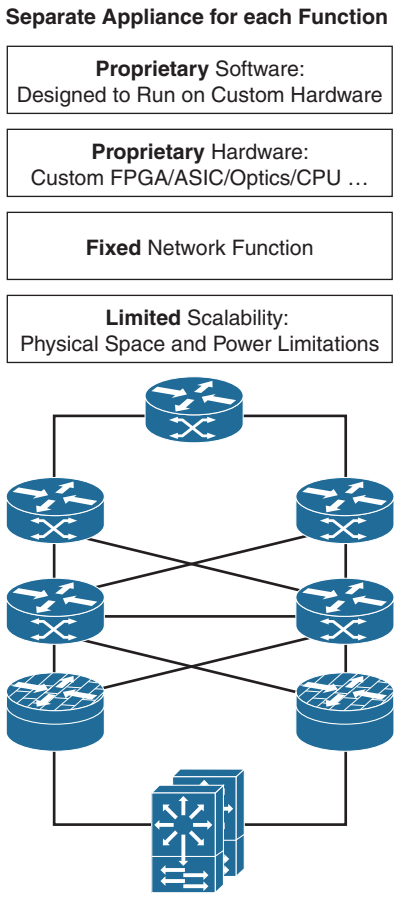

Figure 1-1 *Traditional Network Devices*

operational efficiency of the network. This situation forces the operators to consider alternatives that can remove the limitations. Let's examine some of these limitations.

Flexibility Limitations

Vendors design and develop their equipment with a generic set of requirements and offer the functionality as a combination of specific hardware and software. The hardware and software are packaged as a unit and limited to the vendor's implementation. This restricts the choices of feature combinations and hardware capabilities that can be deployed. The lack of flexibility and customization to meet fast-changing requirements results in inefficient use of resources.

Scalability Constraints

Physical network devices have scalability limitations in both hardware and software. The hardware requires power and space, which can become a constraint in densely populated areas. The lack of these resources may limit the hardware that can be deployed. On the software side, these traditional devices may not be able to keep up with the scale of changes in the data network, such as number of routes or labels. Each device is designed to handle a limited multi-dimensional scale, and once that ceiling is hit, the operator has a very limited set of options aside from upgrading the device.

Time-to-Market Challenges

As requirements grow and change over time, equipment isn't always able to quickly keep up with these changes. Service providers often delay offering new services to meet the shift in the market requirements. Implementing new services requires upgrading the networking equipment. This leads to complex decisions to choose the appropriate migration path. This route may imply re-evaluation of new equipment, redesign of the network, or possibly new vendors that may be more suitable to meet the new needs. This increases the cost of ownership and longer timeline to offer new services to customers, resulting in loss of business and revenue.

Manageability Issues

Monitoring tools employed in the networks implement standardized monitoring protocols such as a Simple Network Management Protocol (SNMP), NetFlow, syslog, or similar systems for gathering device state and information. However, for monitoring vendor-specific parameters, relying on standard protocols may not suffice. For example, a vendor may be using nonstandard MIB or vendor-defined syslog messages. For such in-depth level of monitoring and control the management tools become very specific and tailored for the vendor's implementation. Whether these management tools are built in-house or offered directly by the vendors, it is sometimes not feasible to port these to a different vendor's devices.

High Operational Costs

The operational costs are high because of the need to have highly trained teams for each vendor-specific system being deployed in the network. This also tends to lock the provider into a specific vendor, because switching to a different vendor would mean additional costs to retrain operational staff and revamp operational tools.

Migration Considerations

Devices and networks need to be upgraded or reoptimized over a period of time. This requires physical access and on-site personnel to deploy new hardware,

reconfigure physical connectivity, and upgrade facilities at the site. This creates a cost barrier for migration and network upgrade decisions, slowing down the offering of new services.

Capacity Over-Provisioning

Short- and long-term network capacity demands are hard to predict, and as a result networks are built with excess capacity and are often more than 50% undersubscribed. Underutilized and overprovisioned networks result in lower return on investment.

Interoperability

For faster time to market and deployment, some vendors try to implement new networking functionality before it is fully standardized. In many cases, this implementation becomes proprietary, which creates inter-operability challenges that require service providers to validate interoperability before deploying it in production environment.

Introducing NFV

In data centers, the server virtualization approach is already proven technology, where stacks of independent server hardware systems have mostly been replaced by virtualized servers running on shared hardware.

NFV builds on this concept of server virtualization. It expands the concept beyond servers, widening the scope to include network devices. It also allows the ecosystem to manage, provision, monitor, and deploy these virtualized network entities.

The acronym NFV is used as a blanket term to reference the overall ecosystem that comprises the virtual network devices, the management tools, and the infrastructure that integrates these software pieces with computer hardware. However, NFV is more accurately defined as the method and technology that enables you to replace physical network devices performing specific network functions with one or more software programs executing the same network functions while running on generic computer hardware. One example is replacing a physical firewall appliance with a software-based virtual machine. This virtual machine provides the firewall functions, runs the same operating system, and has the same look and feel—but on non-dedicated, shared, and generic hardware.

With NFV, the network functions can be implemented on any generic hardware that offers the basic resources for processing, storage, and data transmission. Virtualization has matured to the point that it can mask the physical device, making it possible to use commercial off the shelf (COTS) hardware to provide the infrastructure for NFV.

COTS

Commercial off the shelf (COTS) refers to any product or service that is developed and marketed commercially. COTS hardware refers to general-purpose computing, storage, and networking gear that is built and sold for any use case that requires these resources. It doesn't enforce usage of a proprietary hardware or software.

Figure 1-2 shows the transition from traditional network devices to NFV.

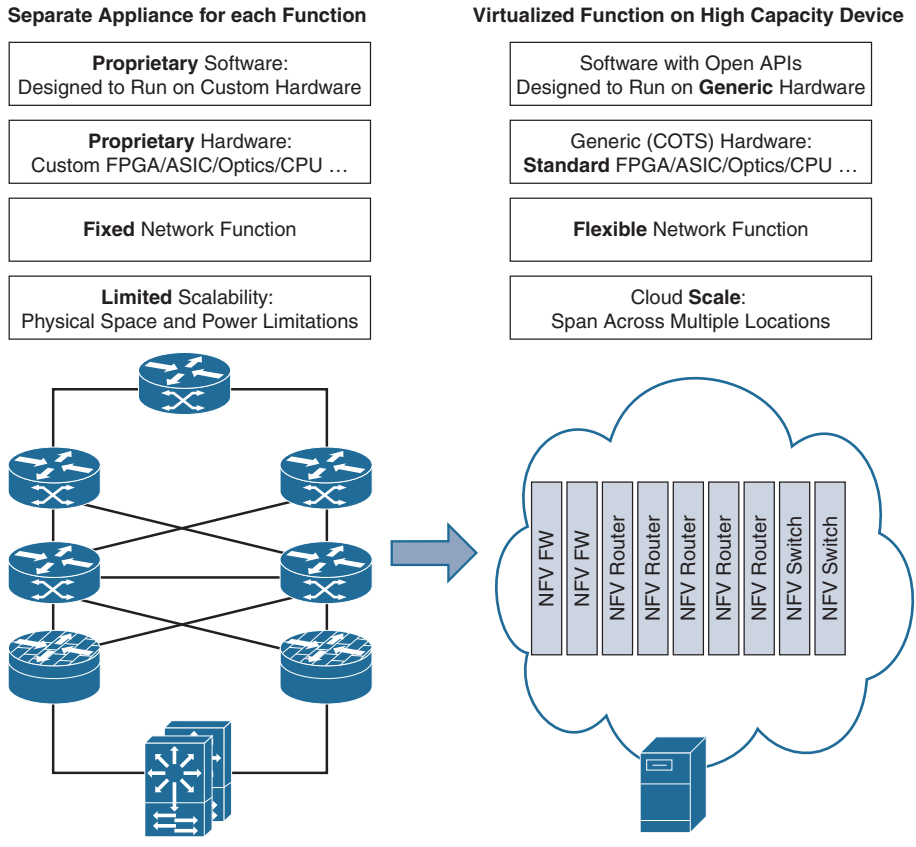

Figure 1-2 *Transition to NFV*

In traditional network architecture, vendors are not concerned about the hardware on which their code will run, because that hardware is developed, customized, and deployed as dedicated equipment for the specific network function. They have complete control over both the hardware and the software running on the device.

That allows the vendors flexibility to design the hardware and its performance factors based on the roles these devices will play in the network. For example, a device designed for the network core will have carrier-class resiliency built into it, while a device designed for the network edge will be kept simpler and will not offer high availability to keep its cost low. In this context, many of the capabilities of these devices are made possible with the tight integration of hardware and software. This changes with NFV.

In the case of virtualized network functions, it is not realistic to make assumptions about the capabilities that hardware has to offer, nor is it possible to very tightly integrate with the bare hardware. NFV decouples the software from hardware, and boasts to offer the ability to use any commercially available hardware to implement the virtualized flavor of very specific network functions.

Virtualization of networks opens up new possibilities in how networks can be deployed and managed. The flexibility, agility, capital and operational cost savings and scalability that is made possible with NFV opens up new innovation, design paradigm and enables new network architectures.

NFV Architectural Framework

The architecture that defines traditional network devices is fairly basic, because both the hardware and software are customized and tightly integrated. In contrast, NFV allows software developed by the vendors to run on generic shared hardware, creating multiple touch points for management.

The NFV architectural framework is developed to ensure that these touch points are standardized and compatible between the implementations of different vendors. This section provides a comprehensive discussion on the framework and the rationale behind its blocks. Understanding the framework enables readers to envision the flexibility and freedom of choice that NFV has to offer.

Need for a Framework

The architecture that defines the traditional network devices is fairly basic since both the hardware and software are customized and tightly integrated. In contrast, NFV allows software developed by the vendors to run on generic shared hardware creating multiple touch points for management. In the NFV jargon the virtual implementation of the network functions is referred to as virtualized network function (VNF). A VNF is meant to perform a certain network function e.g. router, switch, firewall, load-balancer, etc. and a combination of these VNFs may be required to implement the complete network segment that is being virtualized.

VNF

VNF (virtualized network function) replaces a vendor's specialized hardware with systems performing the same function, yet running on a generic hardware.

Different vendors may offer these VNFs, and the service providers can choose a combination of vendors and functions that best suit their needs. This freedom of choice creates the need for a standardized method of communication between the VNFs as well as a way to manage them in the virtual environment. The management of NFV needs to take into account the following considerations:

- multivendor implementations of VNFs
- managing the life cycles and interactions of these functions
- managing the hardware resource allocations
- monitoring the utilization
- configuration of the VNFs
- interconnection of the virtualized functions to implement the service
- interaction with the billing and operational support systems

To implement these management roles and keep the system open and nonproprietary, a framework must be defined for standardization. This standard framework should ensure that the VNF deployed is not tied to specific hardware and does not need to be especially tailored for any environment. It should offer vendors a reference architecture that they can follow for consistency and uniformity in the deployment methodologies of any VNF they implement. Additionally, it needs to ensure that the management of these VNFs and the hardware they run upon does not have a dependency on any vendor. There should be no special tweaking required to implement the network functions in this heterogeneous ecosystem. Essentially, this framework must provide the architectural foundations that allow the VNFs, hardware, and the management systems to work seamlessly within the well defined boundaries.

ETSI Framework for NFV

NFV was first introduced at the SDN OpenFlow World Congress in 2012 by a consortium of key service providers. They referenced the major challenges faced by network operators, especially their dependency on introducing new hardware for enabling innovative services to their customers. The group highlighted the challenges associated with the following concepts:

- design changes around the new equipment
- deployment cost and physical constraints
- need for expertise to manage and operate the new proprietary hardware and software
- dealing with hardware complexity in the new proprietary equipment
- the short lifecycle that makes this equipment become obsolete rapidly
- restarting the cycle before the returns from the capital expenses and investments are fully realized

The group proposed NFV as a way to tackle these challenges and improve efficiency by "leveraging standard IT virtualization technology to consolidate many network equipment types onto industry standard high volume servers, switches and storage, which could be located in Datacentres, Network Nodes and in the end user premises." [3]

To realize this goal and define a set of specifications that would make it possible to move from the traditional vendor and network centric approach to an NFV-based network, seven of these leading telecom operators formed an Internet specification group (ISG)—under an independent standardization organization called the European Telecommunications Standards Institute (ETSI). [1]

This group formally started in early 2013, working towards defining requirements and an architectural framework that can support the virtualized implementation of network functions performed by custom hardware devices from vendors.

This group used three key criteria for coming up with the recommendations:

- **Decoupling:** complete separation of hardware and software
- **Flexibility:** automated and scalable deployment of the network functions
- **Dynamic operations:** control of the operational parameters of the network functions through granular control and monitoring of the state of network

Based on these criteria, a high-level architectural framework was established, defining distinct areas of focus as shown in Figure 1-3.

This architectural framework forms the basis of the standardization and development work and is commonly referred to as the ETSI NFV framework. At a high level, the framework encompasses management of VNFs, relationships and interdependencies, data flow between VNFs, and resource allocation. ETSI ISG categorized these roles into three high-level blocks, namely the infrastructure block, virtualized

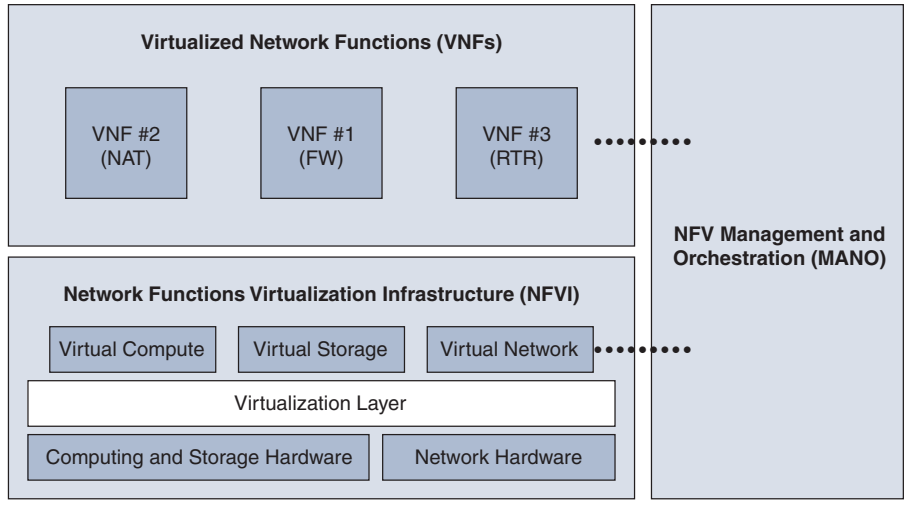

Figure 1-3 *High-Level ETSI NFV Framework*

functions block, and management block. In ETSI's definition, the formal names of these blocks are defined as:

- **Network Functions Virtualization Infrastructure (NFVI) block:** This block forms the foundation of the overall architecture. The hardware to host the virtual machines, the software to make virtualization possible, and the Virtualized resources are grouped into this block.
- **Virtualized Network Function (VNF) block:** The VNF block uses the virtual machines offered by NFVI and builds on top of them by adding the software implementing the virtualized network functions.
- **Management and Orchestration (MANO) block:** MANO is defined as a separate block in the architecture, which interacts with both the NFVI and VNF blocks. The framework delegates to the MANO layer the management of all the resources in the infrastructure layer; in addition, this layer creates and deletes resources and manages their allocation of the VNFs.

Understanding the ETSI Framework

The ETSI framework and the thought process behind its high-level blocks can be better understood if you examine the building process that led to this framework. Let's begin with the fundamental concept of NFV, such as virtualizing the function of a network device. This is achieved through VNFs.

To implement the network service, VNFs may be deployed either as standalone entities or as a combination of multiple VNFs. The protocols associated with the function that is being virtualized within a VNF do not need to be aware of the virtualized implementation. As shown in the Figure 1-4 the VNF implementing the firewall service (FW), NAT device (NAT), and routing (RTR) communicate to each other without the knowledge that they are not physically connected or running on dedicated physical devices.

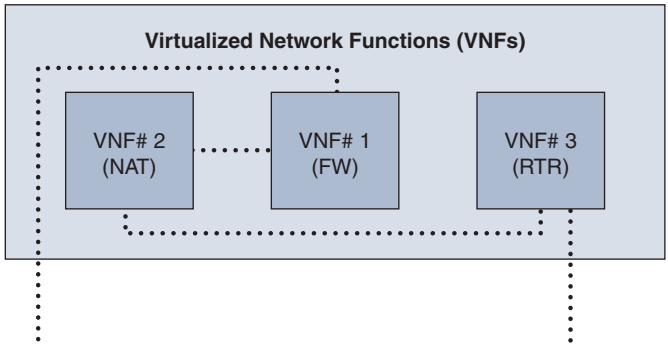

Figure 1-4 *Network Functions Working Together as VNFs*

Since there isn't dedicated or custom hardware designed to run these VNF, a general-purpose hardware device with generic hardware resources such as a processor (CPU), storage, memory, and network interfaces can be used to run these VNFs. This can be made possible by using COTS hardware. It doesn't need to be a single COTS device; it can be an integrated hardware solution providing any combination of the required hardware resources to run the VNFs. Virtualization technologies can be used to share the hardware among multiple VNFs. These technologies, such as hypervisor-based virtualization or container-based virtualization, have been used in data centers for some time and have become fairly mature. These details are covered in Chapter 2, "Virtualization Concepts."

Virtualization of hardware offers an infrastructure for the VNF to run upon. This NFV infrastructure (NFVI) can use COTS hardware as a common pool of resources and carve out subsets of these resources creating "virtualized" compute, storage, and network pools that can be allocated as required by the VNFs, as shown Figure 1-5.

The vendor for the VNF recommends a minimum requirement for the resources that its implementation should have available to it, but the vendor can't control or optimize these hardware parameters. For instance, the vendor can make a recommendation on the CPU cores necessary to execute the code or the storage space and memory the VNF will need—but the vendors no longer get a free hand to design

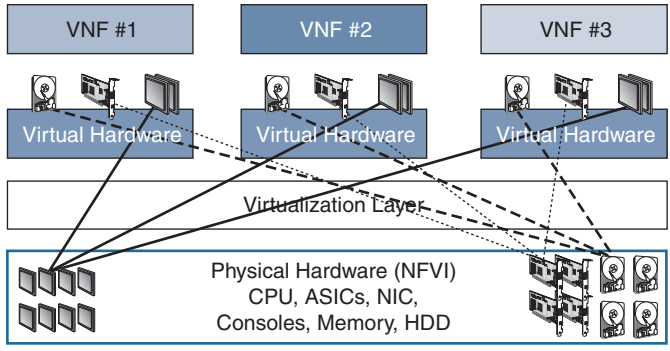

Figure 1-5 *Virtual Computing, Storage, and Networking Resources Provided to VNF*

the hardware around their specific requirements. The virtualization layer using the physical hardware can cater to the VNF resource request. The VNF doesn't have any visibility into this process, nor is that VNF aware of the existences of other VNFs that may be sharing the physical hardware with them.

In this virtualized network's architecture, there are now multiple resources to manage and operate at various levels. In comparison, today's network architecture management is vendor specific and has limited knobs and data points offered by vendors. Any new requirements or enhancements in management capabilities are possible only with vendor support. With NFV it is possible to manage the entities at a more granular and individual level. The NFV architecture, therefore, wouldn't be complete without defining the methodologies to manage, automate, coordinate, and interconnect these layers and functional blocks in an agile, scalable, and automated way.

This requirement leads us to add another functional block to the framework that communicates with and manages both the VNF and NFVI blocks, as shown in Figure 1-6. This block manages the deployment and interconnections of the VNFs on the COTS hardware and allocates the hardware resources to these VNFs.

Since the MANO block is meant to have full visibility of the entities and is responsible for managing them, it is fully aware of the utilization, operational state, and usage statistics of them. That makes MANO the most suitable interface for the operational and billing systems to gather the utilization data.

This completes the step-by-step understanding of the three high-level blocks—NFVI, VNF, and MANO—and captures the reasoning behind defining and positioning these blocks in the ETSI framework.

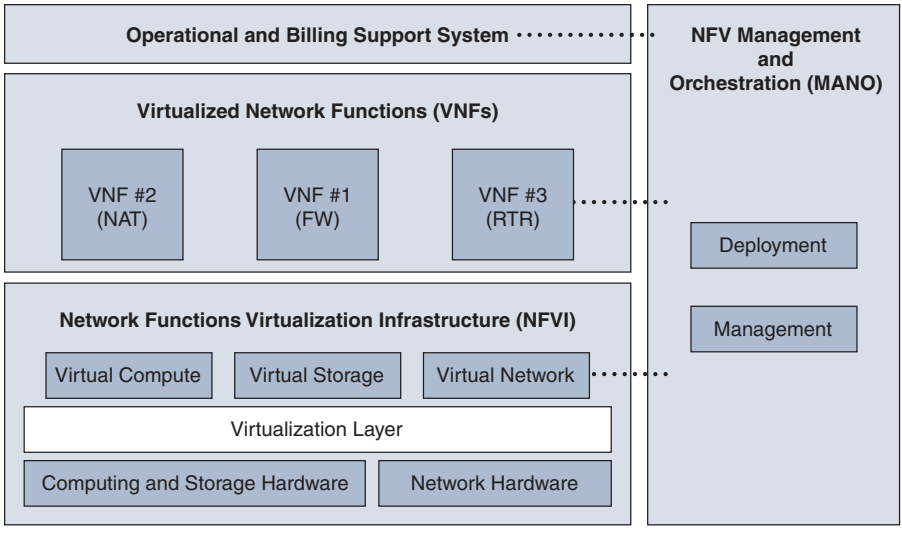

Figure 1-6 *Management and Orchestration Block for NFV*

A Closer Look at ETSI's NFV Framework

The previous section provides a high-level view of the ETSI NFV architecture framework and its basic building blocks. The framework defined by ETSI goes deeper into each of these blocks and defines individual functional blocks with distinct role and responsibility for each of them. The high-level blocks, therefore, comprise multiple functional blocks. For instance, the management block (MANO) is defined as a combination of three functional blocks: the Virtualized Infrastructure Manager (VIM), Virtualized Network Function Manager (VNFM), and NFV Orchestrator (NFVO).

The architecture also defines reference points for the functional blocks to interact, communicate and work with each other. Figure 1-7 shows the detailed view of the framework as defined by ETSI.

This section takes a deeper look into this framework and reviews the suggested functions, the interworking of each of these functional blocks, and their interlinking through the reference points.

For convenience of understanding, these functional blocks are grouped into layers, where each layer deals with a particular aspect of NFV implementation.

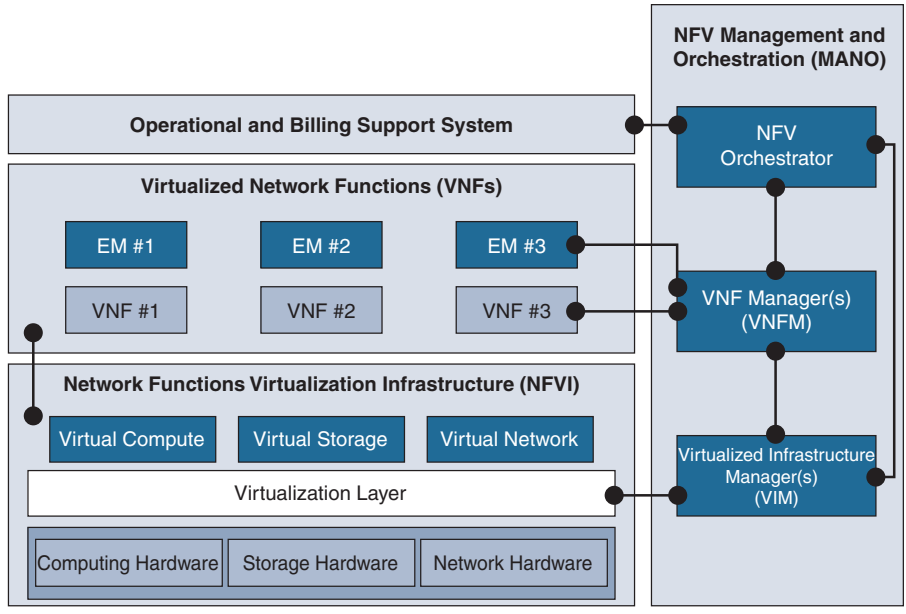

Figure 1-7 *Low Level View of the ETSI NFV Framework*

Infrastructure Layer

The VNFs rely on the availability of virtual hardware, emulated by software resources running on physical hardware. In the ETSI NFV framework, this is made possible by the infrastructure block (NFVI). This infrastructure block comprises physical hardware resources, the virtualization layer, and the virtual resources, as shown in Figure 1-8.

ETSI framework splits the hardware resources into three main categories – computing, storage, and network. The computing hardware includes both the CPU and memory, which may be pooled between hosts using cluster-computing techniques. Storage can be locally attached or distributed with devices such as network-attached storage (NAS) or devices connected using SAN technologies. Networking hardware comprises pools of network interface cards and ports that can be used by the VNFs. None of this hardware is purposely built for any particular network function, but all items are instead generic hardware devices available off the shelf hardware (COTS). These functional blocks can span and scale across multiple devices and interconnected locations, and are not confined to a single physical host, location or point of presence (POP).

It must be mentioned that the networking hardware within the physical location interconnecting the storage and compute devices, or interconnecting multiple locations (such as switches, routers, optical transponders, wireless communication

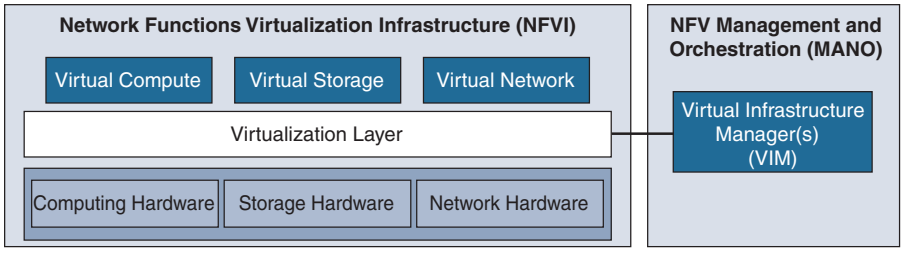

Figure 1-8 *Infrastructure Layer of ETSI NFV Framework*

equipment, etc.) is also considered part of NFVI. However, these network devices are not part of the pool that is allocated as a virtual resource to VNF.

The virtualization layer is another function block that is part of NFVI. It interacts directly with the pool of hardware devices, making them available to VNFs as a virtual machine. The virtual machine offers the virtualized computing, storage, and networking resources to any software that it hosts (VNF in this case) and presents these resources to the VNF as if they were dedicated physical hardware devices.

VM

Virtual machine or VM is a commonly used terminology for the virtualized resource pool, which may be shared hardware resources working independently and isolated from each other.

In summary, it is the virtualization-layer that is decoupling the software for network function (i.e., VNF) from the hardware while provident them isolation from other VNFs and acting as an interface to the physical hardware.

Abstraction

The technique of decoupling hardware and software layers by providing a common independent interface to the software for accessing the hardware resources is referred to as "hardware abstraction", or more simply as "abstraction."

To manage NFVI, ETSI defines a management functional block called the Virtualized Infrastructure Manager (VIM). VIM is part of MANO (Management and Orchestration blocks), and the framework delegates to it the responsibility for managing the computing, storage, and networking hardware, the software that is implementing the virtualization layer, and the virtualized hardware. Because VIM directly manages the hardware resources, it has a full inventory of these resources and visibility into their operational attributes (such as power management, health status, and

availability), as well as the capacity to monitor their performance attributes (such as utilization statistics).

VIM also manages the virtualization layer and controls and influences how the virtualization layer uses the hardware. VIM is therefore responsible for the control of NFVI resources and works with other management functional blocks to determine the requirements and then manage the infrastructure resources to fulfill them. VIM's management scope may be with the same NFVI-POP or spread across the entire domain spanned by the infrastructure.

An instance of VIM may not be restricted to a single NFVI layer. It is possible that a single VIM implementation controls multiple NFVI blocks. Conversely, the framework also allows for the possibility that multiple VIMs can function in parallel and control several separate hardware devices. These VIMs can be in a single location or different physical locations.

Virtualized Network Functions (VNF) Layer

The VNF layer is where the virtualization of network function is implemented. It comprises the VNF-block and the functional block that manages it, called VNF-Manager (VNFM). The VNF-block is defined as a combination of VNF and Element Management (EM) blocks as shown in Figure 1-9.

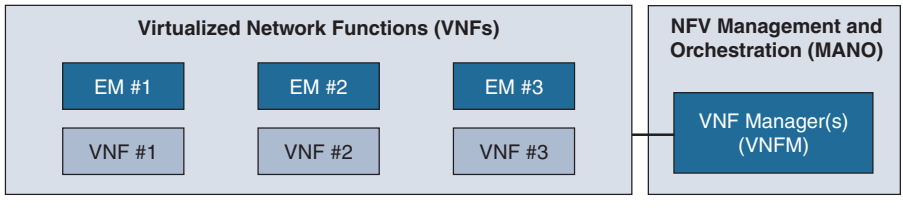

Figure 1-9 *Virtualized Network Function Layer in ETSI NFV Framework*

A virtualized implementation of a network function needs to be developed so it can run on any hardware that has sufficient computing, storage, and network interfaces. However, the details of the virtualized environment are transparent to the VNF, and it is expected to be unaware that the generic hardware it is running on is actually a virtual machine. The behavior and external interface of the VNF is expected to be identical to the physical implementation of the network function and device that it is virtualizing.

The network service being virtualized may be implemented through a single VNF, or it may require multiple VNFs. When a group of VNF are collectively implementing the network service, it is possible that some of the functions have dependencies on others, in which case the VNF needs to process the data in a specific sequence.

When a group of VNFs doesn't have any interdependency, then that group is referred to as a VNF set. An example of this is in a mobile virtual Evolved Packet Core (vEPC), where the Mobile Management Entity (MME) is responsible for authentication of the user and chooses the Service Gateway (SGW). The SGW runs independently of the MME's function and forwards user data packets. These VNFs work collectively to offer part of the functionality of vEPC but are independently implementing their functions.

If, however, the network service requires VNFs to process the data in a specific sequence, then the connectivity between the VNFs needs to be defined and deployed to ensure it. This is referred to as VNF-Forwarding-Graph (VNF-FG) or service chaining. In the previous example of vEPC, if you added another VNF that provides Packet Data Network Gateway (PGW) functionality, that PGW VNF should only process the data after the SGW. As shown in Figure 1-10, this interconnection between SGW, MME, and PGW in this specific order for packet flow makes a VNF-FG. This idea of service chaining is important in the NFV world and requires a more detailed discussion. This topic is covered in depth in Chapter 6, "Stitching It All Together."

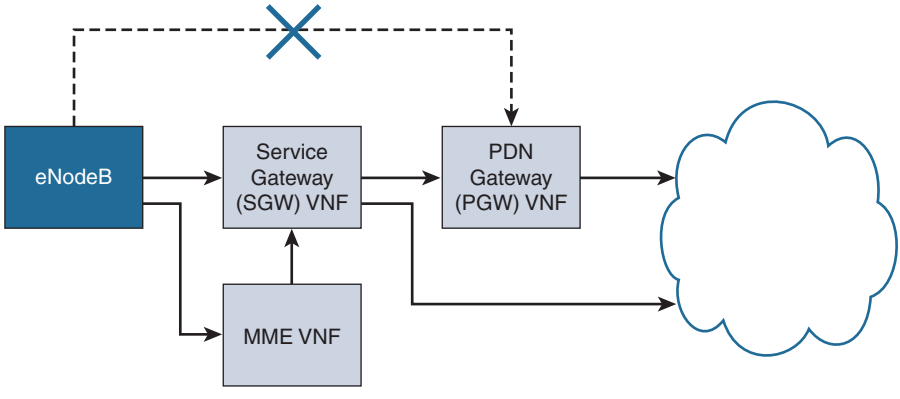

Figure 1-10 *Virtual Evolved Packet Core (vEPC) using VNF-FG*

In the ETSI framework, it is the VNFM's responsibility to bring up the VNF and manage the scaling of its resources. When the VNFM must instantiate a new VNF or add or modify the resources available to a VNF (for example, more CPU or memory) it communicates that requirement to the VIM. In turn, it requests that the virtualization layer modify the resources allocated to the VM that is hosting the VNF. Since the VIM has visibility into the inventory, it can also determine if it is possible for the current hardware to cater to these additional needs. Figure 1-11 shows this flow of events.

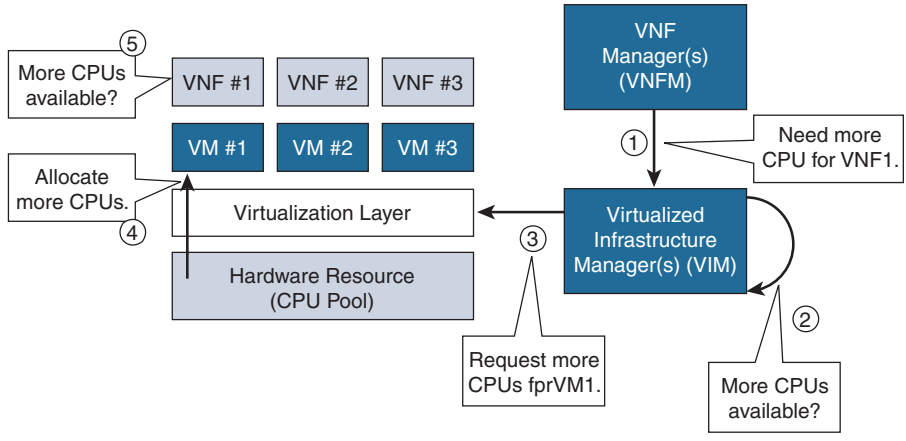

Figure 1-11 *VNFM Scaling Up VNF Resources*

The VNFM also has the responsibility for the FCAPS of the VNFs. It manages this directly by communicating with the VNFs or uses the Element Management (EM) functional block.

FCAPS

FCAPS is a ISO telecommunications management network mode and is an abbreviation for the five main management parameters: fault, configuration, performance, accounting, and security.

Element Management is another functional block defined in the ETSI framework and is meant to assist in implementing the management functions of one or more VNFs. The management scope of EM is analogous to the traditional element management system (EMS), which serves as a layer for interaction between the network management system and the devices performing network functions. EM interacts with the VNFs using proprietary methods while employing open standards to communicate with the VNFM. This provides a proxy to the VNFM for operations and management of the VNFs as shown in Figure 1-12. The FCAPS are still managed by VNFM, but it can take support from the EM to interact with the VNF for this aspect of management.

The framework doesn't restrict the implementation to a single VNFM to manage all the VNFs. It is possible that the vendor that owns the VNF requires its own VNFM to manage that VNF. Therefore, there can be NFV deployments where multiple VNFM are managing multiple VNFs or a single VNFM manages a single VNF, as shown in Figures 1-13 and 1-14.

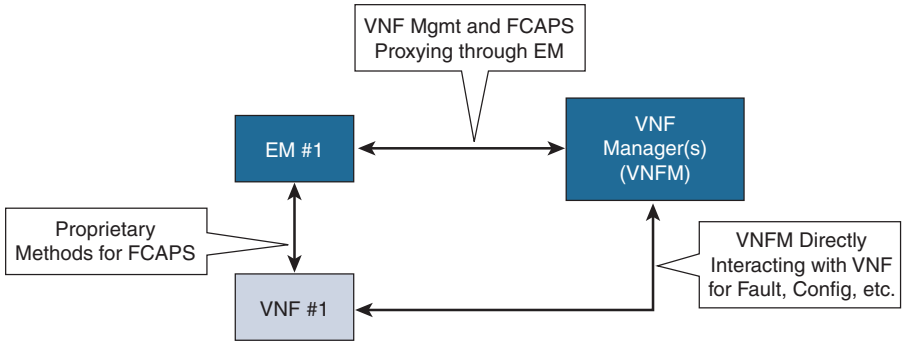

Figure 1-12 *VNFM Managing VNF Directly or through EM*

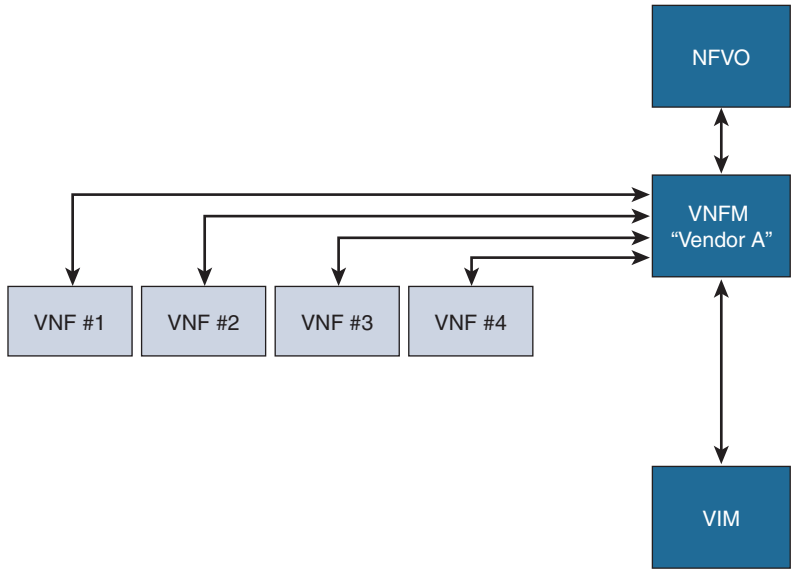

Figure 1-13 *Single VNFM Managing Multiple VNFs*

Operational and Orchestration Layer

When moving from physical to virtual devices, network operators do not want to revamp the management tools and applications that may be deployed for operational and business support systems (OSS/BSS). The framework doesn't require a change in these tools as part of transformation to NFV. It allows them to continue to manage the

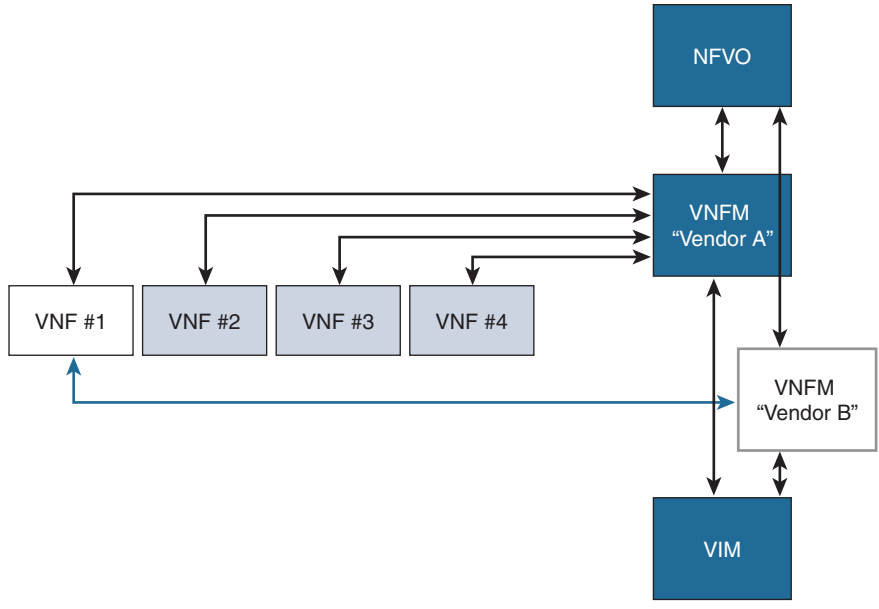

Figure 1-14 *Multiple VNFMs Managing Separate VNFs*

operational and business aspects of the network and work with the devices even though the devices are replaced by VNFs. While this is in line with what is desired, using existing systems has its drawbacks, because it doesn't fully reap the benefits of NFV and is not designed to communicate with NFV's management functional blocks—VNFM and VIM. One path that providers can take is to enhance and evolve the existing tools and systems to use NFV management functional blocks and utilize the NFV benefits (like elasticity, agility, etc.). That's a viable approach for some, but it is not a feasible option for others because these systems are traditionally built in-house or are proprietary implementations that do not allow for managing an open platform like NFV.

The solution that the ETSI framework offers is to use another functional block, NFV Orchestrator (NFVO). It extends the current OSS/BSS and manages the operational aspects and deployment of NFVI and VNF Figure 1-15 shows the two components of the orchestration layer in the framework.

The role of NFVO is not obvious up front and seems like an additional block buffering between current operating tools and VIM and VFNM. NFVO, however, has a critical and important role in the framework by overlooking the end-to-end service deployment, parsing the bigger picture of service virtualization and communicating the needed pieces of information to VIM and VNFM for implementing that service.

NFVO also works with the VIM(s) and has the full view of the resources that they are managing. As indicated previously, there can be multiple VIMs and each one of

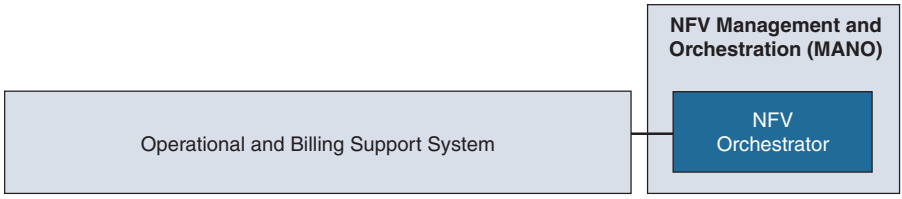

Figure 1-15 *Operational and Orchestration Layer of ETSI NFV Framework*

them has only visibility of the NFVI resources that it is managing. Since NFVO has the collective information from these VIMs, it can coordinate the resource allocation through the VIMs.

Resource orchestration

The process of allocating, deallocating, and managing NFVI resources to the VMs is referred to as resource orchestration.

Similarly, the VNFM is independently managing the VNFs and doesn't have visibility into any connection of the services between the VNFs and how the VNFs combine to form the end to the service path. This knowledge resides in the NFVO, and it's the role of NFVO to work through the VNFM to create the end-to-end service between the VNFs. It is therefore the NFVO that has visibility into the network topology formed by the VNFs for a service instance.

Service Orchestration

The term Service Orchestration refers to defining the service using the VNFs and how these VNFs will interconnect as a topology to implement it.

Despite not being a part of the NFV transformation, the existing OSS/BSS do bring value to management and therefore have a place in the framework. The framework defines the reference points between the existing OSS/BSS and NFVO and defines NFVO as an extension of the OSS/BSS to manage the NFV deployment without attempting to replace any of the roles of OSS/BSS in today's networks.

NFV Reference Points

The ETSI framework defines reference points to identify the communication that must occur between the functional blocks. Identifying and defining these is

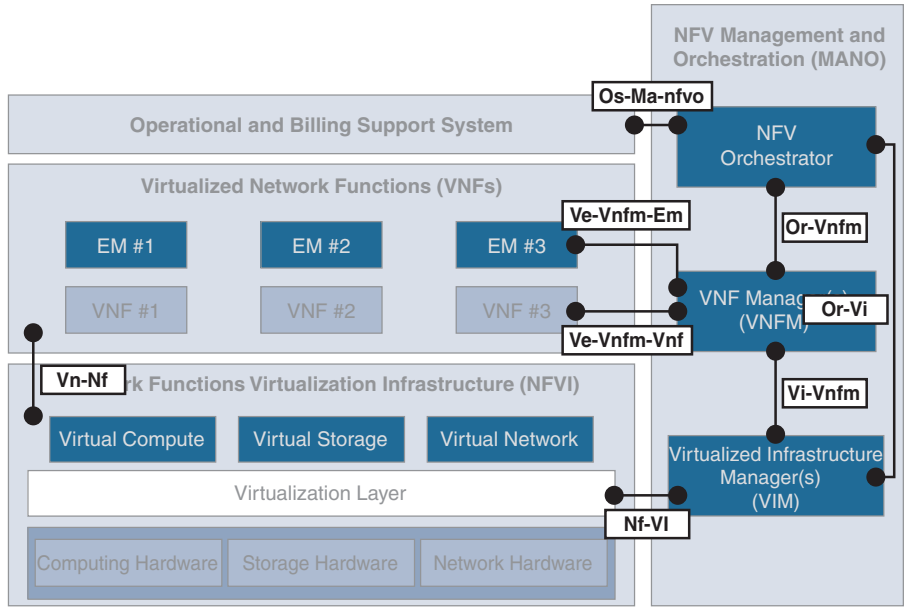

Figure 1-16 *ETSI NFV Framework Reference Points*

important to ensure that the flow of information is consistent across the vendor implementation for functional blocks. It also helps established an open and common way to exchange information between the functional blocks. Figure 1-16 shows the reference points defined by the ETSI NFV framework.

The list that follows describes these reference points in more detail.

- **Os-Ma-nfvo:** This was originally labeled Os-Ma and is meant to define the communication between OSS/BSS and NFVO. This is the only reference point between OSS/BSS and the management block of NFV (MANO).
- **Ve-Vnfm-vnf:** This reference point defines the communication between VNFM and VNF. It is used by VNFM for VNF lifecycle management and to exchange configuration and state information with the VNF.
- **Ve-Vnfm-em:** This was originally defined together with Vn-Vnfm-vnf (jointly labeled Ve-Vnfm) but is now defined separately for communication between the VNFM and EM functional blocks. It supports VNF lifecycle management, fault and configuration management, and other functions, and it is only used if the EM is aware of virtualization.

- **Nf-Vi:** This reference point defines the information exchange between VIM and the functional blocks in NFVI. VIM uses it to allocate, manage, and control the NFVI resources.
- **Or-Vnfm:** Communication between NFVO and VNFM happens through this reference point, such as VNF instantiation and other VNF lifecycle-related information flow.
- **Or-Vi:** The NFV orchestrator (NFVO) is defined to have a direct way of communicating with VIM to influence the management of the infrastructure resources, such as resource reservation for VMs or VNF software addition.
- **Vi-Vnfm:** This reference point is meant to define the standards for information exchange between VIM and VNFM, such as resource update request for VM running a VNF.
- **Vn-Nf:** This is the only reference point that doesn't have a management functional block as one of its boundaries. This reference point is meant to communicate performance and portability needs of the VNF to the infrastructure block.

Table 1-1 summarizes these reference point definitions:

Putting it all Together

Let's see how this model works end to end, taking the example of a simple network service and examining how the functional blocks defined in the ETSI framework collectively interact to implement the service. Figure 1-17 shows a simplified version of the steps involved.

The following steps depict this process:

Step 1. The full view of the end-of-end topology is visible to the NFVO.

Step 2. The NFVO instantiates the required VNFs and communicate this to the VNFM.

Step 3. VNFM determines the number of VMs needed as well as the resources that each of these will need and reverts back to NFVO with this requirement to be able to fulfill the VNF creation.

Step 4. Because NFVO has information about the hardware resources, it validates if there are enough resources available for the VMs to be created. The NFVO now needs to initiate a request to have these VMs created.

Table 1-1 *ETSI NFV Framework Reference-Points*

Reference Point	Boundaries	Use Defined in the Framework
Os-Ma-nfvo	OSS/BSS<->NFVO	• Service description and VNF package management. • Network service lifecycle management (instantiation, query, update, scaling, and termination). • VNF life cycle management. • Policy management (access, authorization, etc.) for network service instances. • Querying network service and VNF instances from OSS/BSS. Forwarding events, usage, and performance of network service instances to OSS/BSS.
Ve-Vnfm-vnf	VNFM<->VNF	• Instantiation, instance query, update, scaling up or down, and termination of the VMs. • Configuration and events regarding VNF, from VNFM to VNF. • Configuration and events from VNF to VNFM.
Ve-Vnmf-em	VNFM<->EM	• Instantiation, instance query, update, scaling up or down, and termination of the VMs. • Configuration and events regarding VNF from VNFM to EM. • Configuration and events from EM to VNFM.
Nf-Vi	NFVI<->VIM	• Allocate, update. migrate, terminate VMs. • Create, configure, remove inter-VM connections. • Failure events, usage records, configuration information to the VIM for NFVI resources (physical, software, virtual).
Or-Vnfm	NFVO<->VNFM	• Instantiation, state query, update, scaling, termination and package query of the VNF. • Forwarding VNF events and state information.
Or-Vi	NFVO<->VIM	• NFVI resource reservation, release, and update. • VNF software image allocation, deallocation, and update. • Configuration, usage, events, and results of NFVI to NFVO.
Vi-Vnfm	VIM<->VNFM	• NFVI resource reservation, allocation, and release information. • Events, usage, measurement results, etc. for a NFVI resource used by a VNF.
Vn-Nf	NFVI<->VNF	• Lifecycle, performance, and portability requirements of VNF.

Step 5. NFVO sends request to VIM to create the VMs and allocate the necessary resources to those VMs.

Step 6. VIM asks the virtualization layer to create these VMs.

Step 7. Once the VMs are successfully created, VIM acknowledges this back to NFVO.

Step 8. NFVO notifies VNFM that the VMs it needs are available to bring up the VNFs.

Step 9. VNF now configures the VNFs with any specific parameters.

Step 10. Upon successful configuration of the VNFs, VNFM communicates to NFVO that the VNFs are ready, configured, and available to use.

Figure 1-17 and the accompanying list depict a simplified flow as an example to help understand the framework. It intentionally doesn't go into many more details associated with this process as well as possible variations. Though these are not being covered in this book, readers can refer to the ETSI document (Section 5, in [2]) for additional details and scenarios.

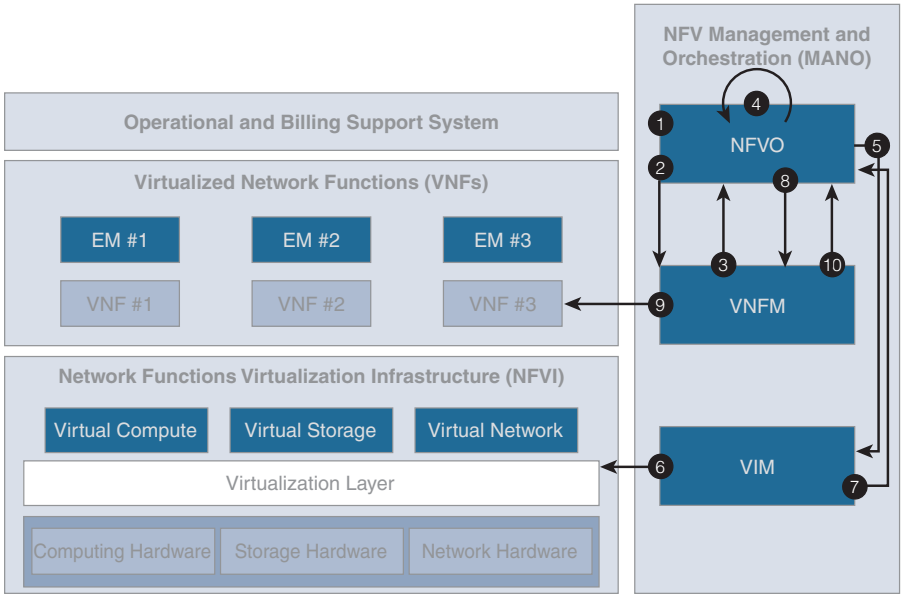

Figure 1-17 *End-to-End Flow in the ETSI NFV Framework*

NFV Framework Summary

The goal of defining the framework and more specifically the individual functional blocks and the reference points is to eliminate (or more realistically, minimize) interoperability challenges and standardize the implementation. The purpose and scope of each of these blocks is well defined in the framework. Similarly, the interdependencies and communications paths are defined through the reference-points and are meant to be open and standard methods.

Vendors can independently develop these functions and deploy them to work smoothly with other functional blocks developed by other vendors. As long as these implementations adhere to the scope and role defined by the framework, communicate with the other blocks using open methods at the reference points, the network can have a heterogeneous deployment of NFV. This means that the service providers will have complete flexibility to choose between vendors for different functional blocks. This is in contrast to the way networks have traditionally been deployed, where service providers were tied to the vendor's hardware (and its limitations) and software (and the challenges to adapt to it for all operational needs), and they had to deal with the interoperability concerns of mixed vendor networks. NFV offers service providers the ability to overcome this limitation and deploy a scalable and agile network using hardware and NFV functional blocks using any combination of vendors.

This doesn't magically eliminate the higher-level protocol interoperability issues that may arise between VNFs implemented by different vendors. For example, a BGP implementation by a vendor of one VNF may have some issue when it is peering with another VNF developed by a different vendor. For these types of interoperability issues, a standardization process already exists and will continue to play a role. Also, NFV doesn't mandate that vendors offer an open and standard way to manage the configuration and monitoring of the VNFs. EM in the NFV framework compensates for that. But in an implementation closer to the ideal, the operations support system should be able to work with the VNFs using standard methods. This is happening through a parallel technology shift towards software-defined networking (SDN). Though NFV and SDN are not interdependent, together they are complementing the benefits and advantages of each other. In this book, the focus is on NFV, but the picture is not complete without some discussion of SDN and how these two complement each other.

Though the NFV framework is well established, the standardization of NFV building blocks is an ongoing effort.

Benefits of NFV

Earlier in this chapter the limitations associated with using the traditional network equipment were listed. Network functions virtualization directly addresses most of

these restrictions and brings many additional benefits. It offers a framework to completely transform the way networks are architected, deployed, managed, and operated, while offering many layers of improvement and efficiency across all of these. Figure 1-18 lists a few of the benefits that NFV offers that are discussed in the sections that follow.

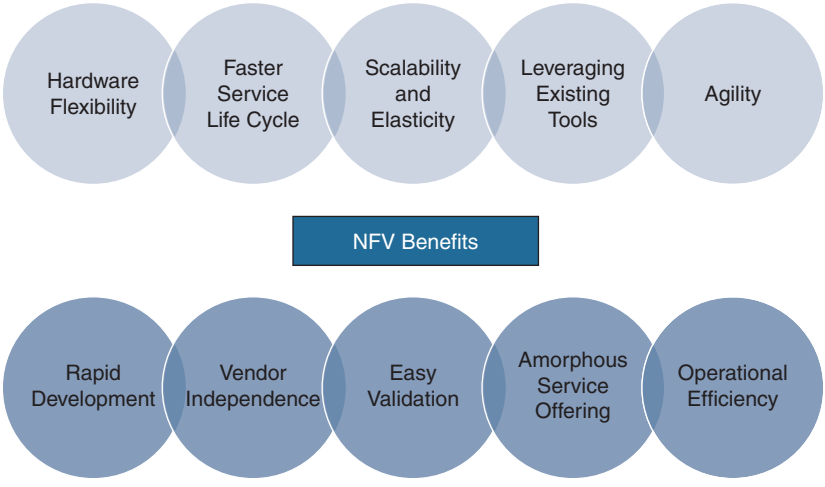

Figure 1-18 *Some Benefits of Network Functions Virtualization*

Hardware Flexibility

Because NFV uses regular COTS hardware, network operators have the freedom to choose and build the hardware in the most efficient way to suit their needs and requirements.

Hardware offered by traditional network vendors has very limited options for its computing, memory, storage, and networking capacities, and any modification leads to a hardware upgrade that costs time and money to the operators. With NFV, providers can now choose between many different vendors and have the flexibility to select the hardware capacities that are optimal for their network architecture and planning. For example, if the Internet gateway being used is running out of capacity to store the full Internet table and needs a memory upgrade, in most current implementations they can achieve this only through a controller upgrade or full device upgrade. In NFV, the provider can allocate more memory to the VM hosting this VNF.

Faster Service Life Cycle

New network services or features can now be deployed more quickly, in an on-demand and on-need basis, providing benefits for end users as well as the network providers.

In contrast to physical hardware, the VNFs can be created and removed on the fly. The lifecycle of VNFs can be much shorter and dynamic compared to physical devices, since these functions can be added when needed, provisioned easily through automated software tools that do not require any on-site activity, and then torn down to free up resources as soon as the need is over. This is in contrast to the deployment effort needed when a new function has to be added to an existing network, which would have required an on-site physical installation, which can be time consuming and costly. The ability to rapidly add new network functions (deployment agility) is one of the biggest advantages of NFV. Services now can also be commissioned or decommissioned with the touch of a button without the need of a delivery truck, drastically reducing deployment times from weeks to minutes.

Agility

The ability to rapidly deploy, terminate, reconfigure or change the topological location of a VNF is commonly referred to as deployment agility.

Scalability and Elasticity

New services and capacity-hungry applications in today's networks keep network operators, especially cloud providers, on their toes to keep up with the fast-increasing demands of consumers. The service providers have been playing catch-up with these requirements, for scaling up the traditional network equipment's capacity takes time, planning, and money. This problem is solved by NFV, which allows capability changes by offering a means to expand and shrink the resources used by the VNFs. For instance, if any of the VNFs requires additional CPU, storage, or bandwidth, it can be requested from the VIM and allocated to the VNF from the hardware pool. In a traditional network device, it would require either a full device replacement or a hardware upgrade to alter any of these parameters. But since VNFs aren't constrained by the limitations of customized physical hardware, they can offer this elasticity. Therefore networks do not need to be substantially overprovisioned to accommodate changes in capacity requirements.

Another way the NFV can implement elasticity is by offloading a VNF's workload and spinning off a new instance to implement the same network function and split the load with an existing VNF. This too is not possible with traditional network equipment.

Elasticity

Elasticity is a word very commonly used in the NFV context to refer to the capability of a VNF to expand and stretch resources or to shirk and scale them down, based on requirements. Also, this term is used to refer to the scenario when we create or remove additional VNFs to share the workload of an existing VNF.

Leveraging Existing Tools

As NFV uses the same infrastructure as data centers, it can reuse and leverage the deployment and management tools already being used in data centers. Using a single centralized pane of glass for management of virtual network and virtual servers gives the advantage of quicker adaption for new deployments without the need for developing new tools and as a result eliminates the cost of deploying, familiarizing, and using new set of tools.

Rapid Development and Vendor Independence

Because NFV provides the means to easily deploy a different vendor's solution without the heavy costs associated with replacing an existing vendor's deployment, it keeps network operators from being locked into a particular vendor. Operators can mix and match vendors and functions, and choose between them based on feature availability, cost of licensing the software, post deployment support model, roadmaps, etc.

New solutions and features can be put into production rapidly, without waiting for the existing deployed vendor to develop and support them. Such rapid deployment is further facilitated by NFV's inherent support for using open source tools and software.

Validation of New Solutions

Service providers often prefer to validate new solutions, services, and functions by deploying them in test setups, prior to introducing them in their production networks. Traditionally, they have had to replicate a subset of their production environment for in-house testing, which increased their operational budget. With NFV, building and managing such a test setups has become much more cost effective. The NFV-based test-setups can be dynamic and thus scaled and changed to meet the test and validation scenarios.

Amorphous Service Offering

An NFV-based deployment is not confined to a one-time design and deployment. It can adapt to market specific needs and offer a targeted set of services to match

changing demands. Through a combination of elasticity and deployment agility, it's possible to rapidly shift the location and capacities of network functions and achieve workload mobility. For example, providers can implement a "follow the sun network" by using constantly moving virtual machines based on time of the day, and spinning up or expanding new VNFs to meet the network's requirements for services and capacity as they change during peak and off-peak usage or when major events take place in any geographic region.

Operational Efficiency and Agility

With common hardware hosting different VNFs, tasks associated with running the business, such as inventory management, procurement process, can be centralized. This reduces the operational overhead compared to segregated deployments of different network services using multiple hardware devices.

NFV is inherently automation friendly, and can increase the benefits that can be achieved through use of Machine to Machine (M2M) tools. For instance, its possible for an automation tool monitoring a device to determine the need for more memory in a network function. With NFV that tool can go ahead and request allocation of that memory—without involving any human intervention.

Network maintenance related activates can also significantly benefit from NFV by reducing possible downtimes. NFV allows for spinning up a new VNF, temporarily shift the workload to that VNF, and free up existing VNF for maintenance activities. This makes it possible to achieve In-Service-Software-Upgrade (ISSU), 24/7 self-healing networks, and minimize the operational loss of revenue due to network outages.

Note

Upgrading to new software for introducing new features, scaling changes, bug fixes, etc. while maintaining a high uptime has traditionally been a challenge and sometimes a source of pain for network service providers. This problem becomes more critical in the network edge devices, for they are not generally deployed with physical redundancy. An In-Service Software Upgrade (ISSU) is if the term for one of the solutions offered by network vendors to enhance the upgrade procedure in a way that allows an upgrade to occur without disrupting the device's functionality. ISSU implementations may not always be completely without disruptions, and they could potentially result in a very brief loss of traffic. However, this brief possibility of traffic loss is sometimes acceptable and preferred to the certain loss of service if the device was upgraded without ISSU.

NFV Market Drivers

NFV is more than a transformative technology. Like any new technology that brings major changes and new benefits, NFV has had to go through acceptance and adaption by the market. The market drivers for NFV are very significant, obvious, and promising. These have played a part in making NFV move beyond its infancy in research labs and bringing it into mainstream deployments in a very short span of time.

Access to the Internet and the trend toward digital services across the world are creating a big market for the network service providers. The scale and the bandwidth needs are already straining the existing network infrastructure. Upgrading this traditional network infrastructure requires high levels of time, money, and resources from the providers. This has forced the providers to rethink the network architecture and use new innovations that could keep up with the new cloud and digitalization world. One of the main drivers is the movement to cloud technology coupled with the utilization of the matured technology such as the virtualization and COTS hardware. Network providers are now using the same cloud infrastructure such as the computers (servers) and storage devices and adding the network function to these elements to provide services for new market requirements. By taking this approach, they gain major cost savings, the ability to bring new services to market faster and capacity to adapt quickly to any change in the market landscape.

The NFV market drivers that bring new business opportunities have made network operators eager to transition to NFV. Figure 1-19 lists some of these market drivers, which are described in the sections that follow.

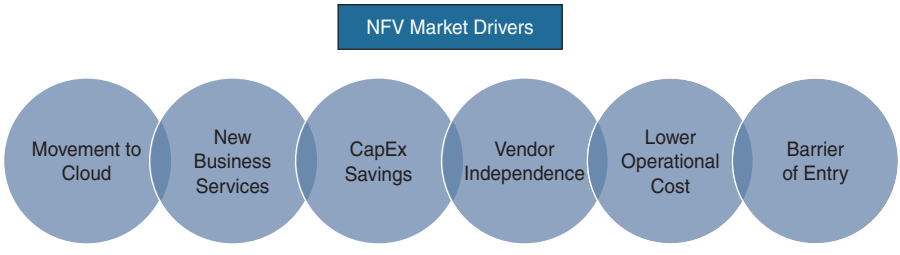

Figure 1-19 *NFV Market Drivers*

Movement to Cloud

With the advent of new smart devices, bandwidth-hungry applications, the new breed of connected devices, and Internet of things (IoT) technologies, the demand for and usage of networks has been increasing exponentially. These recent changes

have created market demand for services being provided anytime, anywhere, and on any device. To meet this market shift, providers are looking to build and offer cloud-based services that can satisfy the new requirements.

Research publications forecast that between 2015 and 2020, the NFV market will grow to beyond $9B with a compound annual growth rate (CAGR) of 83.1% [4]. This is a huge market for the traditional providers to miss, and many new providers are now jumping into this market, such as cloud providers, service providers, enterprise, startups, etc.

New Business Services

Consumption-based growth ensures that the network resources grow in close correlation with demand. With the use of traditional network equipment, network growth occurred in jumps, resulting in first overprovisioned and later underprovisioned network capacities, as depicted in Figure 1-20. The use of NFV avoids wastage of time and resource that would have been spent in continuously reprovisioning and decommissioning the network capacity.

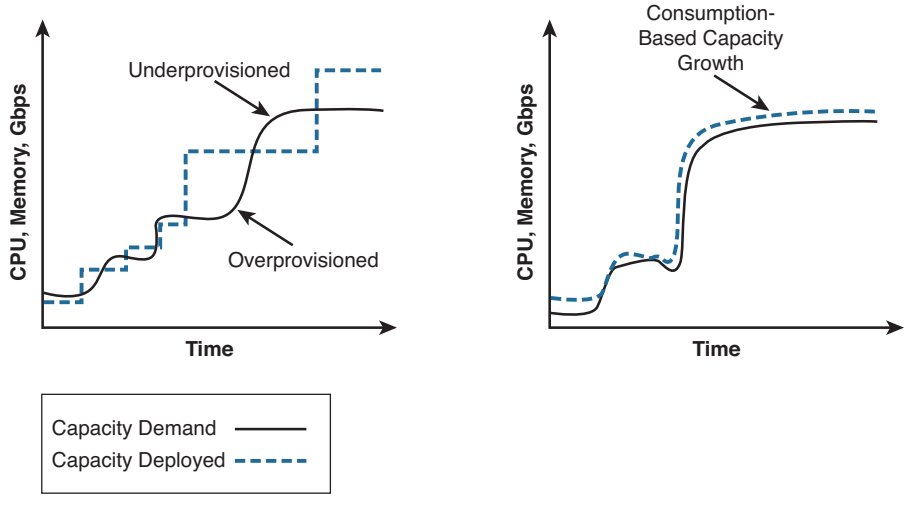

Figure 1-20 *Consumption-Based Capacity Growth*

One of the new business opportunities made possible by NFV is to offer hosting of network and IT services using massive server deployments. This type of service offering has been picking up pace and is proving to be a high-revenue business.

Instead of investing in network and data infrastructure, many enterprises are now opting to lease it as a service through the cloud providers. This is commonly referenced as Infrastructure as a Service (IaaS).

With NFV, providers can offer network services on demand like a shopping-cart experience, allowing consumers to add or delete services and devices through self-managed portals. Since the services will use NFV, these new services can get deployed automatically and become available instantaneously. In this case, if a customer wants to add a new firewall to a branch location, the customer can use such a portal to buy this service with few clicks, and the back-end at the provider will spin off a new virtual machine and deploy the firewall VNF on it as well as connect that VNF to the existing devices for that branch office.

These are just a few examples of NFV's ability to offer new set of business services that can be deployed on-demand and brought up in a short time span. The new breed of services that NFV can make possible are already gaining popularity and creating market enthusiasm. Providers are also offering new business models that utilize consumption-based growth, on-demand deployment, pay-as-you-grow and pay-as-you-use services, and better monetization for the network resources.

Capital Expense Savings

The hardware innovations required in traditional network hardware are costly for vendors to develop and produce. These devices have a small market and are therefore sold in low volume. These two factors are reflected in the cost to the network operators. On top of that, vendors of traditional network equipment have been able to keep the margins high by exploiting the fact that limited alternatives were available to the network operators. NFV revolutionizes this situation by using standard high-volume hardware such as the servers, switches, and storage components.

COTS hardware devices are already mass produced and sold at a reasonable price due to their use in data centers, and the use of off-the-shelf components keeps development costs low and competitive. The low manufacturing cost, combined with economics of scale and efficiency, results in equipment costs that are much less than those of purpose-built hardware.

Operational Expense Savings

With NFV's push for standardized framework, the proprietary aspect of the existing network's vendor-specific hardware and software combination is eliminated or minimized. NFV encourages and supports the use of open standards within its functional blocks as well as their interaction with existing management tools. This makes NFV deploy and operate networks using many existing vendor independent tools from server and data-center space, without new investments.

Virtualized network can share the infrastructure between the network functions as well as applications running on the network, data centers and server farms. The power and space consumed by the infrastructure can therefore be shared and more efficiently used.

Barrier of Entry

With traditional network devices, it's difficult for new vendors or new service providers to enter the market. The development costs for the vendors and the infrastructure costs for the providers present a barrier that is challenging to penetrate. With NFV, which uses open software implementing various network functions and has lower hardware costs, this barrier has been removed. This opens up doors for new vendors and providers to enter the market, bringing innovations and challenging the current vendors by offering lower priced and higher performing implementations of network functions.

Summary

The goal of this chapter is to get the readers acquainted with NFV concepts, standards and benefits. It examines how NFV is transforming the networking industry. It describes how networks evolved from early days of data communication to today's sophisticated networks carrying voice, data, and video traffic. The drawbacks and challenges of traditional network architectures and the ways NFV can help address these issues are discussed. The chapter introduces NFV and examines how it compares with today's networks. It focuses on the importance of understanding the standardization process for NFV. This chapter also provides a detailed study of ETSI's NFV framework. The major advantages of NFV and the market drivers behind it are also covered in this chapter.

References

Please refer to the following for additional information:

[1] http://www.etsi.org/index.php/news-events/news/644-2013-01-isg-nfv-created

[2] http://www.etsi.org/deliver/etsi_gs/NFV-MAN/001_099/001/01.01.01_60/gs_nfv-man001v010101p.pdf

[3] https://portal.etsi.org/NFV/NFV_White_Paper.pdf

[4] http://www.researchandmarkets.com/research/l3cw7s/network_functions

Review Questions

Use the questions here to review what you learned in the chapter. The correct answers are found in Appendix A, "Answers to Review Questions."

1. Which organization is driving the framework for NFV
   a. European Telecommunications Standards Institute (ETSI)
   b. Internet Engineering Task Force (IETF)
   c. International Telecommunication Union (ITU)
   d. Open Network Consortium (ONC)
2. What are the three major blocks of the NFV architecture?
   a. VIM, NFVO, and VNFM
   b. ETSI, MANO, and VNF
   c. NF, NFVI, and MANO
   d. OSS, BSS, and VNF
3. VNFM is responsible for which of the following?
   a. managing the infrastructure hardware and controlling its allocation to the VNFs
   b. managing the lifecycle of the VNF (instantiation, scaling up or down, termination) as well as FCAPS management of the VNF
   c. deploying the end-to-end service in the NFV architecture
   d. gathering the FCAPS information of the physical hardware from VIM and passing them to NFVO so that the resources can be appropriately managed by the upper layers for the ETSI framework
4. Which management functional block facilitates running multiple virtual machines/VNFs on the same hardware?
   a. Virtualized Network Function Manager (VNFM)
   b. Virtualization Infrastructure Manager (VIM)
   c. Element Manager (EM)
   d. Network functions virtualization Orchestrator (NFVO)
5. Communication between different functional blocks such as VIM to VNFM, VNFM to NFVO in the ETSI architecture is called?
   a. communication end points
   b. open network interconnects

c. FCAPS data points

d. reference points

6. List three benefits of NFV compared to traditional network devices:

a. deployment agility

b. hardware-centric

c. elasticity

d. vendor independence

7. The abbreviation "COTS" stands for

a. custom option to service

b. commodity-oriented technical solution

c. commercial off the shelf

d. commercially offered technical solution

Index

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Y