Network Function Virtualization

Internet of Things

Rahul Shandilya

Virtualization

A variety of technologies for managing computer resources by providing an abstraction layer between the software and the physical hardware. These technologies effectively emulate or simulate a hardware platform, such as a server, storage device, or network resource, in software.

- Virtualization turns physical resources into logical, or virtual, resources.

Virtualization

A variety of technologies for managing computer resources by providing an abstraction layer between the software and the physical hardware. These technologies effectively emulate or simulate a hardware platform, such as a server, storage device, or network resource, in software.

- Virtualization turns physical resources into logical, or virtual, resources.
- Virtualization enables users, applications, and management software operating above the abstraction layer to manage and use resources without needing to be aware of the physical details of the underlying resources.

Virtualization

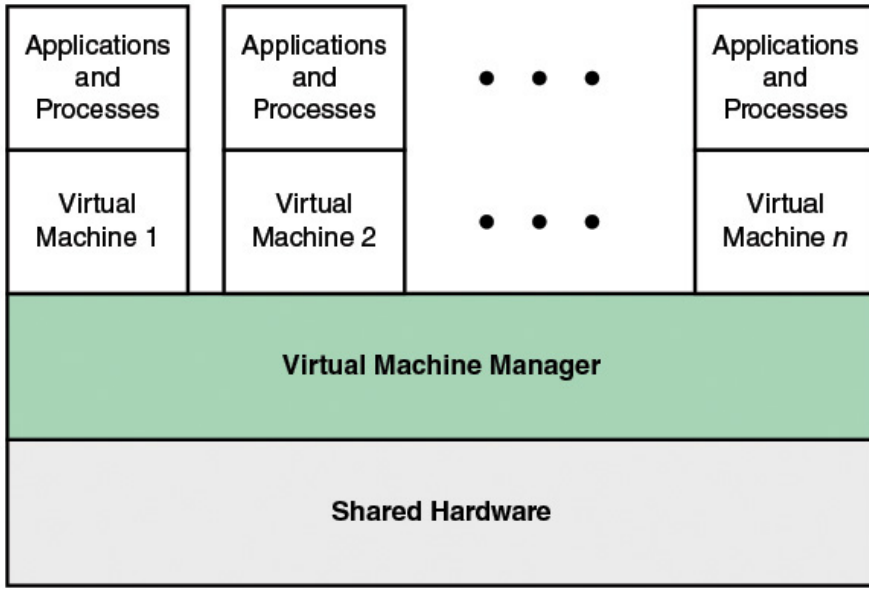
A variety of technologies for managing computer resources by providing an abstraction layer between the software and the physical hardware. These technologies effectively emulate or simulate a hardware platform, such as a server, storage device, or network resource, in software.

- Virtualization turns physical resources into logical, or virtual, resources.
- Virtualization enables users, applications, and management software operating above the abstraction layer to manage and use resources without needing to be aware of the physical details of the underlying resources.
- Virtualization technology enables a single PC or server to simultaneously run multiple operating systems or multiple sessions of a single OS. A machine running virtualization software can host numerous applications, including those that run on different operating systems, on a single hardware platform.

Virtualization

A variety of technologies for managing computer resources by providing an abstraction layer between the software and the physical hardware. These technologies effectively emulate or simulate a hardware platform, such as a server, storage device, or network resource, in software.

- Virtualization turns physical resources into logical, or virtual, resources.
- Virtualization enables users, applications, and management software operating above the abstraction layer to manage and use resources without needing to be aware of the physical details of the underlying resources.
- Virtualization technology enables a single PC or server to simultaneously run multiple operating systems or multiple sessions of a single OS. A machine running virtualization software can host numerous applications, including those that run on different operating systems, on a single hardware platform.
- The solution that enables virtualization is a *virtual machine monitor* (VMM), or commonly known today as a *hypervisor*. This software sits between the hardware and the VMs acting as a resource broker.


Applications and Processes
Applications and Processes
Applications and Processes
Virtual Machine 1
Virtual Machine 2
Virtual Machine n
Virtual Machine Manager
Shared Hardware

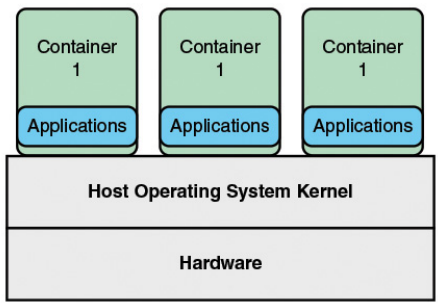
Container Virtualization

In this approach, software, known as a virtualization container, runs on top of the host OS kernel and provides an execution environment for applications.

Container Virtualization

In this approach, software, known as a virtualization container, runs on top of the host OS kernel and provides an execution environment for applications.

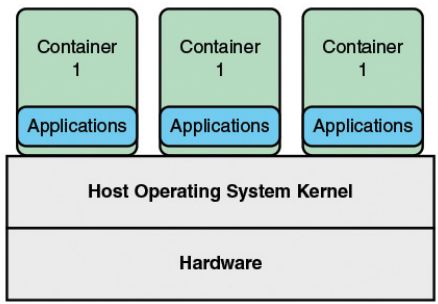
- Unlike hypervisor-based VMs, containers do not aim to emulate physical servers. Instead, all containerized applications on a host share a common OS kernel.



Container Virtualization

In this approach, software, known as a virtualization container, runs on top of the host OS kernel and provides an execution environment for applications.

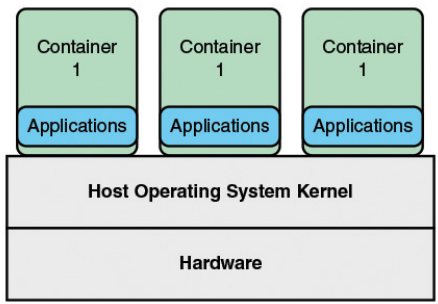
- Unlike hypervisor-based VMs, containers do not aim to emulate physical servers. Instead, all containerized applications on a host share a common OS kernel.
- This eliminates the resources needed to run a separate OS for each application and can greatly reduce overhead.



Container Virtualization

In this approach, software, known as a virtualization container, runs on top of the host OS kernel and provides an execution environment for applications.

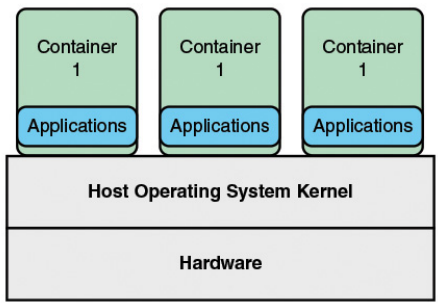
- Unlike hypervisor-based VMs, containers do not aim to emulate physical servers. Instead, all containerized applications on a host share a common OS kernel.
- This eliminates the resources needed to run a separate OS for each application and can greatly reduce overhead.
- Because the containers execute on the same kernel, thus sharing most of the base OS, containers are much smaller and lighter weight compared to a hypervisor/guest OS VM arrangement.



Container Virtualization

In this approach, software, known as a virtualization container, runs on top of the host OS kernel and provides an execution environment for applications.

- Unlike hypervisor-based VMs, containers do not aim to emulate physical servers. Instead, all containerized applications on a host share a common OS kernel.
- This eliminates the resources needed to run a separate OS for each application and can greatly reduce overhead.
- Because the containers execute on the same kernel, thus sharing most of the base OS, containers are much smaller and lighter weight compared to a hypervisor/guest OS VM arrangement.
- An OS can have many containers running on top of it, compared to the limited number of hypervisors and guest operating systems that can be supported.



Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.

Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.
- New hardware means additional capital expenditures.

Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.
- New hardware means additional capital expenditures.
- Once new types of hardware appliances are acquired, operators are faced with the rarity of skills necessary to design, integrate, and operate increasingly complex hardware-based appliances.

Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.
- New hardware means additional capital expenditures.
- Once new types of hardware appliances are acquired, operators are faced with the rarity of skills necessary to design, integrate, and operate increasingly complex hardware-based appliances.
- Hardware-based appliances rapidly reach end of life, requiring much of the procure-design-integrate-deploy cycle to be repeated with little or no revenue benefit.

Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.
- New hardware means additional capital expenditures.
- Once new types of hardware appliances are acquired, operators are faced with the rarity of skills necessary to design, integrate, and operate increasingly complex hardware-based appliances.
- Hardware-based appliances rapidly reach end of life, requiring much of the procure-design-integrate-deploy cycle to be repeated with little or no revenue benefit.
- As technology and services innovation accelerates to meet the demands of an increasingly network-centric IT environment, the need for an increasing variety of hardware platforms inhibits the introduction of new revenue-earning network services.

Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.
- New hardware means additional capital expenditures.
- Once new types of hardware appliances are acquired, operators are faced with the rarity of skills necessary to design, integrate, and operate increasingly complex hardware-based appliances.
- Hardware-based appliances rapidly reach end of life, requiring much of the procure-design-integrate-deploy cycle to be repeated with little or no revenue benefit.
- As technology and services innovation accelerates to meet the demands of an increasingly network-centric IT environment, the need for an increasing variety of hardware platforms inhibits the introduction of new revenue-earning network services.

Motivation for NFV

- New network services may require additional different types of hardware appliances, and finding the space and power to accommodate these boxes is becoming increasingly difficult.
- New hardware means additional capital expenditures.
- Once new types of hardware appliances are acquired, operators are faced with the rarity of skills necessary to design, integrate, and operate increasingly complex hardware-based appliances.
- Hardware-based appliances rapidly reach end of life, requiring much of the procure-design-integrate-deploy cycle to be repeated with little or no revenue benefit.
- As technology and services innovation accelerates to meet the demands of an increasingly network-centric IT environment, the need for an increasing variety of hardware platforms inhibits the introduction of new revenue-earning network services.

The NFV approach moves away from dependence on a variety of hardware platforms to the use of a small number of standardized platform types, with virtualization techniques used to provide the needed network functionality.

NFV Concepts

NFV is a significant departure from traditional approaches to the design, deployment, and management of networking services. NFV decouples network functions, such as Network Address Translation (NAT), firewalling, intrusion detection, Domain Name Service (DNS), and caching, from proprietary hardware appliances so that they can run in software on VMs.

NFV Concepts

NFV is a significant departure from traditional approaches to the design, deployment, and management of networking services. NFV decouples network functions, such as Network Address Translation (NAT), firewalling, intrusion detection, Domain Name Service (DNS), and caching, from proprietary hardware appliances so that they can run in software on VMs.
The various Network-based devices include:

- **Network function devices:** Such as switches, routers, network access points, customer premises equipment (CPE), and deep packet inspectors (for deep packet inspection).

NFV Concepts

NFV is a significant departure from traditional approaches to the design, deployment, and management of networking services. NFV decouples network functions, such as Network Address Translation (NAT), firewalling, intrusion detection, Domain Name Service (DNS), and caching, from proprietary hardware appliances so that they can run in software on VMs.
The various Network-based devices include:

- **Network function devices:** Such as switches, routers, network access points, customer premises equipment (CPE), and deep packet inspectors (for deep packet inspection).
- **Network-related compute devices:** Such as firewalls, intrusion detection systems, and network management systems.

NFV Concepts

NFV is a significant departure from traditional approaches to the design, deployment, and management of networking services. NFV decouples network functions, such as Network Address Translation (NAT), firewalling, intrusion detection, Domain Name Service (DNS), and caching, from proprietary hardware appliances so that they can run in software on VMs.
The various Network-based devices include:

- **Network function devices:** Such as switches, routers, network access points, customer premises equipment (CPE), and deep packet inspectors (for deep packet inspection).
- **Network-related compute devices:** Such as firewalls, intrusion detection systems, and network management systems.
- **Network-attached storage**: File and database servers attached to the network.

NFV Concepts

NFV is a significant departure from traditional approaches to the design, deployment, and management of networking services. NFV decouples network functions, such as Network Address Translation (NAT), firewalling, intrusion detection, Domain Name Service (DNS), and caching, from proprietary hardware appliances so that they can run in software on VMs.
The various Network-based devices include:

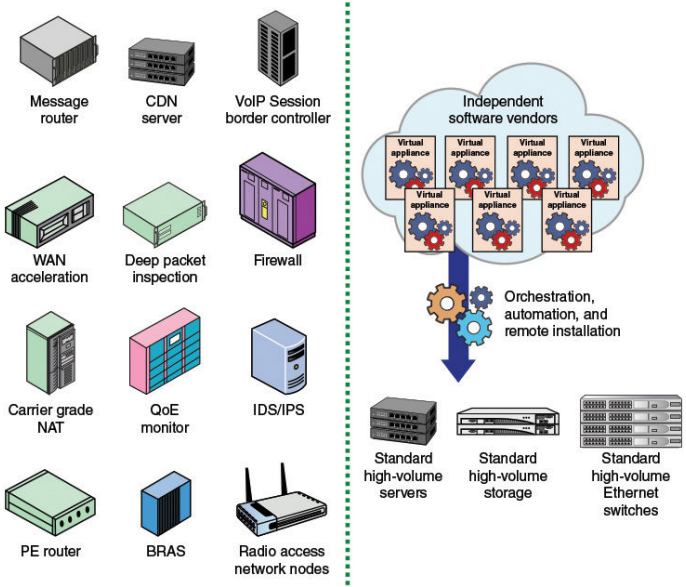
- **Network function devices:** Such as switches, routers, network access points, customer premises equipment (CPE), and deep packet inspectors (for deep packet inspection).
- **Network-related compute devices:** Such as firewalls, intrusion detection systems, and network management systems.
- **Network-attached storage**: File and database servers attached to the network.

NFV Concepts

NFV is a significant departure from traditional approaches to the design, deployment, and management of networking services. NFV decouples network functions, such as Network Address Translation (NAT), firewalling, intrusion detection, Domain Name Service (DNS), and caching, from proprietary hardware appliances so that they can run in software on VMs.
The various Network-based devices include:

- **Network function devices:** Such as switches, routers, network access points, customer premises equipment (CPE), and deep packet inspectors (for deep packet inspection).
- **Network-related compute devices:** Such as firewalls, intrusion detection systems, and network management systems.
- **Network-attached storage**: File and database servers attached to the network.

In traditional networks, all devices are deployed on proprietary/closed platforms. All network elements are enclosed boxes, and hardware cannot be shared.With NFV, however, network elements are independent applications that are flexibly deployed on a unified platform comprising standard servers, storage devices, and switches. In this way, software and hardware are decoupled, and capacity for each application is increased or decreased by adding or reducing virtual resources



CDN = content delivery network
WAN = wide area network
NAT = network address translation
QoE = quality of experience
VoIP = voice over Internet Protocol
IDS = intrusion detection system
IPS = intrusion prevention system
PE = provider edge router
BRAS = broadband remote access server

NFV Terminology

- **Network Function:** A functional block within a network infrastructure that has well-defined external interfaces and well-defined functional behavior. Typically, this is a physical network node or other physical appliance.

NFV Terminology

- **Network Function:** A functional block within a network infrastructure that has well-defined external interfaces and well-defined functional behavior. Typically, this is a physical network node or other physical appliance.
- **Network Services:** A composition of network functions that is defined by its functional and behavioral specification.

NFV Terminology

- **Network Function:** A functional block within a network infrastructure that has well-defined external interfaces and well-defined functional behavior. Typically, this is a physical network node or other physical appliance.
- **Network Services:** A composition of network functions that is defined by its functional and behavioral specification.
- **Network Function Virtualization (NFV):** The principle of separating network functions from the hard- ware they run on by using virtual hardware abstraction.

NFV Terminology

- **Network Function:** A functional block within a network infrastructure that has well-defined external interfaces and well-defined functional behavior. Typically, this is a physical network node or other physical appliance.
- **Network Services:** A composition of network functions that is defined by its functional and behavioral specification.
- **Network Function Virtualization (NFV):** The principle of separating network functions from the hard- ware they run on by using virtual hardware abstraction.
- **Network Function Virtualization Infrastructure (NFVI):** The totality of all hardware and software components that build up the environment in which virtual network functions (VNFs) are deployed. The NFVI can span across several locations (that is, multiple points of presence [N-PoPs). The network providing connectivity between these locations is considered to be part of the NFVI.

NFV Terminology

- **Network Function:** A functional block within a network infrastructure that has well-defined external interfaces and well-defined functional behavior. Typically, this is a physical network node or other physical appliance.
- **Network Services:** A composition of network functions that is defined by its functional and behavioral specification.
- **Network Function Virtualization (NFV):** The principle of separating network functions from the hard- ware they run on by using virtual hardware abstraction.
- **Network Function Virtualization Infrastructure (NFVI):** The totality of all hardware and software components that build up the environment in which virtual network functions (VNFs) are deployed. The NFVI can span across several locations (that is, multiple points of presence [N-PoPs). The network providing connectivity between these locations is considered to be part of the NFVI.
- **NFVI-Node:** Physical devices deployed and managed as a single entity, providing the NFVI functions required to support the execu- tion environment for VNFs.

NFV Terminology

- **Physical Network Function (PNF):** An implementation of a NF via a tightly coupled software and hardware system. This is typically a proprietary system.

NFV Terminology

- **Physical Network Function (PNF):** An implementation of a NF via a tightly coupled software and hardware system. This is typically a proprietary system.
- **Virtual Network:** A topological component used to affect routing of specific characteristic information. The virtual network is bounded by its set of permissible network interfaces. In the NFVI architecture, a virtual network routes information among the network interfaces of VM instances and physical network interfaces, providing the necessary connectivity.

NFV Terminology

- **Physical Network Function (PNF):** An implementation of a NF via a tightly coupled software and hardware system. This is typically a proprietary system.
- **Virtual Network:** A topological component used to affect routing of specific characteristic information. The virtual network is bounded by its set of permissible network interfaces. In the NFVI architecture, a virtual network routes information among the network interfaces of VM instances and physical network interfaces, providing the necessary connectivity.
- **Virtual Network Function (VNF):** An implementation of an NF that can be deployed on an NFVI.

NFV Terminology

- **Physical Network Function (PNF):** An implementation of a NF via a tightly coupled software and hardware system. This is typically a proprietary system.
- **Virtual Network:** A topological component used to affect routing of specific characteristic information. The virtual network is bounded by its set of permissible network interfaces. In the NFVI architecture, a virtual network routes information among the network interfaces of VM instances and physical network interfaces, providing the necessary connectivity.
- **Virtual Network Function (VNF):** An implementation of an NF that can be deployed on an NFVI.
- **NFVI-POP:** An N-PoP where a network function is or could be deployed as a VNF.

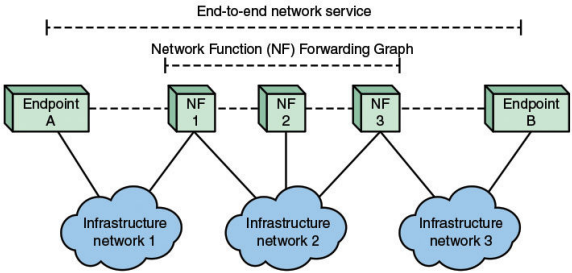
NFV Terminology

- **Physical Network Function (PNF):** An implementation of a NF via a tightly coupled software and hardware system. This is typically a proprietary system.
- **Virtual Network:** A topological component used to affect routing of specific characteristic information. The virtual network is bounded by its set of permissible network interfaces. In the NFVI architecture, a virtual network routes information among the network interfaces of VM instances and physical network interfaces, providing the necessary connectivity.
- **Virtual Network Function (VNF):** An implementation of an NF that can be deployed on an NFVI.
- **NFVI-POP:** An N-PoP where a network function is or could be deployed as a VNF.
- **Network Forwarding Path:** Ordered list of connection points forming a chain of NFs, along with policies associated with the list.

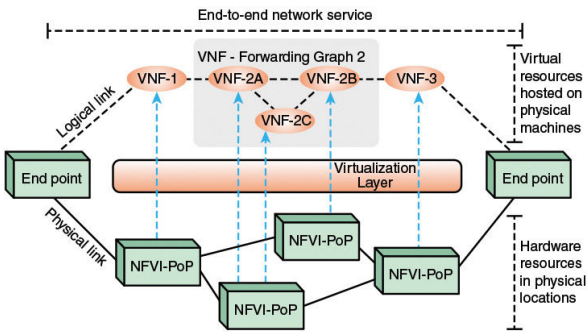
NFV Terminology

- **Physical Network Function (PNF):** An implementation of a NF via a tightly coupled software and hardware system. This is typically a proprietary system.
- **Virtual Network:** A topological component used to affect routing of specific characteristic information. The virtual network is bounded by its set of permissible network interfaces. In the NFVI architecture, a virtual network routes information among the network interfaces of VM instances and physical network interfaces, providing the necessary connectivity.
- **Virtual Network Function (VNF):** An implementation of an NF that can be deployed on an NFVI.
- **NFVI-POP:** An N-PoP where a network function is or could be deployed as a VNF.
- **Network Forwarding Path:** Ordered list of connection points forming a chain of NFs, along with policies associated with the list.
- **VNF Forwarding Path:** Graph of logical links connecting VNF nodes for the pur- pose of describing traffic flow between these network func- tions.

NFV Example



(a) Graph representation of an end-to-end network service



(b) Example of an end-to-end network service with VNFs and nested forwarding graphs

NFV Principles

the VNFs are the building blocks used to create end-to-end network services. Three key NFV principles are involved in creating practical network services:

NFV Principles

the VNFs are the building blocks used to create end-to-end network services. Three key NFV principles are involved in creating practical network services:

- **Service chaining**: VNFs are modular and each VNF provides limited functionality on its own. For a given traffic flow within a given application, the service provider steers the flow through multiple VNFs to achieve the desired network functionality. This is referred to as service chaining.

NFV Principles

the VNFs are the building blocks used to create end-to-end network services. Three key NFV principles are involved in creating practical network services:

- **Service chaining**: VNFs are modular and each VNF provides limited functionality on its own. For a given traffic flow within a given application, the service provider steers the flow through multiple VNFs to achieve the desired network functionality. This is referred to as service chaining.
- **Management and orchestration (MANO)**: This involves deploying and managing the lifecycle of VNF instances. Examples include VNF instance creation, VNF service chaining, monitoring, relocation, shutdown, and billing. MANO also manages the NFV infrastructure elements.

NFV Principles

the VNFs are the building blocks used to create end-to-end network services. Three key NFV principles are involved in creating practical network services:

- **Service chaining**: VNFs are modular and each VNF provides limited functionality on its own. For a given traffic flow within a given application, the service provider steers the flow through multiple VNFs to achieve the desired network functionality. This is referred to as service chaining.
- **Management and orchestration (MANO)**: This involves deploying and managing the lifecycle of VNF instances. Examples include VNF instance creation, VNF service chaining, monitoring, relocation, shutdown, and billing. MANO also manages the NFV infrastructure elements.
- **Distributed architecture**: A VNF may be made up of one or more VNF components (VNFC), each of which implements a subset of the VNF's functionality. Each VNFC may be deployed in one or multiple instances. These instances may be deployed on separate, distributed hosts to provide scalability and redundancy.

High-Level NFV Framework

This framework supports the implementation of network functions as software-only VNFs. The NFV framework consists of three domains of operation:

High-Level NFV Framework

This framework supports the implementation of network functions as software-only VNFs. The NFV framework consists of three domains of operation:

- **Virtualized network functions**: The collection of VNFs, implemented in software, that run over the NFVI.

High-Level NFV Framework

This framework supports the implementation of network functions as software-only VNFs. The NFV framework consists of three domains of operation:

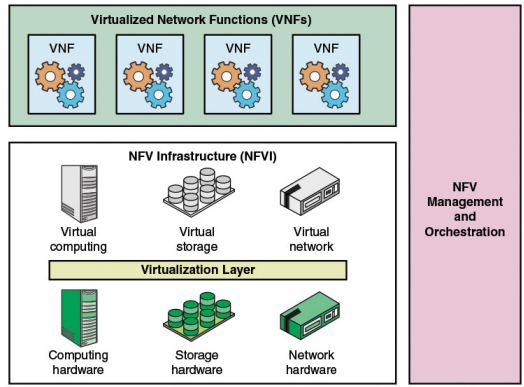
- **Virtualized network functions**: The collection of VNFs, implemented in software, that run over the NFVI.
- **NFV infrastructure (NFVI)**: The NFVI performs a virtualization function on the three main categories of devices in the network service environment: computer devices, storage devices, and network devices.

High-Level NFV Framework

This framework supports the implementation of network functions as software-only VNFs. The NFV framework consists of three domains of operation:

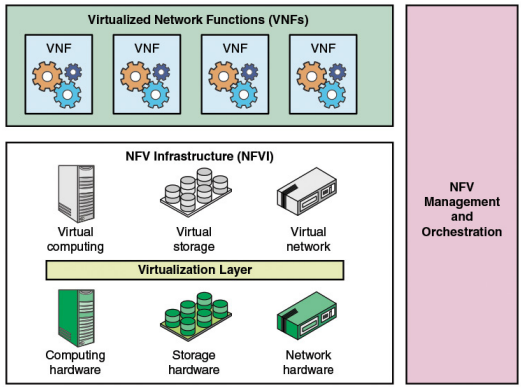
- **Virtualized network functions**: The collection of VNFs, implemented in software, that run over the NFVI.
- **NFV infrastructure (NFVI)**: The NFVI performs a virtualization function on the three main categories of devices in the network service environment: computer devices, storage devices, and network devices.
- **NFV management and orchestration**: Encompasses the orchestration and lifecycle management of physical/software resources that support the infrastructure virtualization, and the lifecycle management of VNFs. NFV management and orchestration focuses on all virtualization-specific management tasks necessary in the NFV framework.

In the deployment, operation, management and orchestration of VNFs, two types of relations between VNFs are supported:



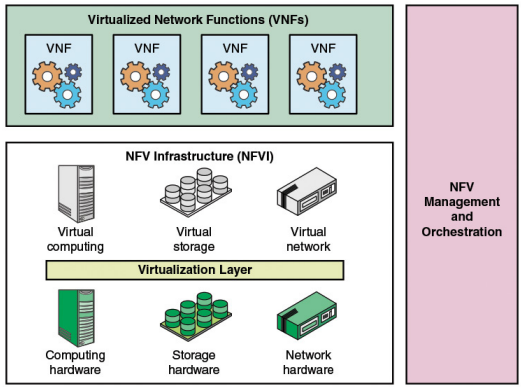
In the deployment, operation, management and orchestration of VNFs, two types of relations between VNFs are supported:

- **VNF forwarding graph (VNF FG)**: Covers the case where network connectivity between VNFs is specified, such as a chain of VNFs on the path to a web server tier (for example, firewall, network address translator, load balancer).



In the deployment, operation, management and orchestration of VNFs, two types of relations between VNFs are supported:

- **VNF forwarding graph (VNF FG)**: Covers the case where network connectivity between VNFs is specified, such as a chain of VNFs on the path to a web server tier (for example, firewall, network address translator, load balancer).
- **VNF set**: Covers the case where the connectivity between VNFs is not specified, such as a web server pool.



NFV Reference Architecture

Important component of ISG NFV reference architectural framework:

- **NFV infrastructure (NFVI)**: Comprises the hardware and software resources that create the environment in which VNFs are deployed. NFVI virtualizes physical computing, storage, and networking and places them into resource pools.

NFV Reference Architecture

Important component of ISG NFV reference architectural framework:

- **NFV infrastructure (NFVI)**: Comprises the hardware and software resources that create the environment in which VNFs are deployed. NFVI virtualizes physical computing, storage, and networking and places them into resource pools.
- **VNF/EMS**: The collection of VNFs implemented in software to run on virtual computing, storage, and networking resources, together with a collection of element management systems (EMS) that manage the VNFs.

NFV Reference Architecture

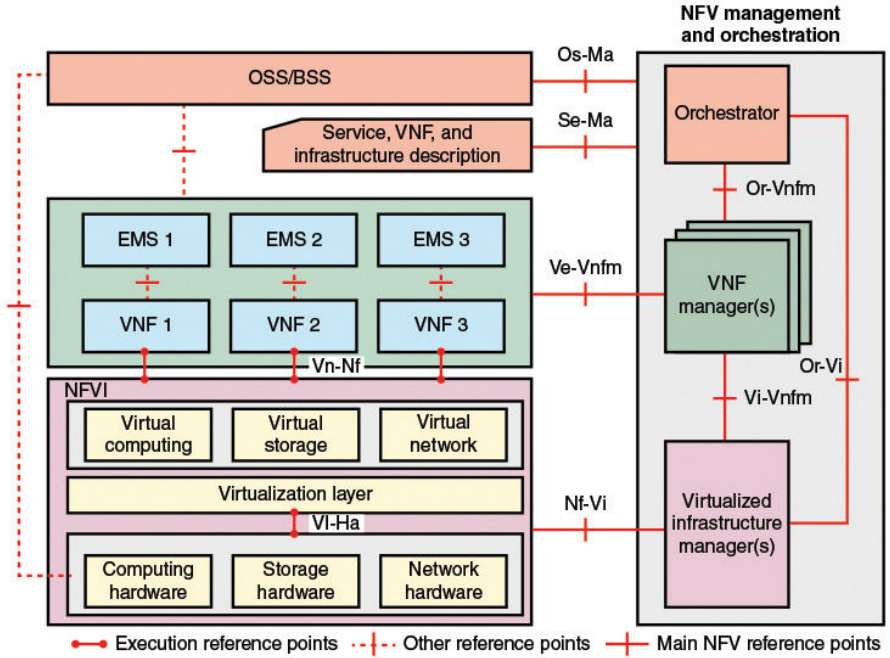
Important component of ISG NFV reference architectural framework:

- **NFV infrastructure (NFVI)**: Comprises the hardware and software resources that create the environment in which VNFs are deployed. NFVI virtualizes physical computing, storage, and networking and places them into resource pools.
- **VNF/EMS**: The collection of VNFs implemented in software to run on virtual computing, storage, and networking resources, together with a collection of element management systems (EMS) that manage the VNFs.
- **NFV management and orchestration (NFV-MANO)**: Framework for the management and orchestration of all resources in the NFV environment. This includes computing, networking, storage, and VM resources.

NFV Reference Architecture

Important component of ISG NFV reference architectural framework:

- **NFV infrastructure (NFVI)**: Comprises the hardware and software resources that create the environment in which VNFs are deployed. NFVI virtualizes physical computing, storage, and networking and places them into resource pools.
- **VNF/EMS**: The collection of VNFs implemented in software to run on virtual computing, storage, and networking resources, together with a collection of element management systems (EMS) that manage the VNFs.
- **NFV management and orchestration (NFV-MANO)**: Framework for the management and orchestration of all resources in the NFV environment. This includes computing, networking, storage, and VM resources.
- **OSS/BSS**: Operational and business support systems implemented by the VNF service provider.


NFV management and orchestration
OSS/BSS
Os-Ma
Service, VNF, and infrastructure description
Se-Ma
Orchestrator
Or-Vnfm
EMS 1
EMS 2
EMS 3
VNF 1
VNF 2
VNF 3
Ve-Vnfm
VNF manager(s)
Vn-Nf
Or-Vi
Vi-Vnfm
NFVI
Virtual computing
Virtual storage
Virtual network
Virtualization layer
Vl-Ha
Nf-Vi
Virtualized infrastructure manager(s)
Computing hardware
Storage hardware
Network hardware
Execution reference points
Other reference points
Main NFV reference points

NFV Management and Orchestration

The NFV management and orchestration facility includes the following functional blocks:

- **NFV orchestrator**: Responsible for installing and configuring new network services (NS) and virtual network function (VNF) packages, NS lifecycle management, global resource management, and validation and authorization of NFVI resource requests.

NFV Management and Orchestration

The NFV management and orchestration facility includes the following functional blocks:

- **NFV orchestrator**: Responsible for installing and configuring new network services (NS) and virtual network function (VNF) packages, NS lifecycle management, global resource management, and validation and authorization of NFVI resource requests.
- **VNF manager**: Oversees lifecycle management of VNF instances.

NFV Management and Orchestration

The NFV management and orchestration facility includes the following functional blocks:

- **NFV orchestrator**: Responsible for installing and configuring new network services (NS) and virtual network function (VNF) packages, NS lifecycle management, global resource management, and validation and authorization of NFVI resource requests.
- **VNF manager**: Oversees lifecycle management of VNF instances.
- **Virtualized infrastructure manager**: Controls and manages the interaction of a VNF with computing, storage, and network resources under its authority, in addition to their virtualization.

NFV Management and Orchestration

The NFV management and orchestration facility includes the following functional blocks:

- **NFV orchestrator**: Responsible for installing and configuring new network services (NS) and virtual network function (VNF) packages, NS lifecycle management, global resource management, and validation and authorization of NFVI resource requests.
- **VNF manager**: Oversees lifecycle management of VNF instances.
- **Virtualized infrastructure manager**: Controls and manages the interaction of a VNF with computing, storage, and network resources under its authority, in addition to their virtualization.

NFV Management and Orchestration

The NFV management and orchestration facility includes the following functional blocks:

- **NFV orchestrator**: Responsible for installing and configuring new network services (NS) and virtual network function (VNF) packages, NS lifecycle management, global resource management, and validation and authorization of NFVI resource requests.
- **VNF manager**: Oversees lifecycle management of VNF instances.
- **Virtualized infrastructure manager**: Controls and manages the interaction of a VNF with computing, storage, and network resources under its authority, in addition to their virtualization.

Reference Points

Architecture defines a number of reference points that constitute interfaces between functional blocks.

- **Vi-Ha**: Marks interfaces to the physical hardware. A well-defined interface specification will facilitate for operators sharing physical resources for different purposes, reassigning resources for different purposes, evolving software and hardware independently, and obtaining software and hardware component from different vendors.

- **Vn-Nf**: These interfaces are APIs used by VNFs to execute on the virtual infrastructure. Application developers, whether migrating existing network functions or developing new VNFs, require a consistent interface the provides functionality and the ability to specify performance, reliability, and scalability requirements.

- **Vn-Nf**: These interfaces are APIs used by VNFs to execute on the virtual infrastructure. Application developers, whether migrating existing network functions or developing new VNFs, require a consistent interface the provides functionality and the ability to specify performance, reliability, and scalability requirements.
- **Nf-Vi**: Marks interfaces between the NFVI and the virtualized infrastructure manager (VIM). This interface can facilitate specification of the capabilities that the NFVI provides for the VIM. The VIM must be able to manage all the NFVI virtual resources, including allocation, monitoring of system utilization, and fault management.

- **Vn-Nf**: These interfaces are APIs used by VNFs to execute on the virtual infrastructure. Application developers, whether migrating existing network functions or developing new VNFs, require a consistent interface the provides functionality and the ability to specify performance, reliability, and scalability requirements.
- **Nf-Vi**: Marks interfaces between the NFVI and the virtualized infrastructure manager (VIM). This interface can facilitate specification of the capabilities that the NFVI provides for the VIM. The VIM must be able to manage all the NFVI virtual resources, including allocation, monitoring of system utilization, and fault management.
- **Or-Vnfm**: This reference point is used for sending configuration information to the VNF manager and collecting state information of the VNFs necessary for network service lifecycle management.

- **Vn-Nf**: These interfaces are APIs used by VNFs to execute on the virtual infrastructure. Application developers, whether migrating existing network functions or developing new VNFs, require a consistent interface the provides functionality and the ability to specify performance, reliability, and scalability requirements.
- **Nf-Vi**: Marks interfaces between the NFVI and the virtualized infrastructure manager (VIM). This interface can facilitate specification of the capabilities that the NFVI provides for the VIM. The VIM must be able to manage all the NFVI virtual resources, including allocation, monitoring of system utilization, and fault management.
- **Or-Vnfm**: This reference point is used for sending configuration information to the VNF manager and collecting state information of the VNFs necessary for network service lifecycle management.
- **Vi-Vnfm**: Used for resource allocation requests by the VNF manager and the exchange of resource configuration and state information.

- **Vn-Nf**: These interfaces are APIs used by VNFs to execute on the virtual infrastructure. Application developers, whether migrating existing network functions or developing new VNFs, require a consistent interface the provides functionality and the ability to specify performance, reliability, and scalability requirements.
- **Nf-Vi**: Marks interfaces between the NFVI and the virtualized infrastructure manager (VIM). This interface can facilitate specification of the capabilities that the NFVI provides for the VIM. The VIM must be able to manage all the NFVI virtual resources, including allocation, monitoring of system utilization, and fault management.
- **Or-Vnfm**: This reference point is used for sending configuration information to the VNF manager and collecting state information of the VNFs necessary for network service lifecycle management.
- **Vi-Vnfm**: Used for resource allocation requests by the VNF manager and the exchange of resource configuration and state information.
- **Or-Vi**: Used for resource allocation requests by the NFV orchestrator and the exchange of resource configuration and state information.

NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.

NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.
- Reduced OpEx, in terms of power consumption and space usage, by using commodity servers and switches, consolidating equipment, and exploiting economies of scale, and reduced network management and control expenses.

NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.
- Reduced OpEx, in terms of power consumption and space usage, by using commodity servers and switches, consolidating equipment, and exploiting economies of scale, and reduced network management and control expenses.
- The ability to innovate and roll out services quickly and also lowers the risks associated with rolling out new services, allowing providers to easily trial and evolve services to determine what best meets the needs of customers.

NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.
- Reduced OpEx, in terms of power consumption and space usage, by using commodity servers and switches, consolidating equipment, and exploiting economies of scale, and reduced network management and control expenses.
- The ability to innovate and roll out services quickly and also lowers the risks associated with rolling out new services, allowing providers to easily trial and evolve services to determine what best meets the needs of customers.
- Ease of interoperability because of standardized and open interfaces.

NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.
- Reduced OpEx, in terms of power consumption and space usage, by using commodity servers and switches, consolidating equipment, and exploiting economies of scale, and reduced network management and control expenses.
- The ability to innovate and roll out services quickly and also lowers the risks associated with rolling out new services, allowing providers to easily trial and evolve services to determine what best meets the needs of customers.
- Ease of interoperability because of standardized and open interfaces.
- Use of a single platform for different applications, users and tenants. This allows network operators to share resources across services and across different customer bases.

NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.
- Reduced OpEx, in terms of power consumption and space usage, by using commodity servers and switches, consolidating equipment, and exploiting economies of scale, and reduced network management and control expenses.
- The ability to innovate and roll out services quickly and also lowers the risks associated with rolling out new services, allowing providers to easily trial and evolve services to determine what best meets the needs of customers.
- Ease of interoperability because of standardized and open interfaces.
- Use of a single platform for different applications, users and tenants. This allows network operators to share resources across services and across different customer bases.
- Provided agility and flexibility, by quickly scaling up or down services to address changing demands.

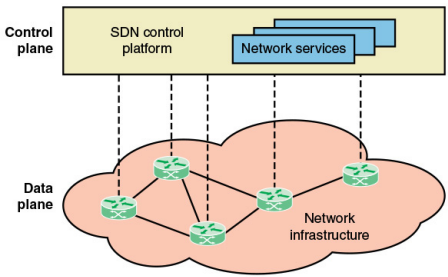
NFV Advantages

If NFV is implemented efficiently and effectively, it can provide a number of benefits compared to traditional networking approaches.

- Reduced CapEx, by using commodity servers and switches, consolidating equipment, exploiting economies of scale, and supporting pay-as-you grow models to eliminate wasteful overprovisioning.
- Reduced OpEx, in terms of power consumption and space usage, by using commodity servers and switches, consolidating equipment, and exploiting economies of scale, and reduced network management and control expenses.
- The ability to innovate and roll out services quickly and also lowers the risks associated with rolling out new services, allowing providers to easily trial and evolve services to determine what best meets the needs of customers.
- Ease of interoperability because of standardized and open interfaces.
- Use of a single platform for different applications, users and tenants. This allows network operators to share resources across services and across different customer bases.
- Provided agility and flexibility, by quickly scaling up or down services to address changing demands.
- Targeted service introduction based on geography or customer sets is possible. Services can be rapidly scaled up/down as required.

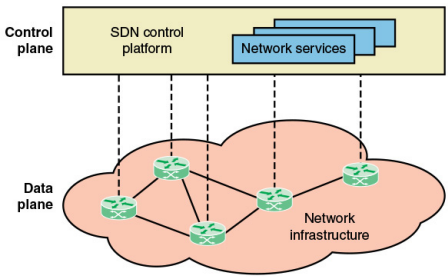
SDN and NFV Differences and Similarities

SDN

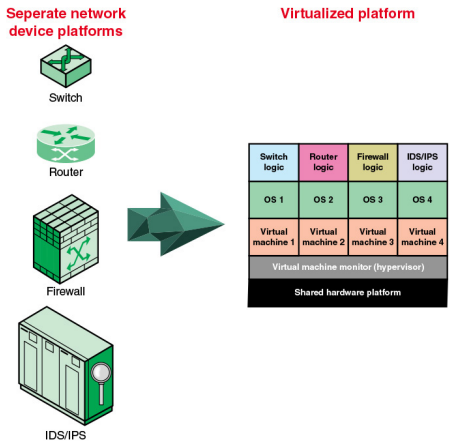


SDN and NFV Differences and Similarities

SDN



NFV



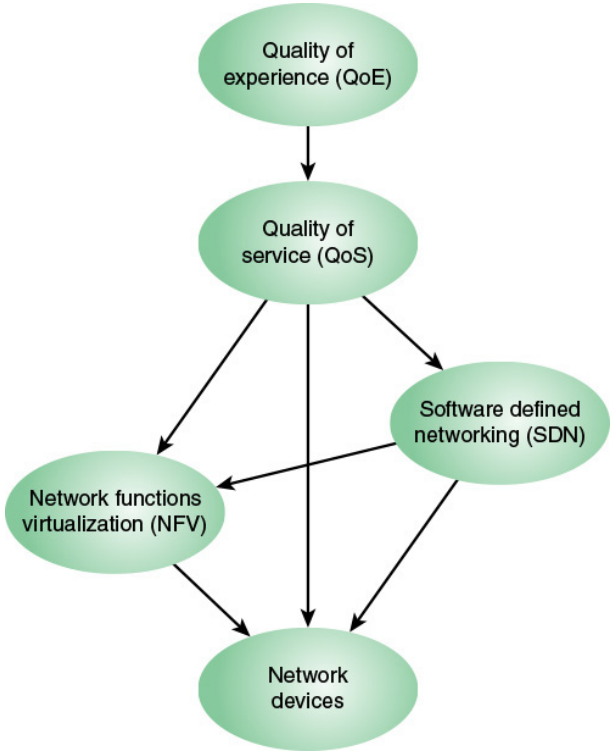
- The core similarity between software-defined networking (SDN) and network functions virtualization (NFV) is that they both use network abstraction.

- The core similarity between software-defined networking (SDN) and network functions virtualization (NFV) is that they both use network abstraction.
- Both depend heavily on virtualization to enable network design and infrastructure to be abstracted in software and then implemented by underlying software across hardware platforms and devices.

- The core similarity between software-defined networking (SDN) and network functions virtualization (NFV) is that they both use network abstraction.
- Both depend heavily on virtualization to enable network design and infrastructure to be abstracted in software and then implemented by underlying software across hardware platforms and devices.
- SDN seeks to separate network control functions from network forwarding functions, while NFV seeks to abstract network forwarding and other networking functions from the hardware on which it runs.

- The core similarity between software-defined networking (SDN) and network functions virtualization (NFV) is that they both use network abstraction.
- Both depend heavily on virtualization to enable network design and infrastructure to be abstracted in software and then implemented by underlying software across hardware platforms and devices.
- SDN seeks to separate network control functions from network forwarding functions, while NFV seeks to abstract network forwarding and other networking functions from the hardware on which it runs.
- SDN abstracts physical networking resources –switches, routers and so on – and moves decision making to a virtual network control plane. In this approach, the control plane decides where to send traffic, while the hardware continues to direct and handle the traffic. NFV aims to virtualize all physical network resources beneath a hypervisor, which allows the network to grow without the addition of more devices.

- The core similarity between software-defined networking (SDN) and network functions virtualization (NFV) is that they both use network abstraction.
- Both depend heavily on virtualization to enable network design and infrastructure to be abstracted in software and then implemented by underlying software across hardware platforms and devices.
- SDN seeks to separate network control functions from network forwarding functions, while NFV seeks to abstract network forwarding and other networking functions from the hardware on which it runs.
- SDN abstracts physical networking resources –switches, routers and so on – and moves decision making to a virtual network control plane. In this approach, the control plane decides where to send traffic, while the hardware continues to direct and handle the traffic. NFV aims to virtualize all physical network resources beneath a hypervisor, which allows the network to grow without the addition of more devices.
- When SDN executes on an NFV infrastructure, SDN forwards data packets from one network device to another. At the same time, SDN's networking control functions for routing, policy definition and applications run in a virtual machine somewhere on the network. Thus, NFV provides basic networking functions, while SDN controls and orchestrates them for specific uses. SDN further allows configuration and behavior to be programmatically defined and modified.


Quality of experience (QoE)
Quality of service (QoS)
Software defined networking (SDN)
Network functions virtualization (NFV)
Network devices

SDN and NFV

The concern of a network service provider is about the set of network devices (such as routers) and the control and management of the functions they perform (such as packet forwarding). Both SDF and NFV can be used seperately to provide this but SDN and NFV are not mutually exclusive. If both SDN and NFV are implemented for a network, the following relationships hold:

- The relationship between SDN and NFV is perhaps viewed as SDN functioning as an enabler of NFV.

SDN and NFV

The concern of a network service provider is about the set of network devices (such as routers) and the control and management of the functions they perform (such as packet forwarding). Both SDF and NFV can be used seperately to provide this but SDN and NFV are not mutually exclusive. If both SDN and NFV are implemented for a network, the following relationships hold:

- The relationship between SDN and NFV is perhaps viewed as SDN functioning as an enabler of NFV.
- Network data plane functionality is implemented on VMs.

SDN and NFV

The concern of a network service provider is about the set of network devices (such as routers) and the control and management of the functions they perform (such as packet forwarding). Both SDF and NFV can be used seperately to provide this but SDN and NFV are not mutually exclusive. If both SDN and NFV are implemented for a network, the following relationships hold:

- The relationship between SDN and NFV is perhaps viewed as SDN functioning as an enabler of NFV.
- Network data plane functionality is implemented on VMs.
- The control plane functionality may be implemented on a dedicated SDN platform or on an SDN VM.

SDN and NFV

The concern of a network service provider is about the set of network devices (such as routers) and the control and management of the functions they perform (such as packet forwarding). Both SDF and NFV can be used seperately to provide this but SDN and NFV are not mutually exclusive. If both SDN and NFV are implemented for a network, the following relationships hold:

- The relationship between SDN and NFV is perhaps viewed as SDN functioning as an enabler of NFV.
- Network data plane functionality is implemented on VMs.
- The control plane functionality may be implemented on a dedicated SDN platform or on an SDN VM.
- A major challenge with NFV is to best enable the user to configure a network so that VNFs running on servers are connected to the network at the appropriate place, with the appropriate connectivity to other VNFs, and with desired QoS

SDN and NFV

The concern of a network service provider is about the set of network devices (such as routers) and the control and management of the functions they perform (such as packet forwarding). Both SDF and NFV can be used seperately to provide this but SDN and NFV are not mutually exclusive. If both SDN and NFV are implemented for a network, the following relationships hold:

- The relationship between SDN and NFV is perhaps viewed as SDN functioning as an enabler of NFV.
- Network data plane functionality is implemented on VMs.
- The control plane functionality may be implemented on a dedicated SDN platform or on an SDN VM.
- A major challenge with NFV is to best enable the user to configure a network so that VNFs running on servers are connected to the network at the appropriate place, with the appropriate connectivity to other VNFs, and with desired QoS
- With SDN, users and orchestration software can dynamically configure the network and the distribution and connectivity of VNFs.

SDN and NFV

The concern of a network service provider is about the set of network devices (such as routers) and the control and management of the functions they perform (such as packet forwarding). Both SDF and NFV can be used seperately to provide this but SDN and NFV are not mutually exclusive. If both SDN and NFV are implemented for a network, the following relationships hold:

- The relationship between SDN and NFV is perhaps viewed as SDN functioning as an enabler of NFV.
- Network data plane functionality is implemented on VMs.
- The control plane functionality may be implemented on a dedicated SDN platform or on an SDN VM.
- A major challenge with NFV is to best enable the user to configure a network so that VNFs running on servers are connected to the network at the appropriate place, with the appropriate connectivity to other VNFs, and with desired QoS
- With SDN, users and orchestration software can dynamically configure the network and the distribution and connectivity of VNFs.
- Without SDN, NFV requires much more manual intervention, especially when resources beyond the scope of NFVI are part of the environment.

Some of the ways that ETSI believes that NFV and SDN complement each other include the following:

- The SDN controller fits well into the broader concept of a network controller in an NFVI network domain.

Some of the ways that ETSI believes that NFV and SDN complement each other include the following:

- The SDN controller fits well into the broader concept of a network controller in an NFVI network domain.
- SDN can play a significant role in the orchestration of the NFVI resources, both physical and virtual, enabling functionality such as provisioning, configuration of network connectivity, bandwidth allocation, automation of operations, monitoring, security, and policy control.

Some of the ways that ETSI believes that NFV and SDN complement each other include the following:

- The SDN controller fits well into the broader concept of a network controller in an NFVI network domain.
- SDN can play a significant role in the orchestration of the NFVI resources, both physical and virtual, enabling functionality such as provisioning, configuration of network connectivity, bandwidth allocation, automation of operations, monitoring, security, and policy control.
- SDN can provide the network virtualization required to support multitenant NFVIs.

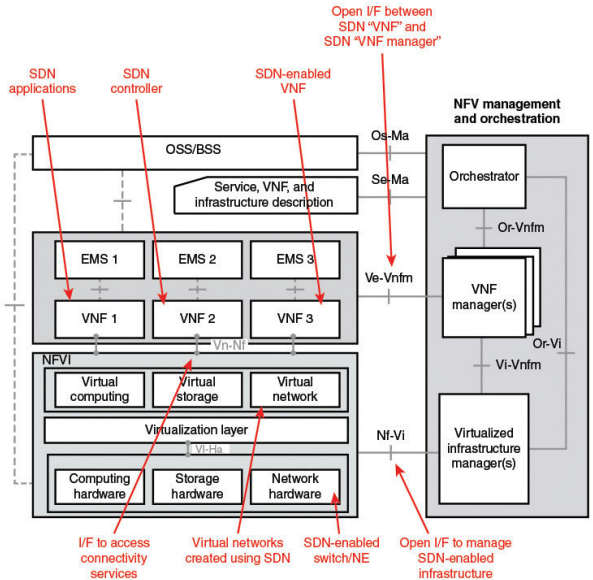
Some of the ways that ETSI believes that NFV and SDN complement each other include the following:

- The SDN controller fits well into the broader concept of a network controller in an NFVI network domain.
- SDN can play a significant role in the orchestration of the NFVI resources, both physical and virtual, enabling functionality such as provisioning, configuration of network connectivity, bandwidth allocation, automation of operations, monitoring, security, and policy control.
- SDN can provide the network virtualization required to support multitenant NFVIs.
- Forwarding graphs can be implemented using the SDN controller to provide automated provisioning of service chains, while ensuring strong and consistent implementation of security and other policies.

Some of the ways that ETSI believes that NFV and SDN complement each other include the following:

- The SDN controller fits well into the broader concept of a network controller in an NFVI network domain.
- SDN can play a significant role in the orchestration of the NFVI resources, both physical and virtual, enabling functionality such as provisioning, configuration of network connectivity, bandwidth allocation, automation of operations, monitoring, security, and policy control.
- SDN can provide the network virtualization required to support multitenant NFVIs.
- Forwarding graphs can be implemented using the SDN controller to provide automated provisioning of service chains, while ensuring strong and consistent implementation of security and other policies.
- The SDN controller can be run as a VNF, possibly as part of a service chain including other VNFs. For example, applications and services originally developed to run on the SDN controller could also be implemented as separate VNFs.

Mapping of SDN Components with NFV Architecture



- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.
- SDN controller can be virtualized, running as a VNF with its EM and VNF manager. Note that there may be SDN controllers for the physical infrastructure, the virtual infrastructure, and the virtual and physical network functions. As such, some of these SDN controllers may reside in the NFVI or management and orchestration (MANO) functional blocks (not shown in figure).

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.
- SDN controller can be virtualized, running as a VNF with its EM and VNF manager. Note that there may be SDN controllers for the physical infrastructure, the virtual infrastructure, and the virtual and physical network functions. As such, some of these SDN controllers may reside in the NFVI or management and orchestration (MANO) functional blocks (not shown in figure).
- SDN enabled VNF includes any VNF that may be under the control of an SDN controller (for example, virtual router, virtual firewall).

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.
- SDN controller can be virtualized, running as a VNF with its EM and VNF manager. Note that there may be SDN controllers for the physical infrastructure, the virtual infrastructure, and the virtual and physical network functions. As such, some of these SDN controllers may reside in the NFVI or management and orchestration (MANO) functional blocks (not shown in figure).
- SDN enabled VNF includes any VNF that may be under the control of an SDN controller (for example, virtual router, virtual firewall).
- SDN applications, for example service chaining applications, can be VNF themselves.

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.
- SDN controller can be virtualized, running as a VNF with its EM and VNF manager. Note that there may be SDN controllers for the physical infrastructure, the virtual infrastructure, and the virtual and physical network functions. As such, some of these SDN controllers may reside in the NFVI or management and orchestration (MANO) functional blocks (not shown in figure).
- SDN enabled VNF includes any VNF that may be under the control of an SDN controller (for example, virtual router, virtual firewall).
- SDN applications, for example service chaining applications, can be VNF themselves.
- Nf-Vi interface allows management of the SDN enabled infrastructure.

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.
- SDN controller can be virtualized, running as a VNF with its EM and VNF manager. Note that there may be SDN controllers for the physical infrastructure, the virtual infrastructure, and the virtual and physical network functions. As such, some of these SDN controllers may reside in the NFVI or management and orchestration (MANO) functional blocks (not shown in figure).
- SDN enabled VNF includes any VNF that may be under the control of an SDN controller (for example, virtual router, virtual firewall).
- SDN applications, for example service chaining applications, can be VNF themselves.
- Nf-Vi interface allows management of the SDN enabled infrastructure.
- Ve-Vnfm interface is used between the SDN VNF (SDN controller VNF, SDN network functions VNF, SDN applications VNF) and their respective VNF Manager for lifecycle management.

- SDN enabled switch/NEs include physical switches, hypervisor virtual switches, and embedded switches on the NICs.
- Virtual networks created using an infrastructure network SDN controller provide connectivity services between VNFC instances.
- SDN controller can be virtualized, running as a VNF with its EM and VNF manager. Note that there may be SDN controllers for the physical infrastructure, the virtual infrastructure, and the virtual and physical network functions. As such, some of these SDN controllers may reside in the NFVI or management and orchestration (MANO) functional blocks (not shown in figure).
- SDN enabled VNF includes any VNF that may be under the control of an SDN controller (for example, virtual router, virtual firewall).
- SDN applications, for example service chaining applications, can be VNF themselves.
- Nf-Vi interface allows management of the SDN enabled infrastructure.
- Ve-Vnfm interface is used between the SDN VNF (SDN controller VNF, SDN network functions VNF, SDN applications VNF) and their respective VNF Manager for lifecycle management.
- Vn-Nf allows SDN VNFs to access connectivity services between VNFC

Importance of SDN/NFV for IoT

Due to heterogenous and dynamic nature of IoT traffic it is very difficult to manage the internet traffic using traditional networking infrastructure. There are several challanges that arises due to ever increasing IoT services.

Importance of SDN/NFV for IoT

Due to heterogenous and dynamic nature of IoT traffic it is very difficult to manage the internet traffic using traditional networking infrastructure. There are several challanges that arises due to ever increasing IoT services.

- The complexity, possible limitations and heterogeneity of various IoT devices connected to the internet will require more specific tools to manage them and to improve the performance of the whole network.

Importance of SDN/NFV for IoT

Due to heterogenous and dynamic nature of IoT traffic it is very difficult to manage the internet traffic using traditional networking infrastructure. There are several challanges that arises due to ever increasing IoT services.

- The complexity, possible limitations and heterogeneity of various IoT devices connected to the internet will require more specific tools to manage them and to improve the performance of the whole network.
- Traditional architectures and network protocols for IoT devices are not designed to support high level of scalability, high amount of traffic and mobility together with the above mentioned requirements. They are inefficient and have limitations to satisfy these new requirements.

Importance of SDN/NFV for IoT

Due to heterogenous and dynamic nature of IoT traffic it is very difficult to manage the internet traffic using traditional networking infrastructure. There are several challanges that arises due to ever increasing IoT services.

- The complexity, possible limitations and heterogeneity of various IoT devices connected to the internet will require more specific tools to manage them and to improve the performance of the whole network.
- Traditional architectures and network protocols for IoT devices are not designed to support high level of scalability, high amount of traffic and mobility together with the above mentioned requirements. They are inefficient and have limitations to satisfy these new requirements.
- it's difficult to manage incredible number of connected devices generating an impressive amount of data as a whole, without having elasticity and flexibility inherently defined in the network. If the networks are not prepared, the flood of IoT where a lot of traffic are generated could leave the network paralysed.

Advantages of using SDN/NFV for IoT

Solving Interoperability in the Internet of Things

- Interoperability challenge in IoT arises when we have heterogeneous devices exchanging data formats and diverse protocols for machine to machine (M2M) data exchange, and also with the interconnectivity of large number of different devices, there is lack of cooperation and capability mismatch between devices which can hinder the performance of the network.

Advantages of using SDN/NFV for IoT

Solving Interoperability in the Internet of Things

- Interoperability challenge in IoT arises when we have heterogeneous devices exchanging data formats and diverse protocols for machine to machine (M2M) data exchange, and also with the interconnectivity of large number of different devices, there is lack of cooperation and capability mismatch between devices which can hinder the performance of the network.
- SDN approach brings flexibility which can be used to allow different objects connected to heterogeneous networks to communicate with each other. This will be able to handle simultaneous connections of various communication technologies.

Advantages of using SDN/NFV for IoT

Solving Interoperability in the Internet of Things

- Interoperability challenge in IoT arises when we have heterogeneous devices exchanging data formats and diverse protocols for machine to machine (M2M) data exchange, and also with the interconnectivity of large number of different devices, there is lack of cooperation and capability mismatch between devices which can hinder the performance of the network.
- SDN approach brings flexibility which can be used to allow different objects connected to heterogeneous networks to communicate with each other. This will be able to handle simultaneous connections of various communication technologies.
- Network management decisions such as routing, scheduling can be done at the SDN controller and moreover, the programmability allows for any updates for new proposals or even clean state approaches.

Discoverability

- Discoverability in IoT devices is one of the main factors in achieving a successful deployment of IoT applications to prevent long outages and configuration errors.

Discoverability

- Discoverability in IoT devices is one of the main factors in achieving a successful deployment of IoT applications to prevent long outages and configuration errors.
- The ability to self-configure and adapt to the environment without human intervention brings about requirements such as resource and service discovery.

Discoverability

- Discoverability in IoT devices is one of the main factors in achieving a successful deployment of IoT applications to prevent long outages and configuration errors.
- The ability to self-configure and adapt to the environment without human intervention brings about requirements such as resource and service discovery.
- It is not feasible to manually configure each and every device in order to discover which objects are nearby and which functions they provide.

Discoverability

- Discoverability in IoT devices is one of the main factors in achieving a successful deployment of IoT applications to prevent long outages and configuration errors.
- The ability to self-configure and adapt to the environment without human intervention brings about requirements such as resource and service discovery.
- It is not feasible to manually configure each and every device in order to discover which objects are nearby and which functions they provide.
- SDN approach can be used to address this issue which allows applications to operate with devices with minimal or no configuration.

Security

- Having large number of heterogeneous devices that are involved in the Internet of Things, there is high propensity to be vulnerable to attacks, and ensuring data protection, data authentication needs to be taken seriously.

Security

- Having large number of heterogeneous devices that are involved in the Internet of Things, there is high propensity to be vulnerable to attacks, and ensuring data protection, data authentication needs to be taken seriously.
- Security threats can be easier to attack through the improved visibility SDN provides to the network.

Security

- Having large number of heterogeneous devices that are involved in the Internet of Things, there is high propensity to be vulnerable to attacks, and ensuring data protection, data authentication needs to be taken seriously.
- Security threats can be easier to attack through the improved visibility SDN provides to the network.
- SDN can also provide a dynamic, intelligent, selflearning layered model of security that provides access rules to ensure authorization for people who are allowed to change the configuration of devices.

Management

- The SDN controller manages and supervises the entire network. The centralized position of the SDN controller makes it suitable to have a global vision of the network topology and conditions, performing network control such as routing and QoS control.

Management

- The SDN controller manages and supervises the entire network. The centralized position of the SDN controller makes it suitable to have a global vision of the network topology and conditions, performing network control such as routing and QoS control.
- The controller can determine the best routing decisions and inserting these decisions into the flow tables.

Management

- The SDN controller manages and supervises the entire network. The centralized position of the SDN controller makes it suitable to have a global vision of the network topology and conditions, performing network control such as routing and QoS control.
- The controller can determine the best routing decisions and inserting these decisions into the flow tables.
- The sensor nodes do not make routing decisions but only forward and drop packets according to the rules set by the controller.

Management

- The SDN controller manages and supervises the entire network. The centralized position of the SDN controller makes it suitable to have a global vision of the network topology and conditions, performing network control such as routing and QoS control.
- The controller can determine the best routing decisions and inserting these decisions into the flow tables.
- The sensor nodes do not make routing decisions but only forward and drop packets according to the rules set by the controller.
- The scheduling must be built over defined routes and the controller can optimize the sleep/active cycles of the sensors by choosing the most energy-efficient set in every scheduling cycle.

Management

- The SDN controller manages and supervises the entire network. The centralized position of the SDN controller makes it suitable to have a global vision of the network topology and conditions, performing network control such as routing and QoS control.
- The controller can determine the best routing decisions and inserting these decisions into the flow tables.
- The sensor nodes do not make routing decisions but only forward and drop packets according to the rules set by the controller.
- The scheduling must be built over defined routes and the controller can optimize the sleep/active cycles of the sensors by choosing the most energy-efficient set in every scheduling cycle.
- Latency can be reduced and significant energy savings can be achieved.

Scalability Issues

- The rapid growth of embedded technologies is leading to enormous deployment of miniaturized devices (sensors, actuators, etc.). As the number of devices grows, the data produced by these devices grow unboundedly.

Scalability Issues

- The rapid growth of embedded technologies is leading to enormous deployment of miniaturized devices (sensors, actuators, etc.). As the number of devices grows, the data produced by these devices grow unboundedly.
- SDN can improve scalability issues in IoT network where the SDN controller oversees the network domain and communicates with other SDN controllers to exchange aggregated network-wide formation.

Scalability Issues

- The rapid growth of embedded technologies is leading to enormous deployment of miniaturized devices (sensors, actuators, etc.). As the number of devices grows, the data produced by these devices grow unboundedly.
- SDN can improve scalability issues in IoT network where the SDN controller oversees the network domain and communicates with other SDN controllers to exchange aggregated network-wide formation.
- The distributed SDN model tends to share the load among several controllers. This easily helps in adapting to the users and applications.

Scalability Issues

- The rapid growth of embedded technologies is leading to enormous deployment of miniaturized devices (sensors, actuators, etc.). As the number of devices grows, the data produced by these devices grow unboundedly.
- SDN can improve scalability issues in IoT network where the SDN controller oversees the network domain and communicates with other SDN controllers to exchange aggregated network-wide formation.
- The distributed SDN model tends to share the load among several controllers. This easily helps in adapting to the users and applications.
- This distributed way brings many benefits such as (1) scalability (2) reducing latency from the sensor nodes to closest controller, (3) load balancing (4) fault-tolerance among others.

Scalability Issues

- The rapid growth of embedded technologies is leading to enormous deployment of miniaturized devices (sensors, actuators, etc.). As the number of devices grows, the data produced by these devices grow unboundedly.
- SDN can improve scalability issues in IoT network where the SDN controller oversees the network domain and communicates with other SDN controllers to exchange aggregated network-wide formation.
- The distributed SDN model tends to share the load among several controllers. This easily helps in adapting to the users and applications.
- This distributed way brings many benefits such as (1) scalability (2) reducing latency from the sensor nodes to closest controller, (3) load balancing (4) fault-tolerance among others.
- With multiple controllers, this can be used to offload computational tasks which brings benefits in terms of administration. Each domain has its SDN controller which controls all traffic in its domain. When one SDN controller fails, another SDN controller can take control to avoid network failures.

Application Specific Requirements

- Some IoT applications works in realtime, so a need to support real-time applications is required in an IoT environment where it is expected to monitor different things at different time periods.

Application Specific Requirements

- Some IoT applications works in realtime, so a need to support real-time applications is required in an IoT environment where it is expected to monitor different things at different time periods.
- SDN is able to strengthen network controlling ability and also perform dynamic adaptation of control logic by the devices in real-time.

Application Specific Requirements

- Some IoT applications works in realtime, so a need to support real-time applications is required in an IoT environment where it is expected to monitor different things at different time periods.
- SDN is able to strengthen network controlling ability and also perform dynamic adaptation of control logic by the devices in real-time.
- SDN and its extension to the Wireless Sensors and Actuators Domain will give the possibility to support application specific requirements with control logic that jointly act at the network and processing level enhancing the QoS/QoE of the entire system.

Service Chain Simplification and Application Provisioning

- SDN and NFV can make service chain shorter and simpler by increasing the efficiency and capacity of the network without radically changing hardware making it easier to spin up IoT applications.

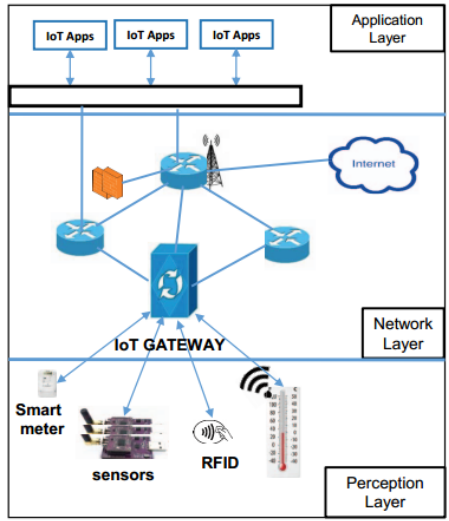
Service Chain Simplification and Application Provisioning

- SDN and NFV can make service chain shorter and simpler by increasing the efficiency and capacity of the network without radically changing hardware making it easier to spin up IoT applications.
- SDN & NFV will help service providers to enhance their service delivery infrastructure which is very important in the contribution to the IoT.

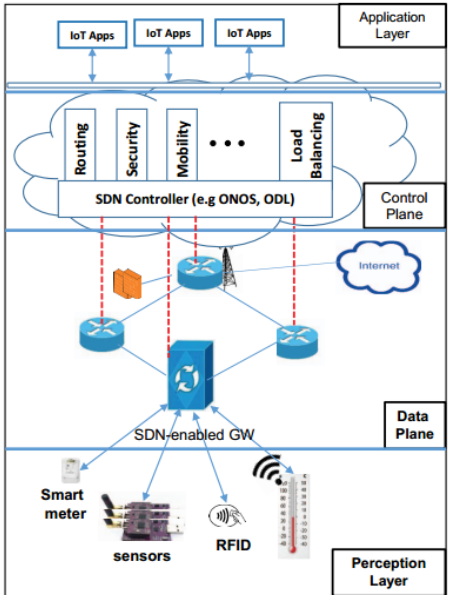
Service Chain Simplification and Application Provisioning

- SDN and NFV can make service chain shorter and simpler by increasing the efficiency and capacity of the network without radically changing hardware making it easier to spin up IoT applications.
- SDN & NFV will help service providers to enhance their service delivery infrastructure which is very important in the contribution to the IoT.
- This addresses the issues of network ossification by utilizing the network resources in a better way and transiting to software-centric programmable networks due to the rapid evolution and dynamicity of IoT applications.

A Simple IoT Netowrk



IOT with SDN


IoT Apps
IoT Apps
IoT Apps
Application Layer
Routing
Security
Mobility
Load Balancing
SDN Controller (e.g ONOS, ODL)
Control Plane
Internet
SDN-enabled GW
Data Plane
Smart meter
sensors
RFID
Perception Layer

IOT with SDN and NFV