

**An incubated co-working space for technology innovation**

**NFV Report**

**Working Document**

Contents

[**An incubated co-working space for technology innovation 1**](#_Toc420842126)

[**NFV Report 1**](#_Toc420842127)

[**Team Register iii**](#_Toc420842128)

[**Description of Acronyms iv**](#_Toc420842129)

[**BACKGROUND 5**](#_Toc420842130)

[**Introduction of network function virtualization 5**](#_Toc420842131)

[**Definition 6**](#_Toc420842132)

[**What Makes NFV Different? 6**](#_Toc420842133)

[**Overview of the ETSI NFV ISG 7**](#_Toc420842134)

[**NFV architecture Model 8**](#_Toc420842135)

[**Why NFV is the future? 9**](#_Toc420842136)

[**Benefits of NFV 9**](#_Toc420842137)

[**The Changing Telecoms industry landscape 11**](#_Toc420842138)

[**Enablers for network function virtualization 12**](#_Toc420842139)

[**Cloud computing 12**](#_Toc420842140)

[**Industry Standard High Volume Servers 12**](#_Toc420842141)

[**Challenges for Network Functions Virtualization 14**](#_Toc420842142)

[**Recommendations/Call for Action 16**](#_Toc420842143)

[**The key objectives, activities and outputs of the ISG are expected to be: 17**](#_Toc420842144)

[**MANO Functional Blocks 18**](#_Toc420842145)

[**NFV Orchestrator: 18**](#_Toc420842146)

[**VNF Manager: 18**](#_Toc420842147)

[**Virtualized Infrastructure Manager (VIM): 18**](#_Toc420842148)

[**NFV Management and Orchestration Architecture 18**](#_Toc420842149)

[**NFV Entities to deploy and manage: 19**](#_Toc420842150)

[**SDN DEFINITION 20**](#_Toc420842151)

[**(NFV) NETWORK FUNCTION VIRTUALISATION AND (SDN) SOFTWARE DEFINED NETWORKS 21**](#_Toc420842152)

[**WHY NETWORK FUNCTIONS VIRTUALIZATION & SDN 23**](#_Toc420842153)

[**How NFV will push SDN beyond the data center 24**](#_Toc420842154)

[**Will NFV unify differing SDN models? 24**](#_Toc420842155)

[**Virtualization Approach 24**](#_Toc420842156)

[**Virtualization Targets 25**](#_Toc420842157)

[**Virtualization of the CPE (vE-CPE) 25**](#_Toc420842158)

[**Virtualization of the PE(vPE) 26**](#_Toc420842159)

[**Coexistence of Virtualized and non-virtualized network functions 26**](#_Toc420842160)

[**Implementation of VMware vSphere 27**](#_Toc420842161)

[**Overview 27**](#_Toc420842162)

[**What is vSphere? 27**](#_Toc420842163)

[**What is ESX? 27**](#_Toc420842164)

[**Key features and capabilities of ESXi 4 28**](#_Toc420842165)

[**Prior to ESX Installation 28**](#_Toc420842166)

[**Installation of ESXi 4 29**](#_Toc420842167)

[**Setting the Root Password 29**](#_Toc420842168)

[**Configuring Management Network 30**](#_Toc420842169)

[**Setting-up Network Adapter 30**](#_Toc420842170)

[**Configuring IP and DNS 31**](#_Toc420842171)

[**Installation of vCenter 32**](#_Toc420842172)

[**vCenter Web Access 33**](#_Toc420842173)

[**Configure Your Database for vCenter 33**](#_Toc420842174)

[**Create the Database and Setting Permissions 33**](#_Toc420842175)

[**Configuring a DSN Connection 34**](#_Toc420842176)

[**Setting Up vCenter 35**](#_Toc420842177)

[**vCenter Converter 37**](#_Toc420842178)

[**What is VMware vCenter Converter? 37**](#_Toc420842179)

[**Key features and capabilities 37**](#_Toc420842180)

[**How Does VMware vCenter Converter Work? 38**](#_Toc420842181)

[**VMware vCloud Air Disaster Recovery 38**](#_Toc420842182)

[**STANDARD NERTWORKING 38**](#_Toc420842183)

[**vSphere Standard Switches 39**](#_Toc420842184)

[**Standard Port Groups 39**](#_Toc420842185)

[**Add a Virtual Machine Port Group 41**](#_Toc420842186)

[**VMkernel Networking Configuration 41**](#_Toc420842187)

[**Introduction to Storage 45**](#_Toc420842188)

[**Storage Virtualization 45**](#_Toc420842189)

[**Types of Physical Storage 46**](#_Toc420842190)

[**Local Storage 46**](#_Toc420842191)

[**Networked Storage 47**](#_Toc420842192)

[**How Virtual Machines Access Storage 47**](#_Toc420842193)

[**Conclusion 48**](#_Toc420842194)

[**References 49**](#_Toc420842195)

**Team Register**

|  |  |
| --- | --- |
| **Member Name** | **Role Description** |
| Vuyiseka Mqwala | Leader |
| Mandilakhe Hlaula | Member |
| Pumza Tyotyo | Member |

|  |  |
| --- | --- |
| **Acronym** | **Description** |
| CH | The Cortex Hub |
| VNF | Virtualized Network Function |
| NFV | Network Function Virtualized |
| ETSI | European Telecommunications Standards Institute |
| SDN | Software Defined Network |
| ESXi | Exclusive hypervisor |
| VLAN | Virtual Local Area Network |
| IP | Internet Protocol |
| DSN | Data Source Name |
| NIC | Network Interface Card |
| CD | Compact Disc |
| SQL | Standardized Query Language |
| ISO | International Organization for Standardization |
| ARP | Address Resolution Protocol |
| DSN | Data Source Name |
| VLAN ID | Virtual Local Area Network Tagging |
| VDS | Virtual Dedicated Server |
| LACP | Link Aggregation Control Protocol |
| ERSPAN | Encapsulated Remote SPAN |
| IPFIX | Internet Protocol Flow Information Export |
| SR-ION | STRONTIUM ION |
| VSS | Volume Shadow Copy Service |
| SCSI | Small Computer System Interface |
| LSI | Latent semantic indexing |
| SAS | Statistical Analysis System |
| ISCSI | Internet Small Computer System Interface |
| FCoE | Fibre Channel over Ethernet |
| RDM | Raw Device Mapping |
| SATA | Serial Advanced Technology Attachment |
| IDE | integrated development environment |

**BACKGROUND**

NFV originated within the highly competitive service provider community, as they looked for ways to cut costs and accelerate the roll out of profitable services to better monetize their networks and grow their revenues. Hardware-based network appliances, which are typically expensive and complex to deploy and manage, were limiting the providers’ ability to consolidate functionality and quickly trial new services. Within an increasingly virtualized environment, providers wanted to be able to deploy network functionality whenever and wherever it was needed; they didn’t want to be tied to the capabilities of a specific appliance or topology. They felt if they could decouple the network services from the hardware, it would allow them to deploy networking components that could truly fit and support a fully virtualized infrastructure, including servers, storage and even other networks. A few providers came together, within the European Telecommunications Standards Institute’s (ETSI) and created the Industry Specification Group (ISG) for NFV to accelerate the progress of virtualizing network functions. Launched in January of 2013, the ETSI ISG for NFV has been working to develop the requirements and architecture of virtualized network functions in a telecommunication’s network. In September of 2014, the Linux Foundation announced the Open Platform for NFV Project (OPNFV), which is an open source, carrier-grade integrated platform that aims to help bring new NFV products and services to the industry faster. The goal is to make the recommendations and standards that come out of the ETSI ISG a reality, by leveraging the cumulative resources of the open source community. ETSI ISG for NFV and OPNFV will work closely to advance NFV concepts and technologies. We are starting to see some of the fruits of the industry’s labor, with NFV trials.

**Introduction of network function virtualization**

Network operators’ networks are populated with a large and increasing variety of proprietary hardware appliances. To launch a new network service often requires yet another variety and finding the space and power to accommodate these boxes is becoming increasingly difficult; compounded by the increasing costs of energy, capital investment challenges and the rarity of skills necessary to design, integrate and operate increasingly complex hardware-based appliances. Moreover, 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. Worse, hardware lifecycles are becoming shorter as technology and services innovation accelerates, inhibiting the roll out of new revenue earning network services and constraining innovation in an increasingly network-centric connected world (Margaret Chiosi, 2013).

Network Functions Virtualization (NFV) is a core structural change in the way telecommunication infrastructure gets deployed. This in turn will bring significant changes in the way that applications are delivered to service providers. NFV will bring cost efficiencies, time-to-market improvements and innovation to the telecommunication industry infrastructure and applications. NFV will achieve this through disaggregation of the traditional roles and technology involved in telecommunications applications (Margaret Chiosi, 2013).

**Definition**

Network Functions Virtualization aims to transform the way that network operators architect networks by evolving standard IT virtualization technology to consolidate many network equipment types onto industry standard high volume servers, switches and storage, which could be located in Datacenters, Network Nodes and in the end user premises, as illustrated in Figure 1. It involves the implementation of network functions in software that can run on a range of industry standard server hardware, and that can be moved to, or instantiated in, various locations in the network as required, without the need for installation of new equipment (Margaret Chiosi, 2013).



**What Makes NFV Different?**

While PC-based network devices have been available since the '80s, they were generally used by small companies and networking enthusiasts who didn't or couldn't afford to buy a commercial-based solution. In the last few years many drivers have brought PC-based networking devices back into the limelight, including: Ethernet as the last mile, better network interface cards, and Intel's focus on networking  
processing in its last few generation of chips. Today many vendors are producing PC-based network devices. Advancements in packet handling within Intel's processors, allowing processor cores to be re-programmed into network processors, allow PCbased network devices to push 10's or even 100's of Gbp/s. Values of NFV  
Some of the values to the NFV concept are speed, agility, and cost reduction. By centralizing designs around commodity server hardware, network operators can:

· Do a single PoP/Site design based on commodity compute hardware;

Avoiding designs involving one-off installs of appliances that have different power, cooling and space needs simplifies planning.

Utilize resources more effectively;

Virtualization allows providers to allocate only the necessary resources needed by each feature/function.

Deploy network functions without having to send engineers to each site; “Truck Rolls” are costly both from a time and money standpoint.

Achieve Reductions in OpEX and CapEX; and,

Achieve Reduction of system complexity i.e Deliver Agility and Flexibility: quickly scale up or down services to address changing demands; support innovation by enabling services to be delivered via software on any industry-standard server hardware

Accelerate Time-to-Market: reducing the time to deploy new networking services to support changing business requirements, seize new market opportunities and improve return on investment of new services. 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.

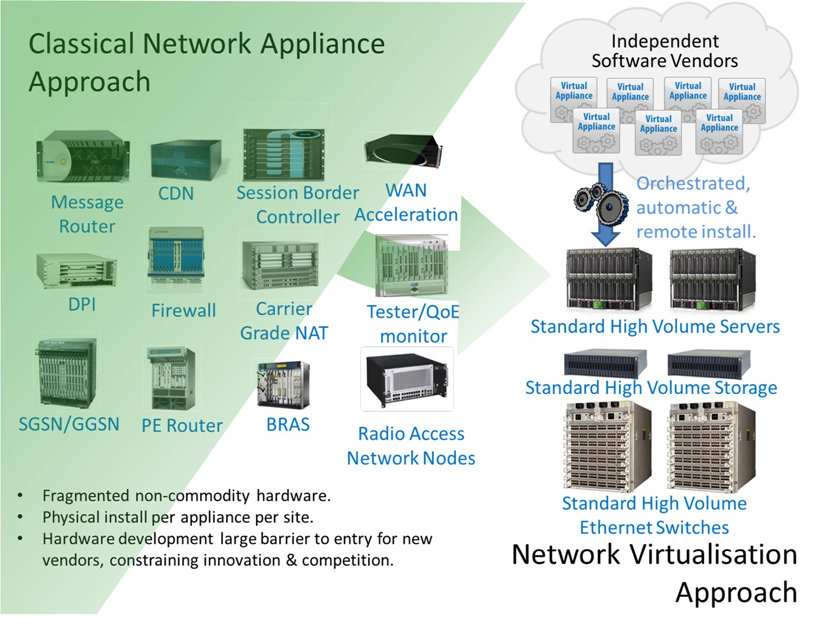
Error resilience: Ensuring the appropriate level of resilience to hardware and software failures.

Scale: Network Functions Virtualization will only scale if all of the functions can be automated.

**Overview of the ETSI NFV ISG**

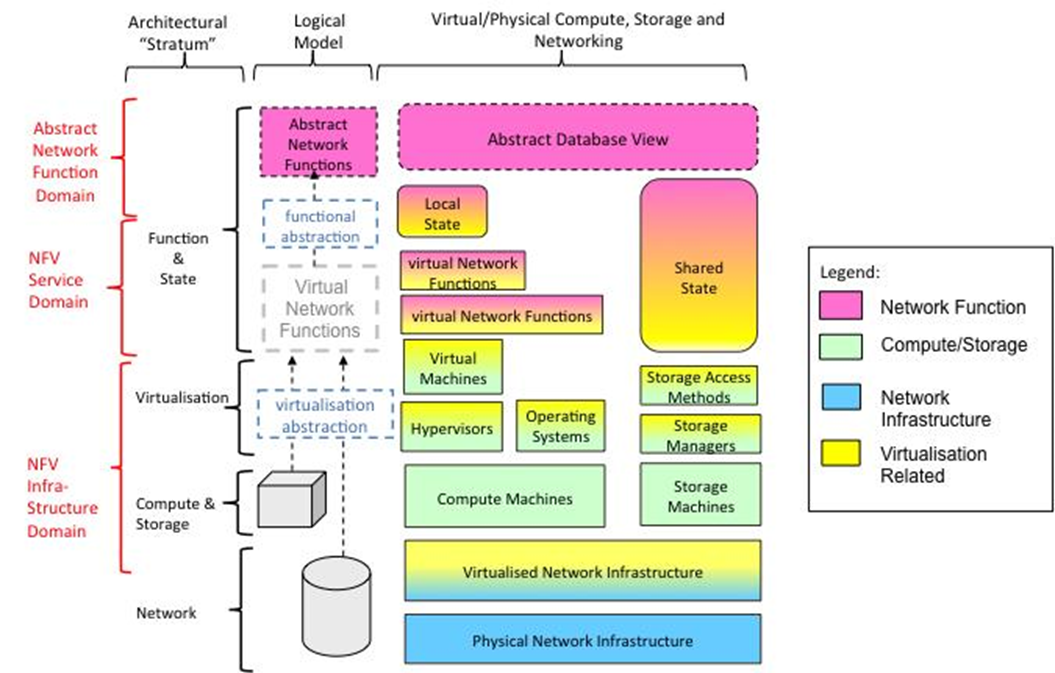
The ETSI Board approved foundation of the NFV ISG in time for publication of our first white paper last October. ETSI is a global organisation and has proved to be an excellent environment in which to Network Functions virtualization progress our work and we extend our thanks to the Director General and the ETSI Board for their accommodation and support. Although ETSI is a Standards Development  
Organisation (SDO), the objective of the NFV ISG is not to produce standards. The key objectives are to achieve industry consensus on business and technical requirements for NFV, and to agree common approaches to meeting these requirements. The outputs are openly published and shared with relevant standards bodies, industry fora and consortia to encourage a wider collaborative effort. The NFV ISG will collaborate with other SDOs if any standardization is necessary to meet the requirements. The NFV ISG also provides an environment for the industry to collaborate on Proof of Concept (PoC) platforms to demonstrate solutions which address the technical challenges for NFV implementation and to encourage growth of an open ecosystem

**NFV: Vision**



**NFV architecture Model**

1. Network operators have proven NFV feasibility via proof of concept test platforms
2. Network operators and vendors have identified numerous “fields of application” spanning all domains (fixed and mobile network infrastructures)
3. Significant CAPEX/OPEX benefits, leveraging also the economies of scale
4. Emerging virtual network appliance market
5. Novel ways to architect and operate networks, spawning a new wave of industry wide innovation
6. Network Functions Virtualization can dramatically change the telecom landscape and industry over the next 2-5 years



**Why NFV is the future?**

Recent tests by network operators and vendors have demonstrated that network functions can operate at the level of several millions of packets per sec, per CPU core. Demonstrates that standard high volume servers have sufficient processing performance to cost-effectively virtualized network appliances o The hypervisor need not be a bottleneck o The OS need not be a bottleneck.Total Cost of Ownership advantages are a huge driver – could be scenario specific but expect significant benefits, e.g., energy savings. Advances in virtualization & server technologies have propelled the importance and use of software in many applications and fields. A concerted industry effort is underway to accelerate this vision by encouraging common approaches which address the challenges for NFV

**Benefits of NFV**

We believe the application of Network Functions Virtualization brings many benefits to network operators, contributing to a dramatic change in the telecommunications industry landscape. Benefits we foresee include (not in any particular order):

* Reduced equipment costs and reduced power consumption through consolidating equipment and exploiting the economies of scale of the IT industry (~9.5M Servers shipped in 2011 compared with ~1.5M routers forecast for 2012).[1,2]
* Increased velocity of Time to Market by minimizing the typical network operator cycle of innovation. Economies of scale required to cover investments in hardware-based functionalities are no longer applicable for software-based development, making feasible other modes of feature evolution. Network Functions Virtualization should enable network operators to significantly reduce the maturation cycle.
* The possibility of running production, test and reference facilities on the same infrastructure provides much more efficient test and integration, reducing development costs and time to market.
* Targeted service introduction based on geography or customer sets is possible. Services can be rapidly scaled up/down as required. In addition, service velocity is improved by provisioning remotely in software without any site visits required to install new hardware.
* Enabling a wide variety of eco-systems and encouraging openness. It opens the virtual appliance market to pure software entrants, small players and academia, encouraging more innovation to bring new services and new revenue streams quickly at much lower risk.
* Optimizing network configuration and/or topology in near real time based on the actual traffic/mobility patterns and service demand. For example, optimization of the location & assignment of resources to network functions automatically and in near real time could provide protection against failures without engineering full 1+1 resiliency.
* Supporting multi-tenancy thereby allowing network operators to provide tailored services and connectivity for multiple users, applications or internal systems or other network operators, all co-existing on the same hardware with appropriate secure separation of administrative domains.
* Reduced energy consumption by exploiting power management features in standard servers and storage, as well as workload consolidation and location optimization. For example, relying on virtualization techniques it would be possible to concentrate the workload on a smaller number of servers during off-peak hours (e.g. overnight) so that all the other servers can be switched off or put into an energy saving mode.[3]

Improved operational efficiency by taking advantage of the higher uniformity of the physical network platform and its homogeneity to other support platforms:

IT orchestration mechanisms provide automated installation, scaling-up and scaling out of capacity, and re-use of Virtual Machine (VM) builds.[4]

Eliminating the need for application-specific hardware. The skills base across the industry for operating standard high volume IT servers is much larger and less fragmented than for today’s telecom-specific network equipment.

Reduction in variety of equipment for planning & provisioning. Assuming tools are developed for automation and to deal with the increased software complexity of virtualization.

Option to temporarily repair failures by automated re-configuration and moving network workloads onto spare capacity using IT orchestration mechanisms. This could be used to reduce the cost of 24/7 operations by mitigating failures automatically.

The potential to gain more efficiency between IT and Network Operations.

The potential to support in-service software upgrade (ISSU) with easy reversion by installing the new version of a Virtualized Network Appliance (VNA) as a new Virtual Machine (VM). Assuming traffic can be transferred from the old VM to the new VM without interrupting service. For some applications it may be necessary to synchronize the state of the new VM with the old VM.

**The Changing Telecoms industry landscape**

Although Network Functions Virtualization brings many advantages to the telecommunications industry it is likely to transform the vendor landscape. Each player will need to position/re-position itself in the new Network Functions Virtualization market (Margaret Chiosi, 2013).

This is not as disruptive as it may seem because network equipment vendors already implement some of their solutions by combining their proprietary software with industry standard hardware and software components, but in a proprietary way. Enabling their proprietary software to run on industry standard hardware in a standardized way may be a significant opportunity for existing players because their software and networking know-how is where the real value is in many cases. Some major industry players are already moving in this direction by offering virtualized versions of their products. The challenge for network operators is how to migrate their operations and skill base to a software based networking environment while carefully re-targeting investment to maximize reuse of existing systems and processes (Margaret Chiosi, 2013).

**Enablers for network function virtualization**

Several recent technology developments make the goals of Network Functions Virtualization achievable. This section describes these enablers and briefly discusses relevance (Margaret Chiosi, 2013).

**Cloud computing**

Network Functions Virtualization will leverage modern technologies such as those developed for cloud computing. At the core of these cloud technologies are virtualization mechanisms: hardware virtualization by means of hypervisors, as well as the usage of virtual Ethernet switches (e.g. switch) for connecting traffic between virtual machines and physical interfaces. For communication-oriented functions, high-performance packet processing is available through high-speed multi-core CPUs with high I/O bandwidth, the use of smart Ethernet NICs for load sharing and TCP Offloading, and routing packets directly to Virtual Machine memory, and poll-mode Ethernet drivers (rather than interrupt driven, for example Linux NAPI and Intel’s DPDK) (Torry Harris, 2009).

Cloud infrastructures provide methods to enhance resource availability and usage by means of orchestration and management mechanisms, applicable to the automatic instantiation of virtual appliances in the network, to the management of resources by assigning virtual appliances to the correct CPU core, memory and interfaces, to the re-initialization of failed VMs, to snapshot VM states and the migration of VMs (Torry Harris, 2009).

Finally, the availability of open APIs for management and data plane control, like Open Flow,

OpenStack, Open Naas or OGF’s NSI, provide an additional degree of integration of Network Functions Virtualization and cloud infrastructure (Torry Harris, 2009) (Conner, 2014).

**Industry Standard High Volume Servers**

The use of industry standard high volume servers is a key element in the economic case for Network Functions Virtualization. Network Functions Virtualization leverages the economies of scale of the IT industry. An industry standard high volume server is a server built using standardized IT components (for example x86 architecture) and sold in the millions. A common feature of industry standard high volume servers is that there is competitive supply of the subcomponents which are interchangeable inside the server (Margaret Chiosi, 2013).

We believe that Network Appliances which depend on the development of bespoke Application Specific Integrated Circuits (ASICs) will become increasingly uncompetitive against general purpose processors as the cost of developing ASICs increases exponentially with decreasing feature size.[5] Merchant silicon will still be applicable for commodity functions implemented at scale, and ASICs will still be applicable for some types of very high throughput applications (Margaret Chiosi, 2013).

**Challenges for Network Functions Virtualization**

There are a number of challenges to implement Network Functions Virtualization which need to be addressed by the community interested in accelerating progress. How this effort could be progressed is described later in this document. Challenges we have identified are (not in any particular order):-

* Portability/Interoperability. The ability to load and execute virtual appliances in different but standardized datacenter environments, provided by different vendors for different operators. The challenge is to define a unified interface which clearly decouples the software instances from the underlying hardware, as represented by virtual machines and their hypervisors. Portability and Interoperability is very important as it creates different ecosystems for virtual appliance vendors and datacenter vendors, while both ecosystems are clearly coupled and depend on each other. Portability also allows the operator the freedom to optimize the location and required resources of the virtual appliances without constraints (Margaret Chiosi, 2013).
* Performance Trade-Off. Since the Network Functions Virtualization approach is based on industry standard hardware (i.e. avoiding any proprietary hardware such as acceleration engines) a probable decrease in performance has to be taken into account. The challenge is how to keep the performance degradation as small as possible by using appropriate hypervisors and modern software technologies, so that the effects on latency, throughput and processing overhead are minimized. The available performance of the underlying platform needs to be clearly indicated, so that virtual appliances know what they can get from the hardware. The authors of the white paper believe that using the right technology choice will allow virtualization not only of network control functions but also data/user plane functions (Margaret Chiosi, 2013).
* Migration and co-existence of legacy & compatibility with existing platforms.
* Implementations of Network Functions Virtualization must co-exist with network operators’ legacy network equipment and be compatible with their existing Element Management Systems, Network Management Systems, OSS and BSS, and potentially existing IT orchestration systems if Network Functions Virtualization orchestration and IT orchestration are to converge. The Network Functions Virtualization architecture must support a migration path from today’s proprietary physical network appliance based solutions to more open standards based virtual network appliance solutions. In other words, Network Functions Virtualization must work in a hybrid network composed of classical physical network appliances and virtual network appliances. Virtual appliances must therefore use existing North Bound Interfaces (for management & control) and interwork with physical appliances implementing the same functions (Margaret Chiosi, 2013).
* Management and Orchestration. A consistent management and orchestration architecture is required. Network Functions Virtualization presents an opportunity, through the flexibility afforded by software network appliances operating in an open and standardized infrastructure, to rapidly align management and orchestration North Bound Interfaces to well defined standards and abstract specifications. This will greatly reduce the cost and time to integrate new virtual appliances into a network operator’s operating environment. Software Defined Networking (SDN) further extends this to streamlining the integration of packet and optical switches into the system e.g. a virtual appliance or Network Functions Virtualization orchestration system may control the forwarding behaviors of physical switches using SDN (Margaret Chiosi, 2013).
* Automation. Network Functions Virtualization will only scale if all of the functions can be automated. Automation of process is paramount to success (Margaret Chiosi, 2013).
* Security & Resilience. Network operators need to be assured that the security, resilience and availability of their networks are not impaired when virtualized network functions are introduced. Our initial expectations are that Network Functions Virtualization improves network resilience and availability by allowing network functions to be recreated on demand after a failure. A virtual appliance should be as secure as a physical appliance if the infrastructure, especially the hypervisor and its configuration, is secure. Network operators will be seeking tools to control and verify hypervisor configurations. They will also require security certified hypervisors and virtual appliances (Margaret Chiosi, 2013).
* Network Stability. Ensuring stability of the network is not impacted when managing and orchestrating a large number of virtual appliances between different hardware vendors and hypervisors. This is particularly important when, for example, virtual functions are relocated, or during re-configuration events (e.g. due to hardware and software failures) or due to cyber-attack. This challenge is not unique to Network Functions Virtualization. Potential instability might also occur in current networks, depending on unwanted combinations of diverse control and optimization mechanisms, for example acting on either the underlying transport network or on the higher layers’ components (e.g. flow admission control, congestion control, dynamic routing and allocations, etc.). It should be noted that occurrence of network instability may have primary effects, such as jeopardizing, even dramatically, performance parameters or compromising an optimized use of resources. Mechanisms capable of ensuring network stability will add further benefits to Network Functions Virtualization (Margaret Chiosi, 2013).
* Simplicity. Ensuring that virtualized network platforms will be simpler to operate than those that exist today. A significant and topical focus for network operators is simplification of the plethora of complex network platforms and support systems which have evolved over decades of network technology evolution, while maintaining continuity to support important revenue generating services. It is important to avoid trading one set of operational headaches for a different but equally intractable set of operational headaches (Margaret Chiosi, 2013).
* Integration. Seamless integration of multiple virtual appliances onto existing industry standard high volume servers and hypervisors is a key challenge for Network Functions Virtualization. Network operators need to be able to “mix & match” servers from different vendors, hypervisors from different vendors and virtual appliances from different vendors without incurring significant integration costs and avoiding lock-in. The ecosystem must offer integration services and maintenance and third-party support; it must be possible to resolve integration issues between several parties. The ecosystem will require mechanisms to validate new Network Functions Virtualization products. Tools must be identified and/or created to address these issues (Margaret Chiosi, 2013).

**Recommendations/Call for Action**

Network Functions Virtualization is already occurring. In a few years, we can expect the communications industry to look and feel similar to the IT industry. There will be a wider range of business models more suited to a software industry. Operations complexity will be abstracted away by more automation and self-provisioning will be more common. As detailed in this white paper, Network Functions Virtualization will deliver many benefits for network operators and their partners and customers whilst offering the opportunity to create new types of eco-systems (alongside traditional supply models based on preferred strategic partners) which will encourage and support rapid innovation with reduced cost and reduced risk. To reap these benefits the technical challenges, as described above, must be addressed by the industry (Margaret Chiosi, 2013).

The authors of this white paper believe that solutions to these technical challenges are available (or could be made available) and recommend that the IT and Network industries combine their complementary expertise and resources in a joint collaborative effort to reach broad agreement on standardized approaches and common architectures which address these technical challenges, and which are interoperable and have economies of scale (Margaret Chiosi, 2013).

To accelerate progress, a new network operator-led Industry Specification Group (ISG) with open membership is being setup under the auspices of ETSI to work through the technical challenges for Network Functions Virtualization as outlined in this white paper. The formal creation process of this ETSI ISG has been started and is expected to be completed by mid-November 2012 (Margaret Chiosi, 2013).

While ETSI will provide the organization for this initiative, the ISG will downstream its work to all relevant organizations and seek to encourage convergence of IT and Network Standardization efforts in this space (Margaret Chiosi, 2013).

**The key objectives, activities and outputs of the ISG are expected to be:**

* The ISG will initiate joint studies which address the technical challenges for Network Functions Virtualization as documented in this white paper. With consensus outputs detailed in white papers openly published to the wider industry. It is expected that study items would be identified in a “Network Operator Requirements” white paper prepared by the ISG as an initial deliverable (Margaret Chiosi, 2013).
* Although parented under ETSI, the ISG will not be a Standards Development Organization (SDO) in itself; it will reference the work of existing SDOs where applicable. Where there are gaps, it will encourage member contributions to SDOs requesting them to take into account the recommendations of the ISG (i.e. requesting updates to existing standards, or to create new standards if appropriate) (Margaret Chiosi, 2013).
* The ISG will identify other expert bodies that could assist with this work, and request that they undertake studies related to their scope to help move the industry forward.
* The ISG will facilitate members to share technical learning in a collaborative way which does not compromise confidentiality.
* The first meeting of the ISG will take place in January 2013. Details will be announced as soon as possible after publication of this white paper.
* In order to chart the way forward for Network Functions Virtualization, the wider industry is asked to provide feedback:
* Network Operators: To state their interest in joining the ISG, thus subscribing to the targets and approach as outlined in this white paper.
* Existing organizations and forums: To state if they see themselves being impacted by the work addressed in the ISG, and to provide information on how they would like to interact with it.
* Vendors: To state their interest in joining the ISG. Vendors are furthermore requested to state the study items in which they are most interested (Margaret Chiosi, 2013).

**MANO Functional Blocks**

**NFV Orchestrator:**

– On-boarding of new Network Service (NS), VNF-FG and VNF Packages

– NS lifecycle management (including instantiation, scale-out/in, performance measurements, event correlation, termination)

– Global resource management, validation and authorization of NFVI resource requests

– Policy management for NS instances

**VNF Manager:**

– Lifecycle management of VNF instances

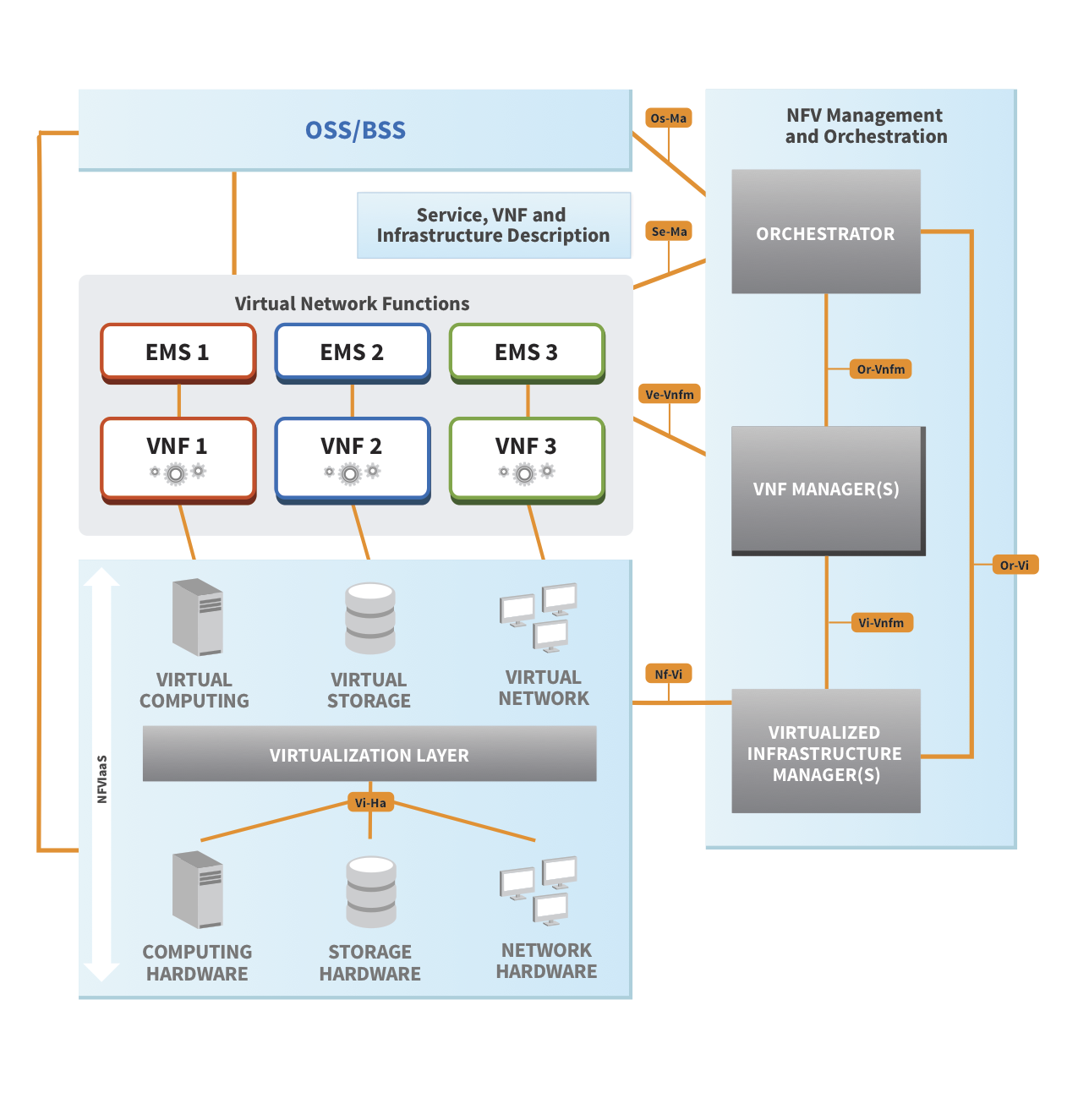
– Overall coordination and adaptation role for configuration and event reporting between NFVI and the E/NMS

**Virtualized Infrastructure Manager (VIM):**

– controlling and managing the NFVI compute, storage and network resources, within one operator’s infrastructure sub-domain

– Collection and forwarding of performance measurements and events

**NFV Management and Orchestration Architecture**



**NFV Entities to deploy and manage:**

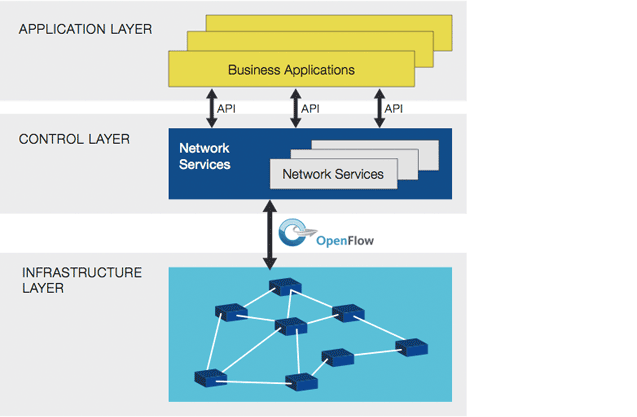
* Network Service (NS): – described by its descriptor file, orchestrated by NFVO, – may cover 1 or more VNF Graphs, VNFs and PNFs.
* VNF Forwarding Graph (VNF-FG): – described by its descriptor file, orchestrated by NFVO, – may cover VNF-FGs, VNFs and NFs
* VNF: – described by its descriptor file, instantiated by the VNF Manager, – covers VNF components (VNFC) each mapped to a VM described with the Virtual Deployment Unit descriptor (Conner, 2014).

# SDN DEFINITION

Software-Defined Networking (SDN) is an emerging architecture that is dynamic, manageable, cost-effective, and adaptable, making it ideal for the high-bandwidth, dynamic nature of today's applications. This architecture decouples the network control and forwarding functions enabling the network control to become directly programmable and the underlying infrastructure to be abstracted for applications and network services. The OpenFlow™ protocol is a foundational element for building SDN solutions.

The SDN architecture is:

* Directly programmable: Network control is directly programmable because it is decoupled from forwarding functions.
* Agile: Abstracting control from forwarding lets administrators dynamically adjust network-wide traffic flow to meet changing needs.
* Centrally managed: Network intelligence is (logically) centralized in software-based SDN controllers that maintain a global view of the network, which appears to applications and policy engines as a single, logical switch.
* Programmatically configured: SDN lets network managers configure, manage, secure, and optimize network resources very quickly via dynamic, automated SDN programs, which they can write themselves because the programs do not depend on proprietary software.
* Open standards-based and vendor-neutral: When implemented through open standards, SDN simplifies network design and operation because instructions are provided by SDN controllers instead of multiple, vendor-specific devices and protocols.



**(NFV) NETWORK FUNCTION VIRTUALISATION AND (SDN) SOFTWARE DEFINED NETWORKS**

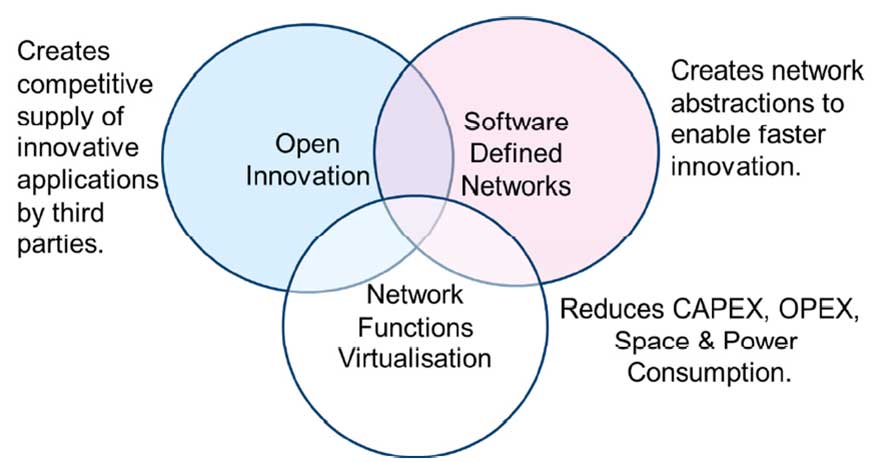
Network function virtualization (NFV) and software-defined networks (SDN) are two closely related technologies that often exist together, but not always. An SDN can be considered a series of network objects (such as switches, routers, firewalls) that deploy in a highly automated manner. The automation may be achieved by using commercial or open source tools customized according to the administrator's requirements. A full SDN may only cover relatively straightforward networking requirements, such as VLAN and interface provisioning.

In many cases, SDN will also be linked to server virtualization, providing the glue that sticks virtual networks together. This may involve NFV, but not necessarily. NFV is the process of moving services, such as load balancing, firewalls and IPS, away from dedicated hardware into a virtualized environment. This is, of course, part of a wider movement toward the virtualization of applications and services.

Functions such as caching and content control can easily be migrated to a virtualized environment but won't necessarily provide any significant reduction in operating costs until some intelligence is introduced. This is because a straight physical to virtual, from an operational perspective, achieves little beyond the initial reduction in power and rack-space consumption. Until some dynamic intelligence is introduced with an SDN technology, NFV inherits many of the same constraints as traditional hardware appliance deployments, such as static, administrator-defined and managed policies.

A good example is virtualized application delivery controllers (ADCs). With careful configuration, it is possible to react to the network state and spin up or down application servers as demands rise and fall. However, traditional hardware deployments have been able to do this for a while, and the configuration is very static; it doesn't cater to the scenario where the ADC itself becomes overloaded or an additional application needs to be brought into production quickly. With SDN features driving NFV, several useful things start to happen. The network can react when things need to change at the micro and macro level. An additional instance can be provisioned in a cluster of virtualized ADCs as the load increases, and production applications can easily be cloned and re-deployed in a development environment. The potential is endless.

So it's perfectly possible to have NFV without the inclusion of a full-blown SDN. The two are often deployed together, and an SDN that drives NFV is a very powerful combination.



Relationship with Software Defined Networks (SDN)

As shown in Figure 2, Network Functions Virtualization is highly complementary to Software Defined

Networking (SDN), but not dependent on it (or vice-versa). Network Functions Virtualization can be

Network Functions Virtualization

Implemented without a SDN being required, although the two concepts and solutions can be combined and potentially greater value accrued. Network Functions Virtualization Relationship with SDN

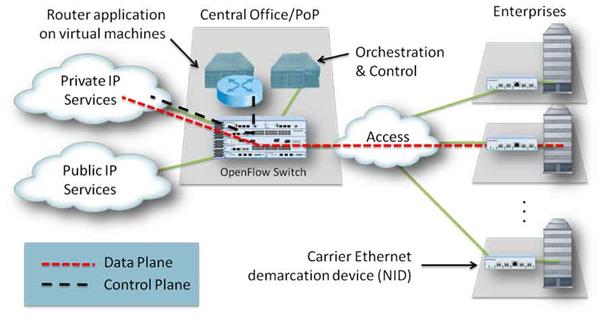
Network Functions Virtualization goals can be achieved using non-SDN mechanisms, relying on the techniques currently in use in many datacenters. But approaches relying on the separation of the control and data forwarding planes as proposed by SDN can enhance performance, simplify compatibility with existing deployments, and facilitate operation and maintenance procedures.

Network Functions Virtualization is able to support SDN by providing the infrastructure upon which the SDN software can be run. Furthermore, Network Functions Virtualization aligns closely with the

SDN objectives to use commodity servers and switches. We intend to work closely with organizations progressing work on SDN such as the Open Networking.

**WHY NETWORK FUNCTIONS VIRTUALIZATION & SDN**

Network Functions Virtualization (NFV) explicitly targets the two biggest problems facing network operators: bringing costs in line with revenue growth expectations and improving service velocity. NFV's premise is that industry standard IT virtualization technology (servers, switches, and storage) located in data centers, network nodes, or end-users' premises can be used to reduce the cost and increase the speed of service delivery for fixed and mobile networking functions. We believe Network Functions Virtualization is applicable to any data plane packet processing and control plane function in fixed and mobile network infrastructures. Finally, SDN is introduced to separate the control and data, as shown in Figure. Now, the data packets are forwarded by an optimized data plane, while the routing (control plane) function is running in a virtual machine running in a rack mount server.



**How NFV will push SDN beyond the data center**

NFV's use of virtual network overlays could also drive an expansion of this SDN model beyond the data center where it's focused most often today. If NFV allows services to be composed of virtual functions hosted in different data centers, that would require virtual networks to stretch across data centers and become end-to-end. An end-to-end virtual network would be far more interesting to enterprises than one limited to the data center. Building application-specific networks that extend to the branch locations might usher in a new model for application access control, application performance management and even application security.

**Will NFV unify differing SDN models?**

With the use of network overlays, NFV could also unify the two models of SDN infrastructure -- centralized and distributed. If connectivity control and application component or user isolation are managed by the network overlay, then the physical-network mission of SDN can be more constrained to traffic management. If SDN manages aggregated routes more than individual application flows, it could be more scalable. Remember that the most commonly referenced SDN applications today -- data center LANs and Google's SDN IP core network -- are more route-driven than flow-driven. Unification of the SDN model might also make it easier to sort out SDN implementations. The lower physical network SDN in this two-layer model might easily be created using revisions to existing protocols, which has already been proposed. While it doesn't offer the kind of application connectivity control some would like, that requirement would be met by the higher software virtual network layer or overlay.

# Virtualization Approach

Virtualization can apply to a range of system layers, including hardware-level virtualization, operating system level virtualization, and high-level language virtual machines. Hardware-level virtualization was pioneered on IBM mainframes in the 1970s, and then more recently Unix/RISC system vendors began with hardware-based partitioning capabilities before moving on to software-based partitioning (VMware, 2006).

A virtualization hypervisor comes in one of two forms, a bare-metal hypervisor, also known as Type 1, or a hosted hypervisor, also known as Type 2. (Siebert, 2015) Type 1 hypervisors support hardware virtualization, whereas Type 2 hypervisors support software virtualization. (Ruest, 2015) Bare-metal virtualization means the hypervisor has direct access to hardware resources, which results in better performance, scalability and stability.  Bare-metal virtualization is well suited for enterprise data centers, because it usually comes with advanced features for resource management, high availability and security. Admins can centrally manage this kind of virtualization hypervisor, which is critical when you have many hosts in your virtual infrastructure. The most popular bare-metal virtualization hypervisors are VMware ESX and ESXi, Microsoft Hyper-V and Citrix Systems XenServer (Siebert, 2015)

In the implementation of NFV only type 1 hypervisor is considered and is referred to as a network hypervisor. A program that provides an abstraction layer for network hardware.  Network hypervisors allow network engineers to create virtual networks that are completely decoupled and independent from the underlying physical network. (Rouse, 2015)

**Virtualization Targets**

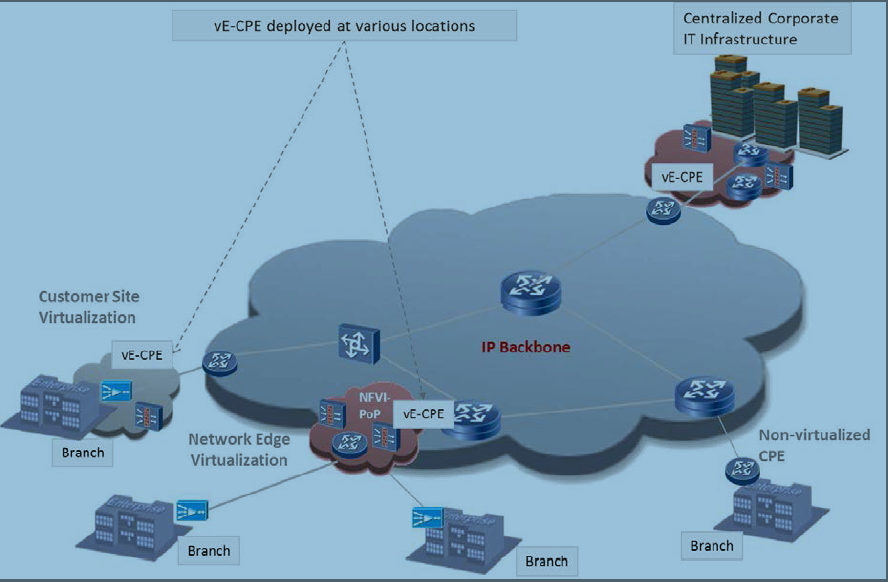
There are a number of Network Functions typically deployed today with Enterprise networks as dedicated hardware infrastructure where it may, in the future be appropriate for a Service Provider to deliver on VNFaaS basis to the Enterprise.

These Enterprise network functions include Enterprise Access Router(AR)/ Enterprise CPE, Provider Edge Router(PE), Enterprise Firewall(FW), Enterprise NG-FW(NG-FW), Enterprise WAN optimization Controller(WOC), Deep Packet Inspection(DPI), Network Performance Monitoring, Intrusion Prevention System(IPS) and other Security appliances (NFV, 2013).

Virtualization of the enterprise includes virtualization of the CPE functions (vE-CPE) in the service provider cloud. Vertualization of the PE functions (vPE) where the virtual network services functions and core-facing PE functions cab be executed in the service provider cloude. The two steps are independent and may be deployed separately. PE routers are typically shared by a high number of customers, whereas a CPE router is used exclusively by a single customer. Thus, economics of scale that can be gained from CPE virtualization are significantly greater compared to PE virtualization. It is likely, that virtualization of the CPE will take place first, providing the largest benefit for both the enterprise users and the Service Providers. Virtualization of the PE may be done at a later stage to complete the transition to a fully Virtualized NFV solution. In some architectures, the vE-CPE and vPE may be controlled by a centralized following the SDN architecture principles and standards(e.g. OpenFlow). The Service Provider is responsible for deploying, configuring, updating and managing the operation of the VNF instance to provide the expected service level (SLA) for subscribers to the VNFaaS (NFV, 2013).

## Virtualization of the CPE (vE-CPE)

The vE-CPE enhances the enterprise network by replacing appliances with NFV compliant Virtualised solutions located at either the enterprise cloud or the operator of the NFV framework. Services provided by the vE-CPE may include a router providing QoS and other high-end services such L7 stateful firewall, intrusion detection and prevention and more. Application accelerators are also deployed either as standalone appliances or as router integrated services (NFV, 2013).



## Virtualization of the PE(vPE)

Virtualization of the core routers may not be feasible in the short term due to high throughput requirements, but the virtualization of the PE router is more likely with additional benefits of providing scalability of Provider provisioned virtual private network services through the dynamic resizing or allocation of virtual resources. The Provider Provisioned Virtual Private Network Service(PPVPNS) functions in the vPE include layer 3 IP VPNs, layer 2 VPLS, EVPN, pseudo-wire services and more (NFV, 2013).

## Coexistence of Virtualized and non-virtualized network functions

A combined virtualization model consists of providing virtual network services functions with or without CPE functions while maintaining the core facing PE functions on the PE.

Operators already today share infrastructure resources when providing services to multiple users. Functions shared within this concept may remain non-Virtualized. The communication with virtual network functions shall be based on standardized interfaces (NFV, 2013).

# Implementation of VMware vSphere

## Overview

vSphere is a software suite that comes under data center product and ESXi is a hypervisor installed on a physical machine. vSphere Client is installed on laptop or desktop PC and is used to access ESXi Server to install and manage virtual machines on ESXi server. vCenter server is installed as virtual machine on top of ESXi server. vCenter server is a vSphere component which is mostly used in large environment where there are many ESXi server and dozens of virtual machines. The vCenter server is also accessed by vSphere client for management purpose. So, vSphere client is used to access ESXi server directly in small environment. In larger environment, vSphere client is used again to access vCenter server which ultimately manages ESXi server (Bipin, 2012).

**What is vSphere?**

**vSphere is a software suite that comes under data center product. vSphere is like Microsoft Office suite which has many software like MS Office, MS Excel, MS Access and so on. Like Microsoft Office, vSphere is also a software suite that has many software components like vCenter, ESXi, vSphere client and so on. So, the combination of all these software components is vSphere. vSphere is not a particular software that you can install and use, “it is just a package name which has other sub components”** (Bipin, 2012)**.**

**What is ESX?**

The ESX host now has an increasing number of APIs that other organizations can plug into projects like VMsafe, vNetwork, and vStorage, which allow companies like McAfee, Cisco, and EMC to plug into ESX. ESX carries out one aspect of an operating system, which is to be the arbitrator to hardware. ESX owns the physical hardware and sits at ring 0, the most privileged ring that exists in most x86 processors. Fundamentally, if any process needs access to hardware resources (CPU, disk, network, memory, and so on), it must liaise with ESX’s VMkernel for that resource (Laverick, 2010).

The only thing ESX executes is the virtual machine, sometimes referred to as the virtual machine monitor. ESX is what is called a hypervisor. The hypervisor ESX’s VMkernel, resides between the hardware and the virtual machine. Its primary task is to run virtual machines, as well as carry out other important ancillary tasks such as monitoring the performance of the processes it manages. This VMkernel is incredibly slim, leaving less of a “surface” area for vulnerabilities that could make it unstable or open to attack by intruders. It’s a good indication of how ESX has become the de facto platform in the corporate datacenter for running virtual machines (Laverick, 2010).

**Key features and capabilities of ESXi 4**

ESXi 4 offers a number of advantages, including the following, reduced patch burden, restricted access, rapid provisioning, greater reliability and hardware monitoring. Reduced patch burden addresses security vulnerabilities in the Service Console and making it more secure. Restricted access in ESX allows an ESX host to be managed only via the ILO (often referred to as the local console) or vCenter. This effectively makes users in Active Directory the only valid centralized account database for managing the ESXi host (Laverick, 2010).

Rapid provisioning, rather than requiring you to install the ESX product to disks or invest some time in automating the ESX installation, ESXi 4 simply does away with the installation process, and eliminates many of the operator decisions required to deploy ESX. However, your networking and vSwitches still must be created, although this process could be automated with remote CLI scripts. ESX provides greater reliability by being integrated with solid state components rather than on hard drives. No longer need to worry about the failure rate of physical disks, unless you are creating local VMFS volumes. ESX integrates hardware monitoring is into the system. Therefore the is no need to manually install any hardware monitoring agents (Laverick, 2010).

**Prior to ESX Installation**

Before an installation of ESX, the physical configuration of the server must be checked. To be more specific, run updates against main system BIOS and firmware upgrades for devices such as host bus adapters (HBAs). Additionally, its recommend that to check BIOS settings to make sure additional features are enabled. For example, for VMware fault tolerance to work, it must have virtualization hardware assist options like Intel Virtualization Technology (Intel VT) or AMD Virtualization (AMD-V) enabled (Laverick, 2010).

**Installation of ESXi 4**

The installation procedure is straightforward. Step 1, boot to the ESXi 4 CD (refer to figure E1). Step 2, choose the Install option by pressing the ENTER key (refer to figure E2). (You also have the Repair option, which reinstalls the ESXi 4 software without affecting your settings.) Step 3, press F11 to accept the EULA. Step 4, select a disk on which to install ESXi 4 (refer to figure E3). If you have a memory stick connected to the system, you can select it as a target for installing ESXi 4. Step 5, press F11 again to confirm you wish to start the installation. Step 6, at the end of the installation when prompted, press ENTER to reboot the server (Laverick, 2010).

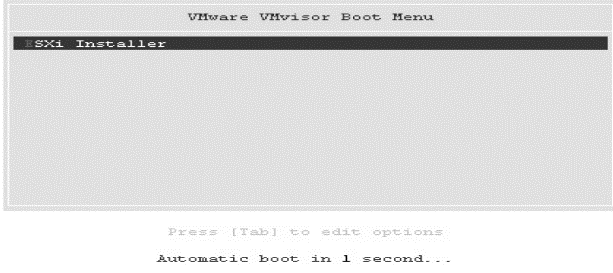


Figure E1: Step 1

Figure E3: Step 4

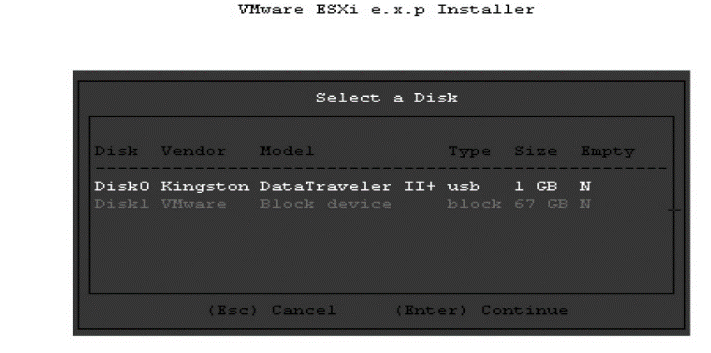
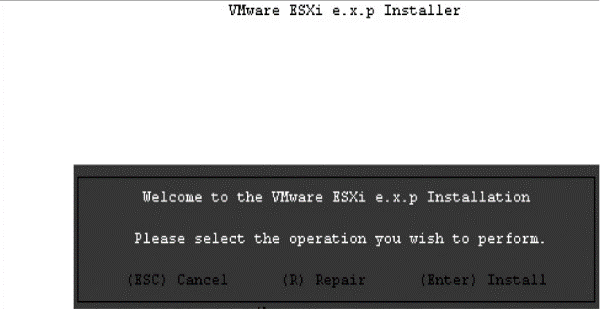


Figure E2: Step 2

**Setting the Root Password**

Step 1, from the initial setup screen, press F2 to choose Customize System (refer to figure F1).. Step 2, under Customize System, ensure Configure Root Password is selected, and then press ENTER to set a root password (refer to figure F2). Step 3, the root password is currently blank. Press ENTER to accept the Old Password field (refer to figure F3).. Step 4, in the New Password and Confirm Password fields, type your preferred password (Laverick, 2010).

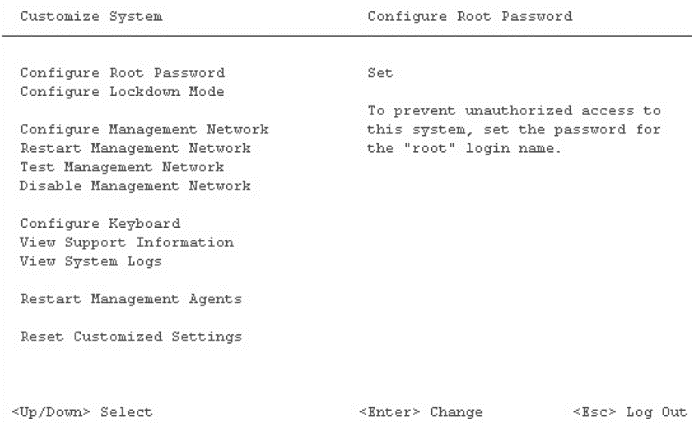
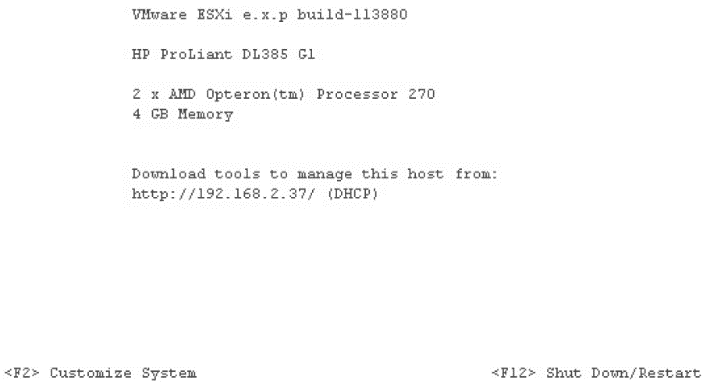


Figure F1: Step 1 Figure F2: Step 2

Figure F3: Step 3



**Configuring Management Network**

The Configure Management Network option allows you to change five main settings such as VLAN settings, IP settings, DNS configuration, custom DNS suffixes and network adapters (Laverick, 2010).

**Setting-up Network Adapter**

ESXi 4 by default selects the first NIC for which it has a valid driver as your management console network connection. This might not be the correct NIC for the network or VLAN from which you carry out management tasks. To set up network adapter, follow these 3 steps: Step 1, from the Customize Management Network screen, select Network Adapters and press ENTER. The Network Adapters screen lists the current vmnic devices used and the MAC addresses of each of the NICs. The following example shows four NICs. These are two dual-port cards. My server has one dual-port on-board Broadcom NIC (00:15) and an external dual-port Intel card (00:02) (refer to figure G2). Step 2, use the cursor keys and the spacebar to select the NIC(s) you would like to use. Finally step 3, press ENTER to save your changes (Laverick, 2010).

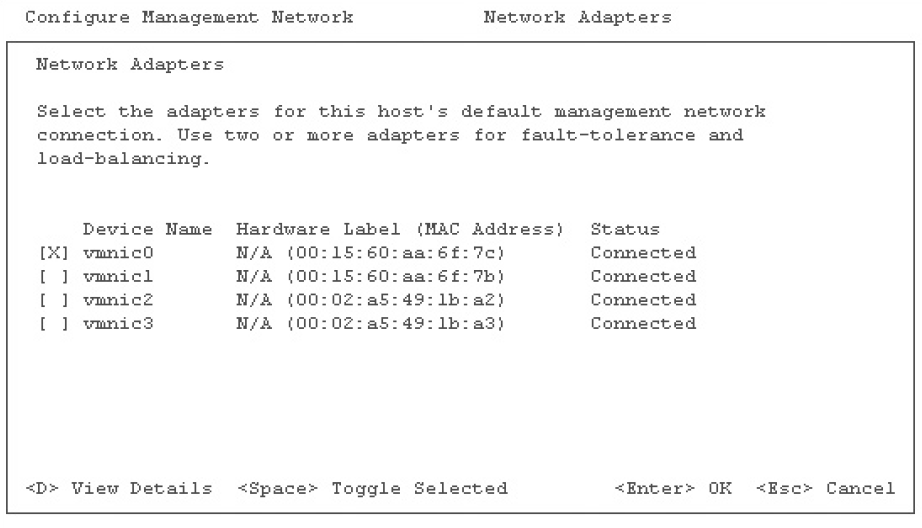


Figure G1: Step 1

**Configuring IP and DNS**

To configure IP and DNS settings, follow these steps. Step 1, from the Customize Management Network screen, select IP Configuration and press ENTER. Step 2, cursor down to select “Use the following IP address and network settings” and press the spacebar. Step 3, modify the IP, subnet mask, and default gateway as appropriate for your network ranges (refer to figure H1). Step 4, press ENTER to confirm your changes. Step 5, select DNS Configuration and press ENTER. Step 6, modify the fields to set your primary and secondary DNS configuration. Currently, it is not possible to set a tertiary DNS from the local console. Additionally, ESXi 4 must be completely statically configured or use DHCP. Unlike Microsoft Windows, an ESXi host cannot be a DHCP client for its IP, subnet, and default gateway addresses and statically configured for DNS (refer to figure H2). Step 7, press ENTER to save your changes. Step 8, select Custom DNS Suffixes and press ENTER (Laverick, 2010).

Step 9, type the appropriate DNS suffix search domain for your ESX host (refer to figure H3). Finally step 10, press ESC to save your changes, and choose Yes at Configure Management Network: Confirm screen to restart the ESXi 4 network layer. Whenever you make major changes to the ESXi management network settings, you will always be prompted to restart the network layer to put settings in effect (refer to figure H4).

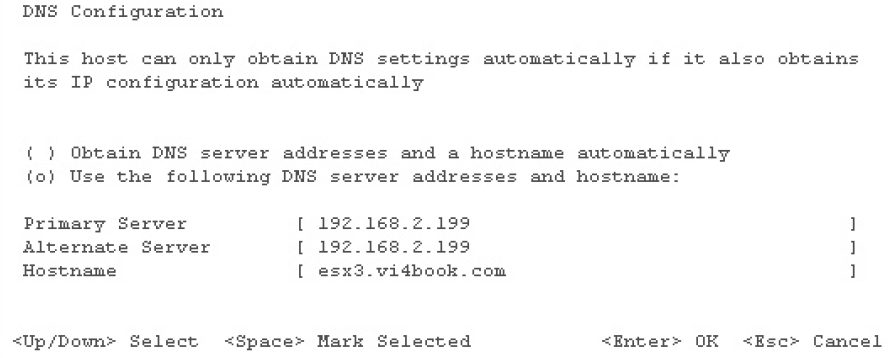
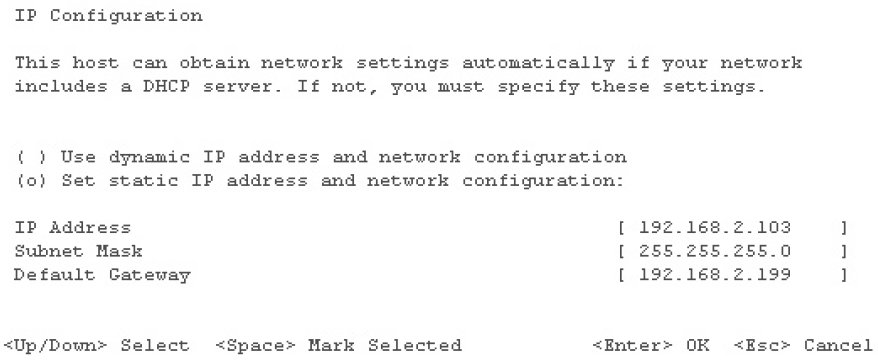


Figure H1: Step 3 Figure H2: Step 6

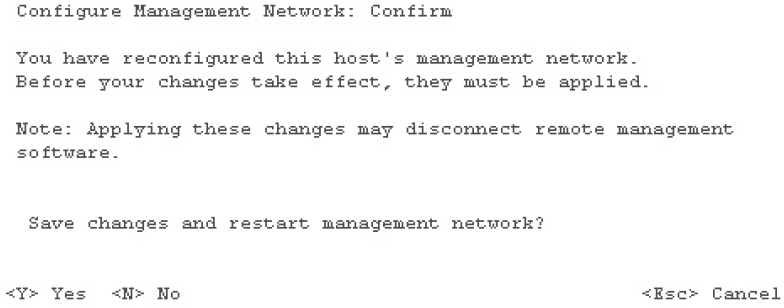
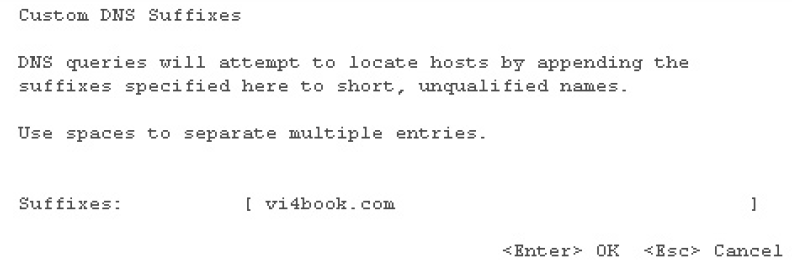


Figure H3: Step 9 Figure H4: Step 10

The other options on the Customize System screen allow you to configure your system as follows. Configure lockdown mode, restart management network and disable management network.Configure Lockdown Mode prevents access to the ESXi 4 host directly via the vSphere Client. Normally, you enable this when you add an ESXi host to vCenter. This effectively makes vCenter the only GUI management system available. Restart management network restarts the management network, which normally happens whenever you change any network settings. Disable management network toggles the option from Disable Management Network to Enable Management Network and will prevent anyone from managing the ESXi 4 host via the vSphere Client (Laverick, 2010).

**Installation of vCenter**

vCenter is a management application that allows you to manage several ESX hosts in a single window. vCenter is beyond an entry-level management and has key features that are available only if you have vCenter and the appropriate add-ons such as distributed virtual switches, microsoft user accounts, templates and template management, cold and hot migrations (VMware VMotion and Storage VMotion), VMware Distributed Resource Scheduler (DRS), VMware High Availability (HA) and VMware Fault Tolerance (FT) (Laverick, 2010).

**vCenter Web Access**

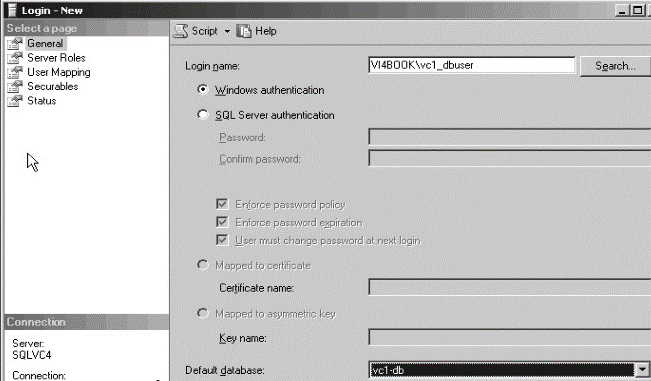
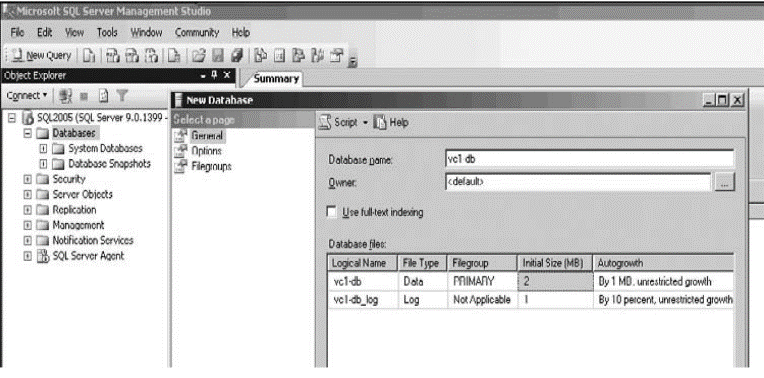
Web Access is another service that can be installed alongside vCenter. It offers a basic user interface that allows operator-style access to virtual machines, without the need for the vSphere Client. The nice thing about Web Access is that all the communication is encrypted with SSL certificates on firewall-friendly 443 TCP ports (Laverick, 2010).

**Configure Your Database for vCenter**

Before you run the installation CD for vCenter, you must first set up your database back end. The vCenter database expands incrementally over time, as it continually collects performance data. Care should be taken to make sure that the database has enough free disk space into which it can grow. vCenter 4, VMware now supports both Windows authentication and SQL Server authentication. If you are installing SQL Server for the first time, you can set the system to support both SQL Server and Windows authentication. If you are using an existing SQL Server implementation, you can change its authentication type using the Management Studio console (Laverick, 2010).

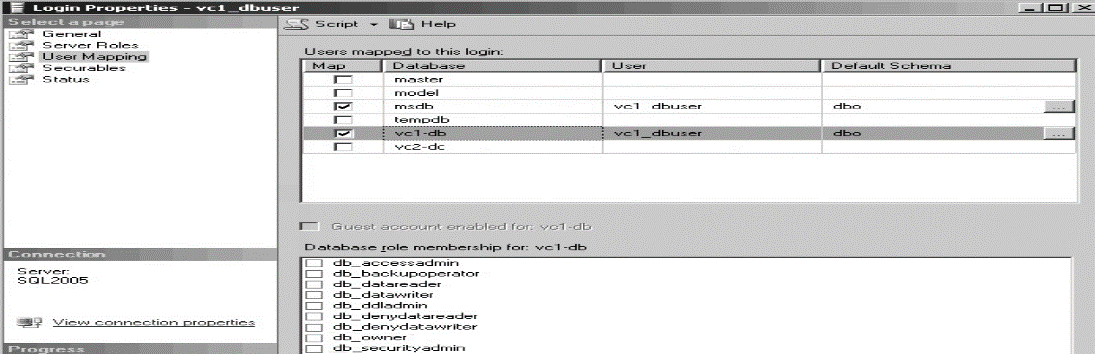
**Create the Database and Setting Permissions**

To create the database using SQL Server 2005 and set the permissions, follow these steps. Step 1, create a local user with Active Directory, called, for example, vc1\_dbuser. Additionally, make this user a member of the local Administrators group on your vCenter server. Step 2, open Microsoft SQL Server Management Studio. Step 3, log in to SQL Server. Step 4, right-click Databases and choose New Database. Step 5, in the New Database dialog box, enter a database name, such as vc1-db. Also, it’s a good idea to store the database somewhere other than on the C: drive (as you know, the C: drive is quite possibly the worst location to hold any data that is important to you). Step 6, click OK to create the database (refer to figure M1). Step 7, expand Security, right-click Logins, and choose New Login. Step 8, in the Login - New dialog box, click the Search button and locate your database user account. Select the Windows authentication radio button. Set the default database to the new database you just created (refer to figure M2). Step 9, click User Mapping on the left side of the dialog box. Step 10, select the MSDB and vc1-db databases. Enable the permission db\_owner. Click OK to create the new login (refer to figure M3). Step 11 click OK, and confirm the password again (Laverick, 2010).



**Figure M1: Step 6 Figure M2: Step 8**

**Figure M3: Step 10**



**Configuring a DSN Connection**

It is recommended to set up Data Source Name (DSN) before you begin and to resolve any database issues before you crank up the vCenter installer engine. Install the SQL Server Native Client to your vCenter server for this part to work. Step 1, log in to the vCenter server with the database user account you created in Active

Directory. Step 2, open the ODBC Data Source Administrator from Administrative Tools on the Start menu. Step 3, in the ODBC Data Source Administrator, choose the System DSN tab. Step 4, click the Add button. Step 5, from the end of the list, choose SQL Server Native Client, and then select Finish. Step 6, in the Create a New Data Source to SQL Server dialog box, type VMware Virtual Center in the Name field. Step 7, from the Server drop-down list, select your SQL Server 2005 server, and then click Next (refer to figure L1). Step 8, select the “With Integrated Windows authentication” option, and then click Next (refer to figure L2). Step 9, select “Change the default database to” and select the database you created earlier (refer to figure L3). Step 10, click Next, and then click Finish (Laverick, 2010).

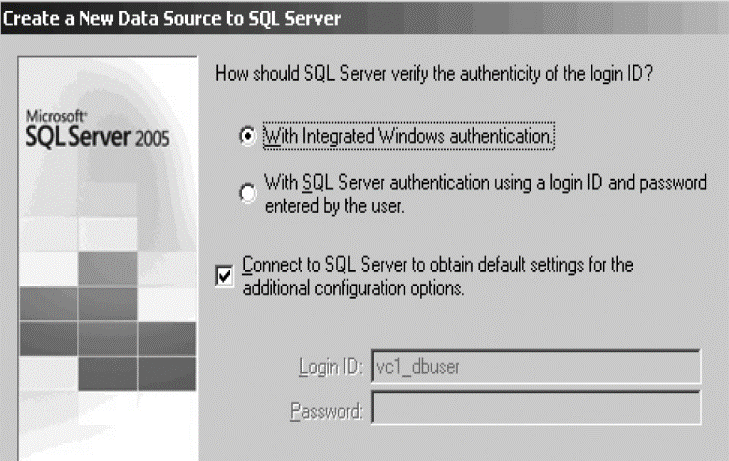
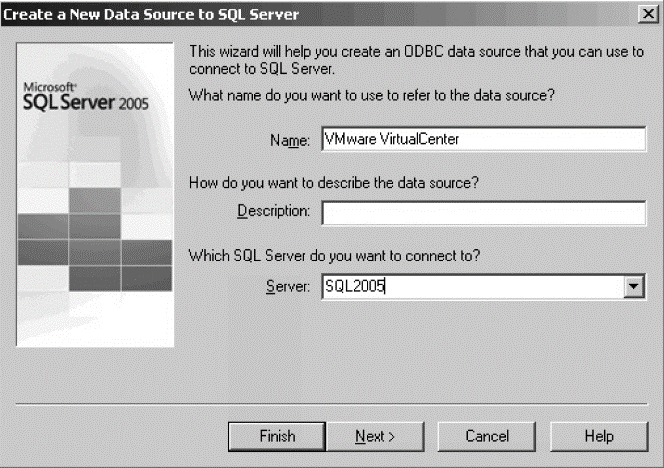
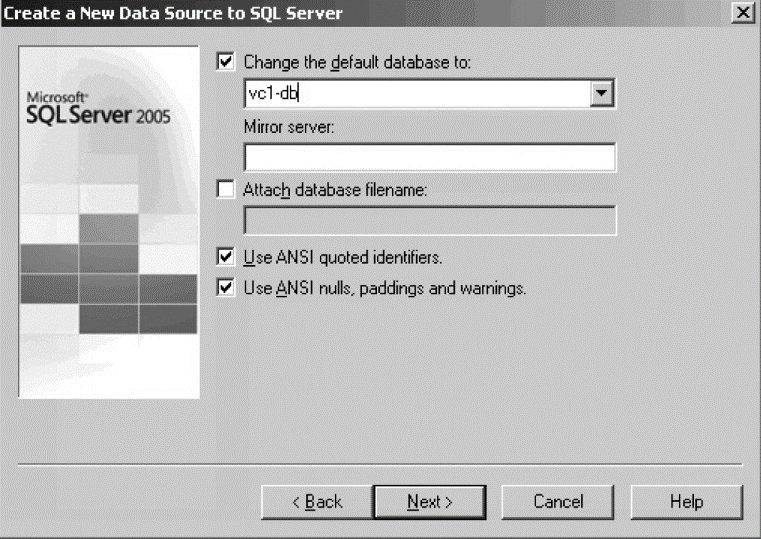


Figure L1: Step 7 Figure L2: Step 8

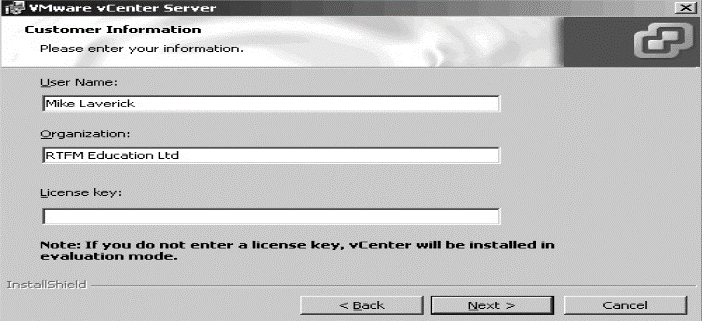
Figure L3: Step 9



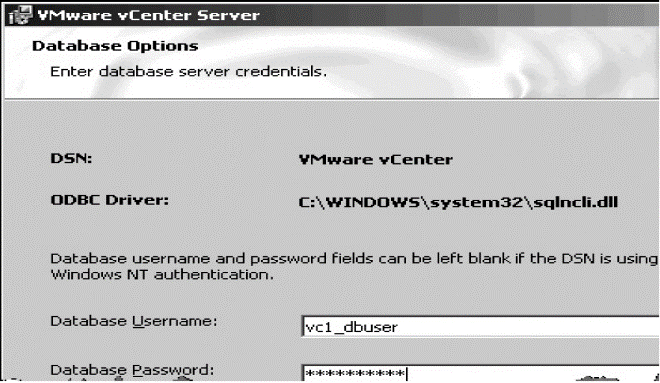
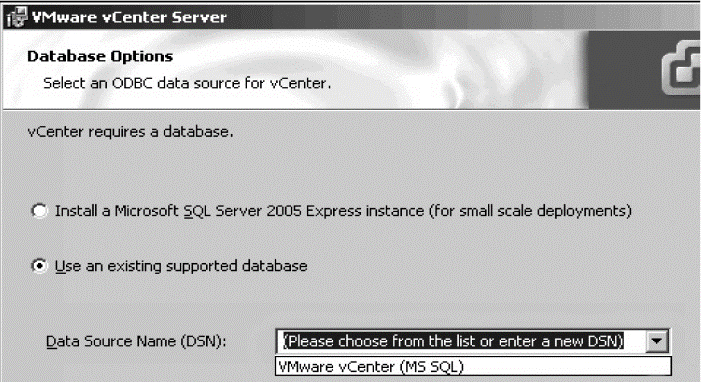
**Setting Up vCenter**

After you’ve configured your database for vCenter, the next step is to install the vCenter server. Then you can add ESX Classic and/or ESXi hosts. Step 1, insert the vCenter CD or connect to its ISO file. If the CD fails to autorun, then you can double-click the autorun.exe file. Step 2, select vCenter Server, and then move through the welcome screen and accept the EULA. Step 3, in the Customer Information dialog box, enter your name and organization, and if you have it, your vCenter license number. Step 4, in the Database Options dialog box, select the option to “Use an existing supported database.” From the drop-down list, select the DSN you created earlier. Step 5, Type in the username and password used by the DSN to access the vCenter database. Step 6, in the vCenter Server Service dialog box, accept the default account for running vCenter. (If you are using SQL Server authentication, the Use SYSTEM Account option in this dialog box will be available, and you can enter an account name and password.). Step 7, accept the default path for the installation, C:\Program Files\VMware\Infrastructure. Step 8, as this is the first vCenter 4 installation in this example, in the vCenter Linked Mode Options dialog box, select “Create a standalone VMware vCenter Server instance.” Step 9, accept the default port numbers for the vCenter services. Step 10, start the installation by clicking the Install button. (The installation will take some time to complete.) If you have a PC with the vSphere Client already installed. There is no need to install the client to the vCenter desktop. The vSphere Client can be used directly at an ESX host (log in in with root and a password), or it can be opened against a vCenter server using the built-in administrator account of the local vCenter server or a domain (Laverick, 2010).

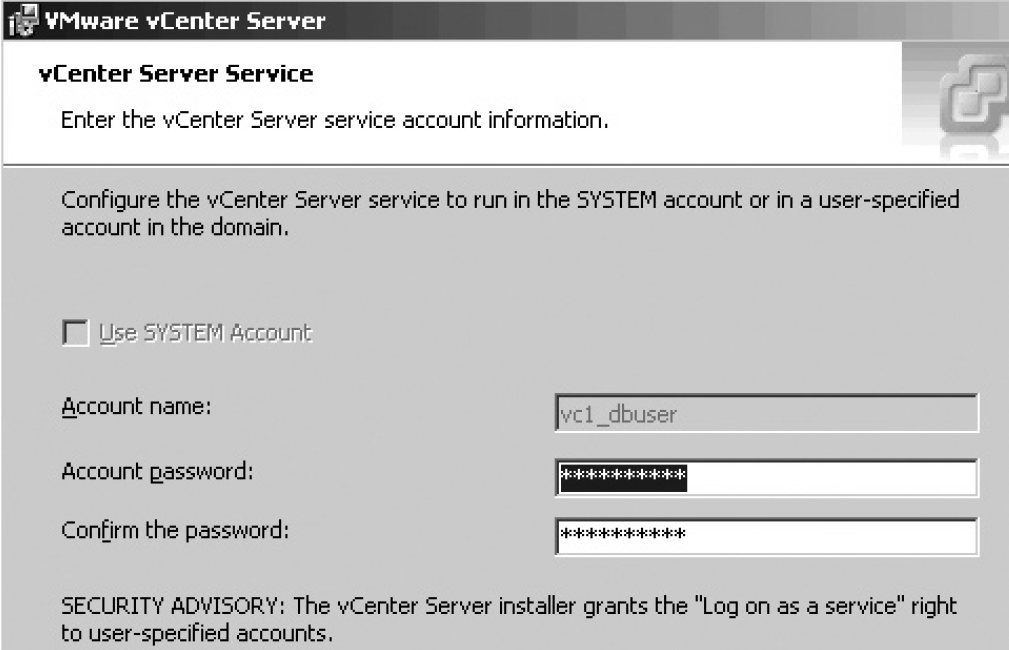
Figure N1: Step 1 Figure N2: Step 3



**Figure N3: Step 4 Figure N4: Step 5**



**Figure N5: Step 6**



**vCenter Converter**

**What is VMware vCenter Converter?**

VMware vCenter Converter is a highly robust and scalable enterprise-class migration tool that automates the process of creating VMware virtual machines from physical machines, other virtual machine formats and third-party image formats. Through an intuitive wizard-driven interface and a centralized management console, VMware vCenter Converter can quickly and reliably convert multiple local and remote physical machines without any disruptions or downtime (VMware, 2009).

**Key features and capabilities**

VMware vCenter Converter reduces the amount of time IT professionals spend on migrating to a virtual infrastructure by enabling fast, reliable and non-disruptive conversions from physical to virtual machines, and from older virtual machines to newer formats. Convert dozens or even hundreds of physical machines VMware virtual machines during a server consolidation project (VMware, 2009).

VMware vCenter Converter automates this process and reduces the time and effort required to manage these multiple large-scale conversions. Convert between multiple types or generations of VMware virtual machines. For example, virtual machines created by VMware Server in a lab environment can be directly migrated to VMware vSphere™ 4 in production environments. Create virtual machine clones of physical or virtual machines and archive them to a file server at an alternative site for disaster recovery. In the event of a disaster, the cloned virtual machines can be deployed on VMware Server or VMware vSphere to provide end users with access to their files or applications (VMware, 2009).

**How Does VMware vCenter Converter Work?**

VMware vCenter Converter is managed through a simple, task-based user interface that enables users to convert physical machines, third-party disk image formats or VMware Consolidated Backup images of virtual machines to VMware virtual machines in three easy steps: Step 1, specify the source physical server, virtual machine or third-party format to convert. Step 2, specify the destination format, virtual machine name, and location for the new virtual machine to be created. Step 3, create/convert virtual machines to a new destination and configure them. VMware vCenter Converter achieves faster conversion speeds using sector-based copying (vs. file-level copying in other products). VMware vCenter Converter first takes a snapshot of the source machine before migrating the data, resulting in fewer failed conversions and no downtime on the source server (VMware, 2009).

**VMware vCloud Air Disaster Recovery**

VMware vCloud Air Disaster Recovery is a recovery-as-a-service (RaaS) solution that introduces native cloud-based disaster recovery capabilities for VMware vSphere® virtual environments. Built on VMware’s hypervisor-based replication engine, vSphere® Replication™, vCloud Air Disaster Recovery includes; self-service disaster recovery protection for virtual machines, recovery point objectives (RPO) from 15 minutes to 24 hours, readily facilitates failover, failback and planned migration as needed, retention of multiple recovery points up to 24 point-in-time instances, Elastic cloud compute and storage capacity. Support for offline data seeding. Private leased line network option and flexible failover testing (VMware, 2015).

# STANDARD NERTWORKING

VMware vSphere is the leading server virtualization platform with consistent management for virtual data centers.

Deliver business value from day one with powerful server virtualization, breakthrough availability, safe automated management and intelligent operational insight that adapts to your environment. Automate workload placement and resource optimization based on preset customizable templates

## vSphere Standard Switches

You can create abstracted network devices called vSphere standard switches. A standard switch can bridge traffic internally between virtual machines in the same port group and link to external networks.

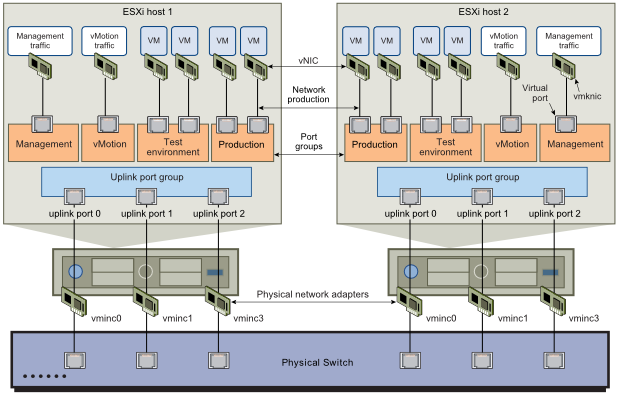
You can use standard switches to combine the bandwidth of multiple network adapters and balance communications traffic among them. You can also configure a standard switch to handle physical NIC failover.

A vSphere standard switch models a physical Ethernet switch. The default number of logical ports for a standard switch is 120. You can connect one network adapter of a virtual machine to each port. Each uplink adapter associated with a standard switch uses one port. Each logical port on the standard switch is a member of a single port group. Each standard switch can also have one or more port groups assigned to it (Laverick, 2010).

## Standard Port Groups

Port groups aggregate multiple ports under a common configuration and provide a stable anchor point for virtual machines connecting to labeled networks.

**vSphere Standard Switch Network**



* Virtual machine group
* Service Console port group
* VMKernel port group (for VMotion, VMware FT logging, and IP storage)

Each port group is identified by a network label, which is unique to the current host. Network labels are used to make virtual machine configuration portable across hosts. All port groups in a datacenter that are physically connected to the same network (in the sense that each can receive broadcasts from the others) are given the same label. Conversely, if two port groups cannot receive broadcasts from each other, they have distinct labels (Laverick, 2010).

A VLAN ID, which restricts port group traffic to a logical Ethernet segment within the physical network, is optional. For a port group to reach port groups located on other VLANs, the VLAN ID must be set to 4095. If you use VLAN IDs, you must change the port group labels and VLAN IDs together so that the labels properly represent connectivity (Laverick, 2010).

**Port Group Configuration for Virtual Machines**

You can add or modify a virtual machine port group from the vSphere Client.

The vSphere Client Add Network wizard guides you through the tasks to create a virtual network to which virtual machines can connect, including creating a vSphere standard switch and configuring settings for a network label.

When you set up virtual machine networks, consider whether you want to migrate the virtual machines in the network between hosts. If so, be sure that both hosts are in the same broadcast domain—that is, the same Layer 2 subnet.

ESXidoes not support virtual machine migration between hosts in different broadcast domains because the migrated virtual machine might require systems and resources that it would no longer have access to in the new network. Even if your network configuration is set up as a high-availability environment or includes intelligent switches that can resolve the virtual machine’s needs across different networks, you might experience lag times as the Address Resolution Protocol (ARP) table updates and resumes network traffic for the virtual machines.

Virtual machines reach physical networks through uplink adapters. A vSphere standard switch can transfer data to external networks only when one or more network adapters are attached to it. When two or more adapters are attached to a single standard switch, they are transparently teamed (VMware, 2015).

### Add a Virtual Machine Port Group

Virtual machine port groups provide networking for virtual machines. Procedure;

1. Log in to the vSphere Client and select the Hosts and Clusters inventory view.
2. Select the host in the inventory pane.
3. On the host Configuration tab, click Networking.
4. Select the vSphere Standard Switch view. Standard switches appear in an overview that includes a detailed layout.
5. On the right side of the page, click Add Networking.
6. Accept the default connection type, Virtual Machines, and click Next.
7. Select Create a vSphere standard switch or one of the listed exisng standard switches and the associated physical adapters to use for this port group. You can create a standard switch with or without Ethernet adapters. If you create a standard switch without physical network adapters, all traffic on that switch is confined to that switch. No other hosts on the physical network or virtual machines on other standard switches can send or receive traffic over this standard switch. You might create a standard switch without physical network adapters if you want a group of virtual machines to be able to communicate with each other, but not with other hosts or with virtual machines outside the group.
8. Click Next.
9. In the Port Group Properties group, enter a network label that identifies the port group that you are creating. Use network labels to identify migration-compatible connections common to two or more hosts.
10. (Optional) If you are using a VLAN, for VLAN ID, enter a number between 1 and 4094. If you enter 0 or leave the option blank, the port group detects only untagged (non-VLAN) traffic. If you enter 4095, the port group can detect traffic on any VLAN while leaving the VLAN tags intact.
11. Click Next.
12. After you determine that the switch is configured correctly, click Finish.

## VMkernel Networking Configuration

A VMkernel networking interface provides network connectivity for the host as well as handling VMware vMotion, IP storage, and Fault Tolerance. Moving a virtual machine from one host to another is called migration. Using vMotion, you can migrate powered on virtual machines with no downtime. Your VMkernel networking stack must be set up properly to accommodate vMotion. IP storage refers to any form of storage that uses TCP/IP network ESXi. Because these storage types are network based, they can use the same VMkernel interface and port group.

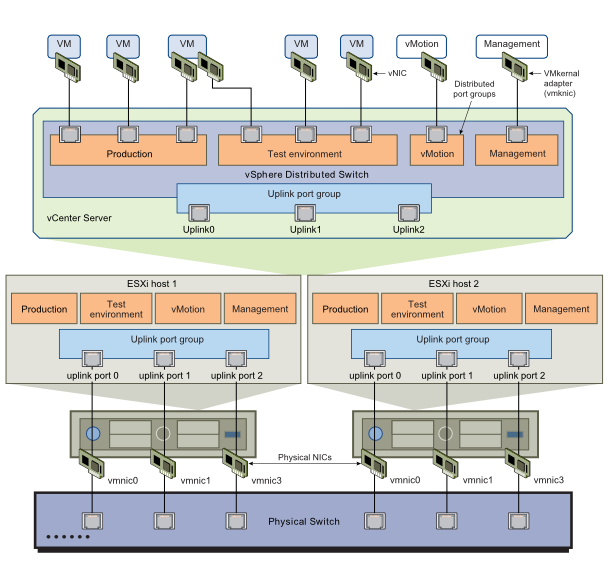
**Setting up Networking with vSphere Distributed Switches**

With vSphere distributed switches you can set up and configure networking in a vSphere environment (VMware, 2015).

**vSphere Distributed Switch Architecture**

A vSphere distributed switch functions as a single switch across all associated hosts. This enables you to set network configurations that span across all member hosts, and allows virtual machines to maintain consistent network configuration as they migrate across multiple hosts (VMware, 2015).

**vSphere Distributed Switch Network**



5Like a vSphere standard switch, each vSphere distributed switch is a network hub that virtual machines can use. A distributed switch can forward traffic internally between virtual machines or link to an external network by connecting to physical Ethernet adapters, also known as uplink adapters. (VMware, 2015)

**Summary of Distributed Virtual Networking**

VMware vSphere Distributed Switch (VDS) provides a centralized interface from which you can configure, monitor and administer virtual machine access switching for the entire data center. The VDS provides:

* Simplified virtual machine network configuration
* Enhanced network monitoring and troubleshooting capabilities
* Support for advanced VMware vSphere networking features

**Simplified Virtual Machine Network Configuration**

Use these VDS features to streamline provisioning, administration and monitoring of virtual networking across multiple hosts:

* Central control of virtual switch port configuration, portgroup naming, filter settings, and others
* Link Aggregation Control Protocol (LACP) that negotiates and automatically configures link aggregation between vSphere hosts and the access layer physical switch
* Network health-check capabilities to verify vSphere to physical network configuration

**Enhanced Network Monitoring and Troubleshooting Capabilities**

* The VDS provides monitoring and troubleshooting capabilities:
* Support for RSPAN and ERSPAN protocols for remote network analysis
* IPFIX Netflow version 10
* SNMPv3 support
* Rollback and recovery for patching and updating the network configuration
* Templates to enable backup and restore for virtual networking configuration
* Network-based coredump (Netdump) to debug hosts without local storage

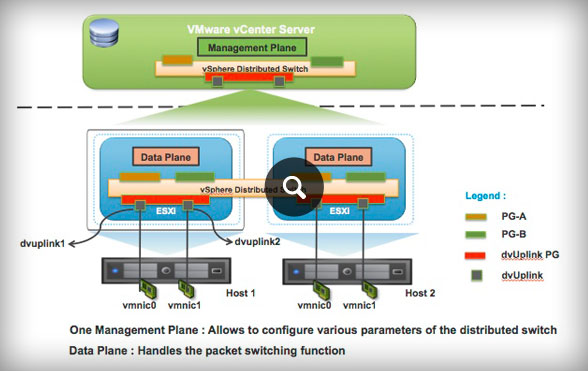
**Support for Advanced vSphere Networking Features**

**The VDS supplies the building blocks for many networking capabilities in the vSphere environment:**

* Provides the core element for VMware vSphere Network I/O Control (NIOC)
* Maintains network runtime state for virtual machines as they move across multiple hosts, enabling inline monitoring and centralized firewall services
* Supports Single Root I/O Virtualization (SR-IOV) to enable low-latency and high-I/O workloads
* Contains a BPDU filter to prevent virtual machines from sending BPDUs to the physical switch

**Technical Details**

The VDS extends the features and capabilities of virtual networks while simplifying provisioning and the ongoing configuration, monitoring and management process. vSphere network switches can be divided into two logical sections: the data plane and the management plane. The data plane implements the packet switching, filtering, tagging and so on. The management plane is the control structure used by the operator to configure data plane functionality. Each vSphere Standard Switch (VSS) contains both data and management planes, and the administrator configures and maintains each switch individually. The VDS eases this management burden by treating the network as an aggregated resource. Individual host-level virtual switches are abstracted into one large VDS spanning multiple hosts at the data center-level. In this design, the data plane remains local to each VDS but the management plane is centralized.Each VMware vCenter Server instance can support up to 128 VDSs; each VDS can manage up to 500 hosts.



* **Distributed Virtual Port Groups (DV Port Groups)** — Port groups that specify port configuration options for each member port
* **Distributed Virtual Uplinks (dvUplinks**) — dvUplinks provide a level of abstraction for the physical NICs (vmnics) on each host
* **Private VLANs (PVLANs)** — PVLAN support enables broader compatibility with existing networking environments using the technology
* Network vMotion — Simplifies monitoring and troubleshooting by tracking the networking state (such as counters and port statistics) of each virtual machine as it moves from host to host on a VDS
* **Bi-directional Traffic Shaping** — Applies traffic shaping policies on DV port group definitions, defined by average bandwidth, peak bandwidth and burst size

**vSphere Storage**

*vSphere Storage* describes storage options available to VMware® ESXi and explains how to configure your ESXi system so that it can use and manage different types of storage. In addition, *vSphere Storage* explicitly concentrates on Fibre Channel and iSCSI storage area networks (SANs) as storage options and discusses specifics of using ESXi in Fibre Channel and iSCSI environments.

**Updated Information**

This *vSphere Storage* is updated with each release of the product or when necessary.

This table provides the update history of *vSphere Storage*.

|  |  |
| --- | --- |
| **Revision** | **Description** |
|  |  |
| EN-001097-01 | Minor updates. |
| EN-001097-00 | Initial release. |

## Introduction to Storage

This introduction describes available storage options for ESXi and explains how to configure your host so that it can use and manage different types of storage.

### Storage Virtualization

ESXi provides host-level storage virtualization, which logically abstracts the physical storage layer from virtual machines.

An ESXi virtual machine uses a virtual disk to store its operating system, program files, and other data associated with its activities. A virtual disk is a large physical file, or a set of files, that can be copied, moved, archived, and backed up as easily as any other file. You can configure virtual machines with multiple virtual disks.

To access virtual disks, a virtual machine uses virtual SCSI controllers. These virtual controllers include BusLogic Parallel, LSI Logic Parallel, LSI Logic SAS, and VMware Paravirtual. These controllers are the only types of SCSI controllers that a virtual machine can see and access. Each virtual disk resides on a vSphere Virtual Machine File System (VMFS) datastore or an NFS-based datastore that are deployed on a physical storage. From the standpoint of the virtual machine, each virtual disk appears as if it were a SCSI drive connected to a SCSI controller. Whether the actual physical storage device is being accessed through parallel SCSI, iSCSI, network, Fibre Channel, or FCoE adapters on the host is transparent to the guest operating system and to applications running on the virtual machine. In addition to virtual disks, vSphere offers a mechanism called raw device mapping (RDM). RDM is useful when a guest operation system inside a virtual machine requires direct access to a storage device.

### Types of Physical Storage

The ESXi storage management process starts with storage space that your storage administrator preallocates on different storage systems.

**ESXi supports the following types of storage:**

|  |  |
| --- | --- |
| **Local Storage** | Stores virtual machine files on internal or directly connected external storage disks. |
| **Networked Storage** | Stores virtual machine files on external storage disks or arrays attached to your host through a direct connection or through a high-speed network. |

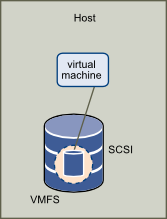
### Local Storage

Local storage can be internal hard disks located inside your ESXi host, or it can be external storage systems located outside and connected to the host directly through protocols such as SAS or SATA.

Local storage does not require a storage network to communicate with your host. You need a cable connected to the storage unit and, when required, a compatible HBA in your host.

The following illustration depicts a virtual machine using local SCSI storage.

**Local Storage**



In this example of a local storage topology, the host uses a single connection to a storage disk. On that disk, you can create a VMFS datastore, which you use to store virtual machine disk files.

Although this storage configuration is possible, it is not a recommended topology. Using single connections between storage arrays and hosts creates single points of failure (SPOF) that can cause interruptions when a connection becomes unreliable or fails.

ESXi supports a variety of internal or external local storage devices, including SCSI, IDE, SATA, USB, and SAS storage systems. Regardless of the type of storage you use, your host hides a physical storage layer from virtual machines.

### Networked Storage

Networked storage consists of external storage systems that your ESXi host uses to store virtual machine files remotely. Typically, the host accesses these systems over a high-speed storage network.

Networked storage devices are shared. Datastores on networked storage devices can be accessed by multiple hosts concurrently. ESXi supports the following networked storage technologies.

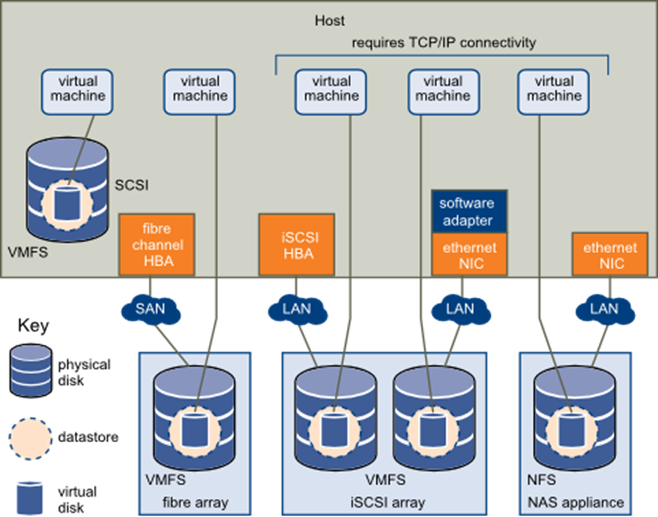
### How Virtual Machines Access Storage

When a virtual machine communicates with its virtual disk stored on a datastore, it issues SCSI commands. Because datastores can exist on various types of physical storage, these commands are encapsulated into other forms, depending on the protocol that the ESXi host uses to connect to a storage device.

ESXi supports Fibre Channel (FC), Internet SCSI (iSCSI), Fibre Channel over Ethernet (FCoE), and NFS protocols. Regardless of the type of storage device your host uses, the virtual disk always appears to the virtual machine as a mounted SCSI device. The virtual disk hides a physical storage layer from the virtual machine’s operating system. This allows you to run operating systems that are not certified for specific storage equipment, such as SAN, inside the virtual machine.

The following graphic depicts five virtual machines using different types of storage to illustrate the differences between each type.

**Virtual machines accessing different types of storage**



# Conclusion

The present document provides a view on virtualization-specific functional blocks and refence points among the functional blocks necessary within an operator’s network. The present document identifies the scope of ETSI NFV ISG and implements a virtualization using vSphere. The present document outlines the benefits of virtualization and how to go about achieving virtualization. Furthermore, the present document also highlights open issues that require further study that are key to achieving the NFV objectives of realizing carrier-grade virtual networks. Many of the reference points, technologies and components identified in the present document are available today. NFV solutions are assembled in a consistent and reusable manner, enabling economies of scale and faster innovation of network services.

Network function virtualization aims to reduce OpEx automation and scalability provided by implementing network function as virtual appliances. It allow all benefits of virtual and cloud computing including orchestration, hardware automation, independence etc. NFV requires standardization of reference points and interfaces to be able to mix match VNF from different sources.

The major goal of Network Function Virtualization (NFV) is to increase service agility while enabling better asset utilization.

**References**

Bipin. (2012, August 24). *Difference between vSphere, ESXi and vCenter*. Retrieved from MustBeGeek: http://www.mustbegeek.com/difference-between-vsphere-esxi-and-vcenter/

Conner, M. (2014, January 02). *ETSI NFV Management and Orchestration - An Overview.* Retrieved from http://www.ietf.org/proceedings/88/slides/slides-88-opsawg-6.pdf

Laverick, M. (2010). *VMware VSphere 4 Implementation.* Retrieved from http://d07vt3.googlecode.com/files/VMware%20VSphere%204%20Implementation.pdf

Margaret Chiosi, S. W. (2013, October 18). *Network Functions Virtualisation (NFV).* Retrieved from Network Functions Virtualisation – Update White Paper: https://portal.etsi.org/NFV/NFV\_White\_Paper2.pdf

NFV, E. G. (2013). *Network Functions Virtualization(NFV); Use Cases*. Retrieved from gsnfv001v010101p-131015124308-phpapp01.pdf

Rouse, M. (2015). *Network hypervisor*. Retrieved from Tech Target: http://searchsdn.techtarget.com/definition/Network-hypervisor

Ruest, D. (2015). *Hypervisors: Which is right for you?* Retrieved from Tech Target: http://searchservervirtualization.techtarget.com/tip/Hypervisors-Which-is-right-for-you

Siebert, E. (2015). *Understanding hosted and bare-metal virtualization hypervisor types*. Retrieved from Tech Target: http://searchservervirtualization.techtarget.com/tip/Understanding-hosted-and-bare-metal-virtualization-hypervisor-types

Torry Harris. (2009, October 22). *CLOUD COMPUTING .* Retrieved from http://www.thbs.com/downloads/Cloud-Computing-Overview.pdf

VMware. (2006). *Virtualization Overview*. Retrieved from https://www.vmware.com/pdf/virtualization.pdf

VMware. (2009). *VMware vCenter Converter.* Retrieved from http://www.vmware.com/files/pdf/converter\_datasheet.pdf

VMware. (2015, March). *VMware vCloud Air Disaster Recovery.* Retrieved from http://www.vmware.com/files/pdf/vcloud-air/VMware-vcloud-air-Disaster-Recovery-DS.pdf