A Report on the Project

SD-WAN service analysis, solution and its applications

Submitted by

Sarabjit Singh

In partial fulfillment for the award of the degree

Master of Science in Internetworking

(From University of Alberta)

Under the guidance of

Juned Noonari

September 2017 – March 2018

ABSTRACT

Software-Defined WAN also known as SD-WAN, is at the leading edge of technology for WAN deployment on the basis of software-based networking. SD-WAN helps to improve the business value for industries and enterprises which have branched distributed over a large area. It provides various advantages to enterprises in terms of business agility and flexibility, it also provides the cost efficient deployment and helps organizations to save big in terms of the Internet bandwidth economics.

During the project we will discuss with examples to comprehend the shift from traditional networking to the advance SDN with respect to the design and network management. We will also try to understand the potential benefits these technologies will bring to an enterprise and its stakeholders. We will also put light on the Network virtualization and how and where Network Function Virtualization does intersect with SDN and further SD-WAN.

This project explains the evolution of Software Defined Networking, Network Function Virtualization and Wide Area Networks to form SD-WAN. It helps to understand the ways for migration towards a SD-WAN or a hybrid WAN which is more flexible and cost efficient to keep up the ever changing IT world. This project can help IT and business managers to understand various options for deployment of SD-WAN. This helps us to understand that SD-WAN is future of WAN technology.

ACKNOWLEDGEMENT

I would first like to thank The Almighty God without whom nothing is possible.

The amount of dedication, research and work has been huge in this project. I would like to express my deepest appreciation to all those who provided me the motivation to complete my report. It was not possible without the support and guidance of my mentor Mr. Juned Noonari, without his support, his superior knowledge and experience, the Project would have lacked in the quality, and thus his support has been essential. I would like to thank him for sharing his pearls of wisdom and for providing me motivation and great confidence throughout this project.

I would like to express my sincere thanks to Mr. Shahnawaz Mir for his guidance and helping me choose my project.

Nevertheless, I would like to express my gratitude towards my parents for their role to inspire me back in the curtains. Their kind co-operation and encouragement throughout my project motivated me and helped me complete my project.

Sarabjit Singh

Table of Contents

List of Figures

Chapter 1

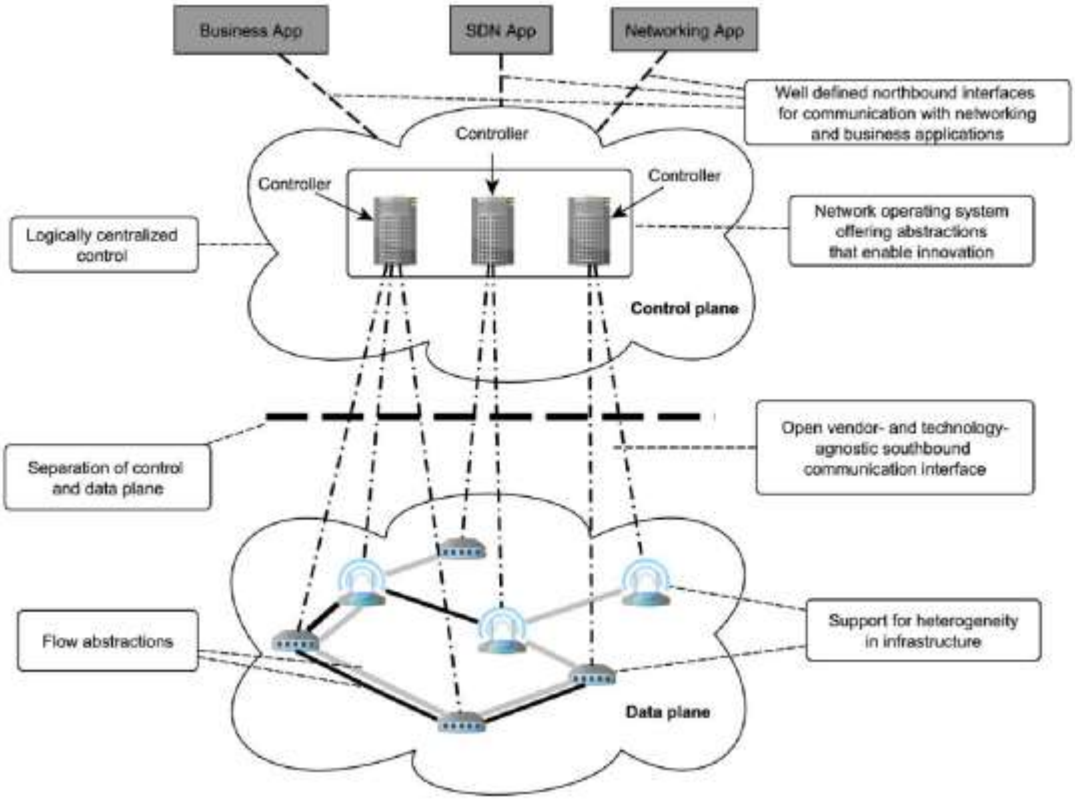
Introduction to SDN

Software-Defined Networking (SDN) is an area which challenges the traditional way of networking and it has recently reinvented the interest of network scholars and researchers towards programmable networks and brought the attention of big networking corporates towards bringing programmable networks by showing the hope to make the process of designing and managing networks more innovative and simplified and cost efficient compared to the well-established but inflexible traditional way of networking, which is to configure every networking device manually and mostly by being at the site. With increasing number of connecting devices, it has become important to bring simplification in networking as designing and managing computer networks can become a very difficult task as it is very complex and challenging and a small network problem can cause a high financial loss to any corporate. So, it was important to bring some evolution to the way networking is done and there has been a substantial improvement in this industry and the number of users are increasing exponentially. In this course of improvement the software defined networking has made its place over the years and it is getting better day by day.

There is a tight binding between a network's control plane and data plane. Control plan is where the decisions of handling traffic are made and data plane is where the actual forwarding of traffic takes place. This coupling brings lot of challenges to the management of network and its evolution. Network administrator and professionals need to manually transform high level business based network policies into low-level network configurations. This process is challenging for complex networks and can be error-prone. Thus, SDN requires separation of data plane and control plane.

In traditional approach of networking, introducing new functionality to the network, like intrusion-detection/prevention systems and load balancers usually requires tampering with the network’s infrastructure and has a wrong impact on its logic and services. Deploying new protocols can be a slow process demanding years of standardization and testing because it should be robust and to ensure its interoperability among the implementations across devices from various vendors. There is less scope of innovation and a slow process in this fast pace world won’t last long so the idea of programmable networks has been proposed over the years by including few features gradually as a means to avoid this situation by promoting innovation in network management and the deployment of network services through programmability of the network entities and this was done using an open network API.

This leads to flexible and robust network which can operate according to the way user want it to and the need for continuous modification in hardware also decreases like the way programming language are used to reprogram computer to perform different tasks and not modifying the hardware platform continuously. SDN is a relatively new standard/prototype of a programmable network which changes the traditional way of networking. Network in SDN approach can be managed by separating the control and data plane and introducing an abstraction that decouples the control plane from the data plane, as illustrated in Figure 1. In this approach a software control program, referred to as the controller constitute the control plane, has an overview of the way whole network will work and is responsible for the decision making, while the hardware like routers, switches etc. constitute the data plane is simply responsible for forwarding packets into their destination as per the controller's instructions, typically a set of packet-handling rules.


Business App
SDN App
Networking App
Well defined northbound interfaces for communication with networking and business applications
Controller
Controller
Controller
Logically centralized control
Network operating system offering abstractions that enable innovation
Control plane
Separation of control and data plane
Open vendor- and technology-agnostic southbound communication interface
Flow abstractions
Support for heterogeneity in infrastructure
Data plane

<u>Figure 1 – Concise representation of SDN- Key ideas underlying the SDN standard</u> [1]

The separation of the logically centralized control plane from the data plane greatly simplifies network management and evolution in the field of network in a number of ways and hence it has become the focus of research interest in the networking community. Over the years, new protocols and applications can be tested and deployed over the network without affecting unrelated network traffic which enhance the testing of these protocols without much effect on business. Additional infrastructure can be introduced without much problem. Networking Devices and middle boxes can be easily improved, transformed and integrated into the software control, allowing new potential solutions to be implemented for problems that are affecting the connectivity since long time, like managing the highly complex core of cellular networks.

This was a general overview of SDN. Through the discussion and the examples presented in the following topics to come. One should be able to comprehend why and how SDN shifts concepts with respect to the design and management of networks and to understand the potential benefits that it has to offer to a number of interested stakeholders like network corporates, operators and researchers. We will start by comprehensive history of programmable networks and their evolution to what we nowadays call SDN. Eventually we will know that the SDN hype is not so recent, many of its underlying ideas are not new and have simply evolved over the past decades and still evolving to solve some more major problem by introducing the concepts of **SDWAN and Hybrid networks**. Hence, Understanding the history of programmable networks will provide a better understanding of the problems and the solutions proposed over time, which helped to shape the modern SDN approach.

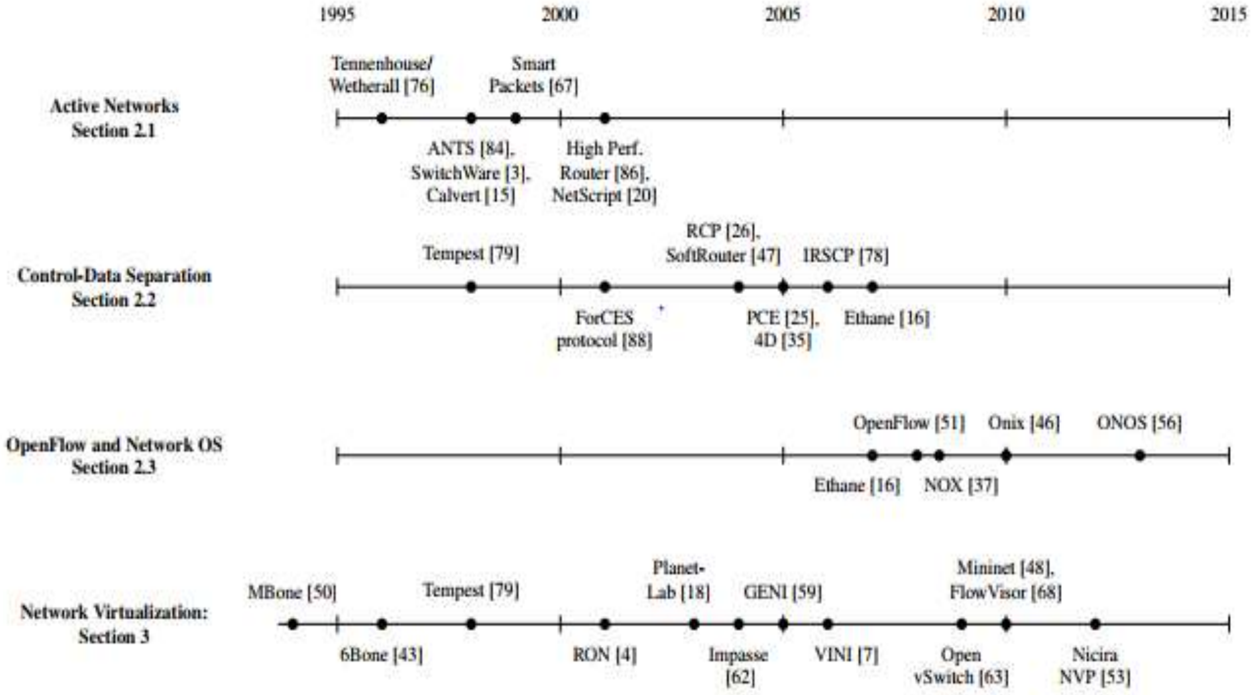
Chapter 2.

SDN History

Everything of SDN was started with programmable networks. Initially the term programmable was used to generalize the concept of the network management and reconfiguration, In fact it is important to understand that in reality it encapsulates a large number of ideas proposed over time, each having a different focus (e.g., in terms of control- or data-plane programmability) and different means of achieving their goals. This section reviews the history of programmable networks right from its early stages- when the need for network programmability first emerged, up to the present with the dominant paradigm of SDN. So, in further discussion, we will discuss the key ideas that were foundation of SDN along with other alternative approaches that were thought of and affected SDN's evolution but which were as successful to be implemented widely.

2.1 Early History of Programmable Networks

As discussed in previous sections, there is no fix time of year when the concept of programmable networks was originated, but we can say that the concept of programmable networks has its origins back in the mid-90s, that is when the Internet started to experience widespread success. The computer network has seen a minimal usage in those days limited to emails and transferring files. With the passing days, the increases usage of internet led to the increased formation of infrastructures for large networks. This further led the deployment and experimentation of new network services and ideas. However, it quickly became obvious that a major obstacle towards this direction was the high complexity of managing the network infrastructure. There was not much emphasis on vendor interoperability as network devices were used as black boxes designed to support specific protocols essential for the operation of the network. Hence, there is no such option as modifying the control plane logic of such devices. This was severely restricting network evolution. To prevent this situation, various efforts were made like focusing on finding solutions for creating more open, extensible and programmable networks. The following figure tells the appropriate timeline of how programmable network progressed to be SDN and how relevant it is today. [6,8]



1995
2000
2005
2010
2015
Active Networks Section 2.1
Tennenhouse/ Wetherall [76]
Smart Packets [67]
ANTS [84], SwitchWare [3], Calvert [15]
High Perf. Router [86], NetScript [20]
Control-Data Separation Section 2.2
Tempest [79]
RCP [26], SoftRouter [47]
IRSCP [78]
ForCES protocol [88]
PCE [25], 4D [35]
Ethane [16]
OpenFlow and Network OS Section 2.3
OpenFlow [51]
Onix [46]
ONOS [56]
Ethane [16]
NOX [37]
Network Virtualization: Section 3
MBone [50]
Tempest [79]
Planet-Lab [18]
GENI [59]
Mininet [48], FlowVisor [68]
6Bone [43]
RON [4]
Impasse [62]
VINI [7]
Open vSwitch [63]
Nicira NVP [53]

Figure 2: Selected developments and evolution in programmable networking over the past 20 years. [6]

The major idea was to find ways to separate the data plane and control plane. The two major ideas that we will discuss under this topic are the most significant early ideas leading ways of separating the control software from the underlying hardware and providing open interfaces for management and control were of the **Open Signalling (OpenSig)** working group and from the **Active Networking** initiative.

I will discuss the role of Open Signalling and Active networking briefly:

1. **Open Signalling (OpenSig) [2] -** The Open Signalling working group was introduced in 1995. At that time ATM networks were common among researchers and so ATM networks were focussed to apply network programmability. In OpenSig, the main idea was to apply signaling between the planes through the open interface and separate the control and data plane of networks. That would result in the possibility to control and program ATM switches remotely. Hence, turning the whole network into a distributed platform. This process of OpenSig eased the process of deploying new services. This idea of open signaling interfaces motivated further research. The open signaling interfaces idea allowed multiple switch controllers to manage multiple partitions of the switch simultaneously and consequently to run multiple control architectures over

the same physical ATM network. There was a sense of freedom for network operator as they don’t have to define a single unified control architecture and so they can satisfy the control requirements of all future network services.

During all the research going on, there was another project named DCAN [5] (Devolved Control of ATM networks) whose aim was to design the necessary infrastructure for the control of ATM networks. The main idea was to strip off the control and management functions of the ATM network switches from the devices and assign them to external dedicated workstations. This was result of presuming the control and management operations of multiservice networks were inherently distributed, as the need to allocate resources across a network path in order to provide QoS guarantees. There has to be a communication between the management entity and the network entity, and it was performed using a moderate protocol, like OpenFlow in modern SDN. This protocol adds up additional management functionality like the synchronization of streams in the management domain. The DCAN project had official conclusion in mid-1998.

2. **Active Networking [3]** - The Active Networking initiative was mainly supported by DARPA [7]. It had its origin in mid 90s. It was another approach towards the creation of programmable networks which would promote network innovations. Active networking basically is a network API that expose the resources of network nodes and this allows the network operator to control the nodes actively as they want and can execute relevant codes. Therefore, Active Networking differs from OpenSig as OpenSig offers a static functionality compared to Active Networking that allows the deployment of customized services and dynamic configuration of networks at run-time.

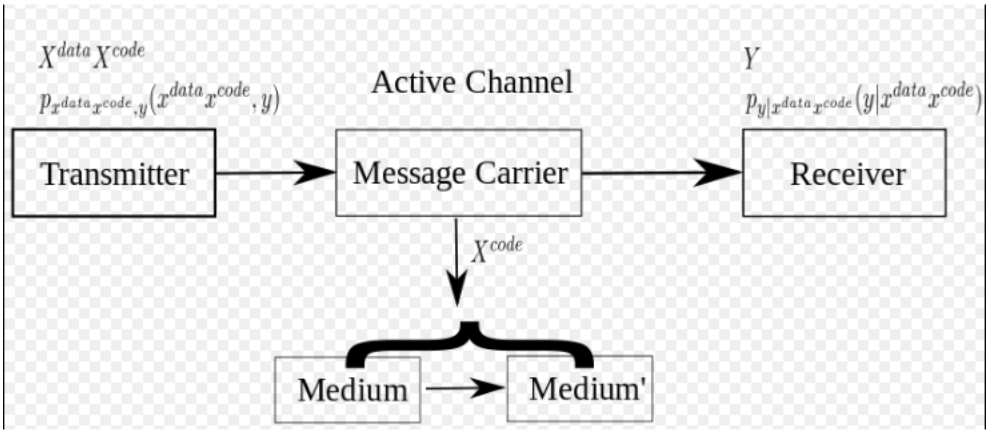


Figure 3 - Controlling the relationship between the transmitted sequence X and the received sequence Y by executable code in the packet using an active network to impact the channel. X is composed of a data portion X^data and a code portion X^code. Upon incorporation of X^code the channel medium may change its operational state and capabilities [8]

There are 2 programming models under active networking community [7]:

- The capsule model: In this model, the code that is to be executed is included in regular data packets.
- Programmable router/switch model: In this model, the code to be executed at network nodes is established through out-of-band mechanisms.

The capsule model however, seemed to be the more innovative and most closely associated with active networking because it was based on a different approach to network management that is to install a new data plane functionality across network paths. However, both models were equally important in development of SDN because many modern day concepts of SDN like separation of the control and data plane, network APIs etc. came directly from Active Networking.

2.2 Evolution of Programmable Networks to SDN

Shortcomings and contributions of previous approaches

The concepts mentioned by these early approaches of programmable networks- OpenSignalling and Active Networking had created programmable networks that would allow research and innovation. They did create an open networking environments but none of the proposed technologies of programmable networks had widespread success. Despite the fact that these were foundation to SDN, they were not as much success. One of the main reasons for this failure was that there was lack of awareness about the problem they solve and also the lack of occurrence of problems that these approaches managed to solve [6] [7]. While there were lot of performance factors that these approaches were meeting. They were beneficial for applications like content distribution and network management due to network programmability, still researchers and corporate did not find a necessity to shift to programmable networks and hence there was not much commercialization of these approaches in early days. Another reason for the failure of active networking and open signaling in becoming applicable widespread was because they focussed on wrong user group which includes only programmers working on vendors developing them.

Thus, learning from these approaches of programmable networks the new paradigm that is SDN advocated and promoted it as the flexibility it would provide to the end users to program the network the way they want to, even though in reality the use case of end user programmers was really rare. This created a negative impact on the programmable networks in the view of research community and the industry. This overshadowed the strong points of programmable networks, even for those that could really benefit from programmable networks approaches like ISPs and network operators.

Also, the main focus of these approaches went against them as many early programmable network approaches were promoting data- instead of control-plane programmability. Let us consider Active Networking which focussed on the manipulation of resources in network devices (like- packet queues, processing, and storage etc.) through an open API but did not provide any abstraction for logical control. These were the reason that obstructed any further innovation in the control plane, which covers more use cases and opportunities than the data plane.

A final reason for the failure of early programmable networks was that the lack of focus on practical issues like the security and performance. These features were clearly important factors to commercialize the idea. Therefore, despite the theoretical advantage that programmable networks had, it failed to be accepted by the industry unless pressing performance and security issues were resolved.

Despite all the shortcomings of early programmable network mentioned above. It led to the success of various other approaches that came after it. These attempts were really significant, as they were the initial key concepts to change the view of traditional networking and they marked the new research areas in this field of high potential. Even these disadvantages were of high importance, since they revealed many factors that should be addressed if the new paradigm was to be successful. In conclusion, these early attempts were of great importance and they shaped the way to the more promising and now widely accepted paradigm of SDN. They led the foundation of SDN which is being accepted and researched day by day to focus on the increasing need of networks.

2.3 Shift to the SDN paradigm

In the early 2000s, there were major changes in the field of networking. There was rise of new technologies like ADSL, providing high-speed Internet access to consumers. That was a time when an internet connection became affordable for an average user which opened up all sorts of activities, from e-mail and teleconference services to large file exchanges and multimedia via internet. The massive increase in user level of high-speed Internet and of all the new services that accompanied it had great effects on research in internet and networks. There was exponential increase in adoption of high speed internet which resulted in the increase in size and scope along with traffic volumes. With the increase in user base there was more emphasis on network reliability, performance and quality of service by Industrial stakeholders like ISPs and network operators. There was increasing need for better approaches in performing important network configuration and management functions like routing and switching.

There have been various reasons behind this network evolution. There were a number of trends that are compelling network providers and users to revaluate traditional approach to network architecture. These are majorly a result of demand, supply and traffic patterns. Let us discuss some of the changing trends that resulted in evolution in networking from time to time: [29]

1. Increasing Demands: With each day adding new users on network and new devices getting connected there has been an increase in the load on enterprise networks, the Internet, and other internets. Few of the trends that add to the pressure of better networks are the following:

- **Cloud computing:** Over the past years there has been a dramatic shift of industrial organization and enterprises to both public and private cloud services. Many firms are providing these services with Amazon leading with AWS and Microsoft with Azure and also now Google joining the game along with tie-ups with Cisco.
- **Big data:** There has been a new saying- “Data is new Gold”. This emphasis the fact that corporates are putting in so much effort and money in processing of huge data sets. It requires huge parallel processing on many servers and there is a need for a

degree of interconnection between these network devices and servers. Hence, giving rise in the demand for network infrastructure and capacity within data centre.

- **Mobile traffic:** With each year there has been a significant rise in the role of mobile devices and thus wireless network. Employees of various companies are increasingly accessing enterprise network resources via mobile personal devices, such as smartphones, tablets, and notebooks. This generate in high amount of wireless traffic, hence which adds up new burdens on the enterprise network.
- **The Internet of Things (IoT):** The latest trend of IOT will further increase the traffic. The increase in connecting more devices like cars, homes and other stuff. This generate further traffic and hence the demand of better networking approaches. There has been a significant load on the enterprise network as the number of such devices increases for some enterprises.

2. **Increasing Supply**: As there has been a remarkable increase in demand on networks, hence there is increasing capacity of network technologies to absorb rising loads. As day by days the key enterprise whether they are in wired and wireless network technologies, Ethernet and Wi-Fi respectively, all are well into high speed transmission range {gigabits per second (Gbps). Similarly, there has been a significant increase in wireless technology as 4G and 5G cellular networks provides a great facility to remote employees because of greater capacity for mobile devices who can access the enterprise network via cellular networks.

As the demand in the capacity of the network transmission technologies increases, the performance of network devices also increases. For example, the devices like LAN switches, routers, firewalls, intrusion detection system/intrusion prevention systems (IDS/IPS), and network monitoring and management systems increases their performance. With each day passing, there has been a significant improvement in buffer capacity, buffer access has become faster and the processors speeds have improve as the memories of these devices have become larger and faster.

3. **Complexity of Traffic:** If with increasing demand and supply and also development of new devices to meet these standards there is an added complexity in traffic pattern because if there was a case of only increasing supply and demand then traditional networks approach would have been able to cope with increasing data traffic. But there has been a significant changes in traffic pattern and there has been added more complexity. Thus, with the increasing demands, traditional enterprise network architectures are getting outdated.

There is a typical way of enterprise network architecture that still exist in some places minutely. It consist of a local tree structure or campus wide topology of Ethernet switches with routers connecting between LANs and these devices connecting to the Internet and WAN facilities. Similar to this, there was a client/server computing model. It was at one of the favorites and dominant at one time in the enterprise environment. It was motivated by similar architecture.

There have been a lot of development projects which have caused more complexity in traffic management. In enterprise data centre, local and regional enterprise networks and carrier networks, these projects have resulted in very dynamic and complex traffic patterns. These include the following:

- In Client/server applications, there is communication between multiple databases and servers which generates traffic in multiple directions. Logically, it generates multi-directional traffic in a system. That is horizontally between servers and also, vertical traffic between servers and clients.
- There is generation of an unpredictable traffic patterns that are often created by network convergence of voice, data, and video traffic that too of large multimedia data transfers.
- There has been rising use of applications that trigger access to multiple servers under Unified communications (UC) strategies.
- In modern day, the most increasingly significant fraction of enterprise network traffic these days is the mobile traffic. There has been a significant rise in use of mobile devices. Also the “personal bring your own device” (BYOD) policies are resulting in major complex traffic as there is an access to corporate content and applications by users from any device anywhere any time. This result in significant rise in use of public clouds which results in increased and often very unpredictable loads on enterprise routers which was a local traffic onto WANs for many enterprises in previous times.
- There has been an increasing number of hosts requiring high-volume network access because of common practice of application and database server virtualization.

4. Traditional Network Architectures are Inadequate: Traditional network devices were not worse of performers. In fact they had evolved themselves to greater capacity of transmission schemes and excellent performance of network devices. But still traditional network architectures are increasingly inadequate because of the growing complexity, variability, and high volume of the imposed load. Also, with increasing demand of quality of service (QoS) and quality of experience (QoE) requirements from the network due to the variety of applications that require these variable to be fulfilled, there should be introduction of a system that handles the traffic load in an agile fashion.

The traditional internetworking and WAN architecture approach is based on the TCP/IP protocol architecture. It refers to the protocol architecture built around two protocols- the TCP and IP protocol. TCP/IP architecture consist of five layers: physical layer, data link layer, network/Internet layer, transport layer and application layer.

Three Important characteristics of this approach are as follows:

- Two-level end system addressing
- Routing based on destination
- Distributed, autonomous control

Let us discuss about each of the characteristics briefly to understand the inadequacy of traditional approach: [29]

There has been a heavy relation between traditional architecture and the network interface identity. Hardware-based identifiers, such as Ethernet MAC addresses are used to identify the devices attached in a network and this is done at the physical layer of the TCP/IP model. The architecture is a network of networks at the internetworking level, including both the Public networks and private networks. There is a logical network identifier attached with physical layer identifier attached to each device. At this layer, its IP address provides global visibility.

The networking of autonomous networks, with distributed control is done using this addressing scheme under the design of TCP/IP. In terms of adding new networks, this architecture provides a high level of resilience and scales really good. IP and distributed routing protocols at the internet layer is used to discover routes and used throughout an internet. Distributed and decentralized algorithms can be implemented using transport-level protocols such as TCP to respond to congestion.

In Traditional approach, packet’s destination address is used for routing. In this datagram approach, there can be different routes for successive packets between a source and destination through the internet because routers constantly try to find the minimum-delay path for each individual packet. Packets are usually considered in terms of flows of packets to satisfy QoS requirements. QoS characteristics is defined by the given flow of associated packets. This affect the routing for the entire flow.

Then there is TCP and UDP. In TCP, packet switching is done and packet that is treated independently of other packets and a logical connection is made first. Whereas in UDP, there is no establishment of a logical connection between the endpoints and a datagram carries information sufficient for routing from the source to the destination without that logical connection.

A packet sends a unit of data in a network. A packet is nothing but a group of bits that includes data to be transferred over the network and control information about the protocol used for communication through that network. The term “packet” usually applies at the network layer to protocol data units. A sequence of packets that are recognized as related or uniform between a source and destination are treated in a same fashion. A method is followed for transmitting of messages through a communications network. In this method, long messages that is to be sent from source to destination are subdivided into short packets and each packet is passed through intermediate nodes between source and destination. The entire message is received at each node, stored briefly for a very short time and then forwarded to the next node. The above process is followed at each node. This distributed, autonomous approach was very much successful in previous times as it was developed during the time of static networks and even the end systems predominantly of fixed location.

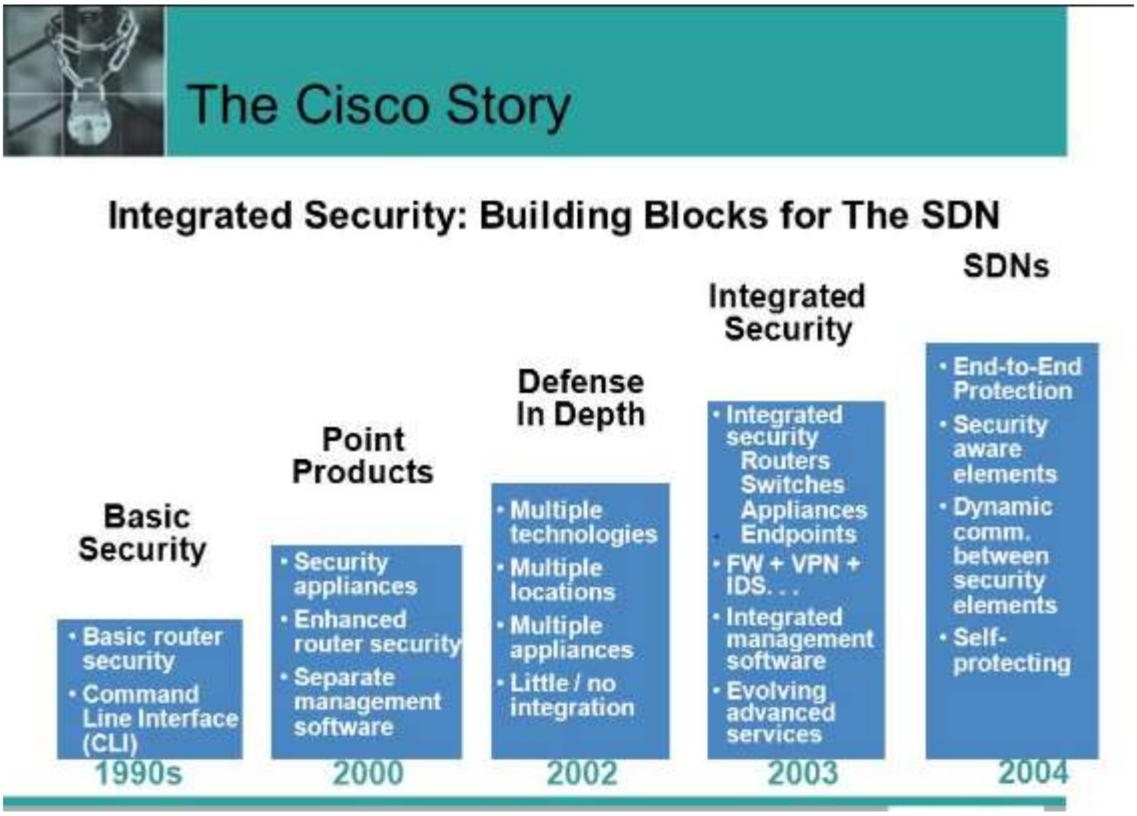
Based on all these characteristics that are discussed above, there are four general limitations of traditional network architectures [ONF12] cited by Open Networking Foundation (ONF). They are as follows:

- **Static, complex architecture**: Networking technology has grown more complex and difficult to manage in order to respond for demands such as different levels of QoS for different users, heavy traffic, fluctuating traffic volumes and network security. This has resulted in introduction of many independently defined protocols to addresses a portion of networking requirements by each protocol. There is an added difficulty when there is need to add or move devices in a network. In order to make changes to configuration parameters, there is a need for the use of device-level management tools to make any changes in network devices like multiple switches, routers, firewalls, web authentication portals, and so on. This is done by network management staff. To include the updates, there are many protocol-related adjustment done, also there are changes done to access control lists (ACLs), virtual LAN settings, QoS settings in many devices in that network. There is also a case where to meet changing user requirements and traffic patterns, an adjustment is done in QoS parameters. Everything is done manually to each device. Manual procedures are followed for all major configurations, even for configuring each vendor's equipment on a per-application and also per-session basis.

- **Inconsistent policies**: It is difficult to implement a network-wide security policy because to do so, configuration changes need to be done in thousands of devices and mechanisms manually. If a new virtual machine is activated in a large network, it may take hours or even days to reconfigure ACLs across the entire network.
- **Inability to scale**: There has been a rapid growth in the demands on networks, both in terms of volume and variety. Because for scaling we need to add more switches and transmission capacity. Also, to add multiple vendor equipment is difficult due to the complex, static nature of the network. To overcome this there is one widely used strategy among enterprises is to oversubscribe network links based on predicted traffic patterns. But the traffic patterns become unpredictable with the increased use of virtualization and the increasing variety of multimedia applications.

All of these concepts mentioned above are important, as they created the motivation for each evolution in network field and it has led to evolution of what today is called as **software-defined networking (SDN).** Early proponents of SDN saw that network device vendors were not meeting their requirements of a modern network in terms of research, innovation and development. In SDN, the infrastructure such as routing and switching equipment were considered as expensive especially their control plane components.

Additional research and innovation led to new trends in the storage and management of information like development of cloud computing and the creating large data centres resulted in apparent need for virtualized environments, accompanied by network virtualization as a means to support their automated provisioning, automation and orchestration.



The Cisco Story

Integrated Security: Building Blocks for The SDN

Basic Security
- Basic router security
- Command Line Interface (CLI)

1990s

Point Products
- Security appliances
- Enhanced router security
- Separate management software

2000

Defense In Depth
- Multiple technologies
- Multiple locations
- Multiple appliances
- Little / no integration

2002

Integrated Security
- Integrated security
  Routers
  Switches
  Appliances
  Endpoints
- FW + VPN + IDS. . .
- Integrated management software
- Evolving advanced services

2003

SDNs
- End-to-End Protection
- Security aware elements
- Dynamic comm. between security elements
- Self-protecting

2004

Figure 4 - Integration of Security in SDN overtime was one of the reason for it to be majorly famous. [12]

Now was the time when due to all these problems compelled the use cases that programmable networks promised to solve and shifted the attention of the networking community and the industry to this topic once more. The improvement of network infrastructure like servers strengthened this shift which became substantially better than the control processors in routers, this lead to movement of the control functions outside network devices. As a result, new improved network programmability attempts were made which marked the beginning of SDN (Software Defined Network). SDN is a huge success since that day because it managed to build on the strong points of early programmable network attempts, also succeeded in addressing their shortcomings. This shift from early programmable networks to SDN was not sudden but it was a continuous process of a series of intermediate steps. The major disadvantage of early programmable networking was that there was lack of a clear distinction between the control and data plane of network devices, this was later addressed by SDN. The (Internet Engineering Task Force) IETF ForCES [9, 10] (Forwarding and Control Element Separation) working group intervened to address this drawback by redefining the internal architecture of network devices through the separation of the control from the data plane. In

ForCES there are two logical entities: The Forward Element (FE) and the Control Element (CE)

- **The Forwarding Element (FE):** This was responsible for per-packet processing and handling. It operated in the data plane
- **The Control Element (CE):** This was responsible for the control logic of network devices that is for the implementation of management protocol and control protocol processing etc.

There is a standardized interconnection protocol between the FE and CE which make sure the forwarding behavior to the FE as directed by the CE. The idea behind ForCES was to separate the data plane and control plane by allowing the forwarding and control planes to evolve separately and by providing a standard means of interconnection between them.

There was one more approach that targeted the clean separation of the control and forwarding elements of network devices, it was called the **4D project** [14]. Similar to ForCES, 4D also concentrated on separation of the decision logic from the low-level network elements. However, in contrary to previous approaches, the 4D project has architecture based on four planes which are explained as follows:

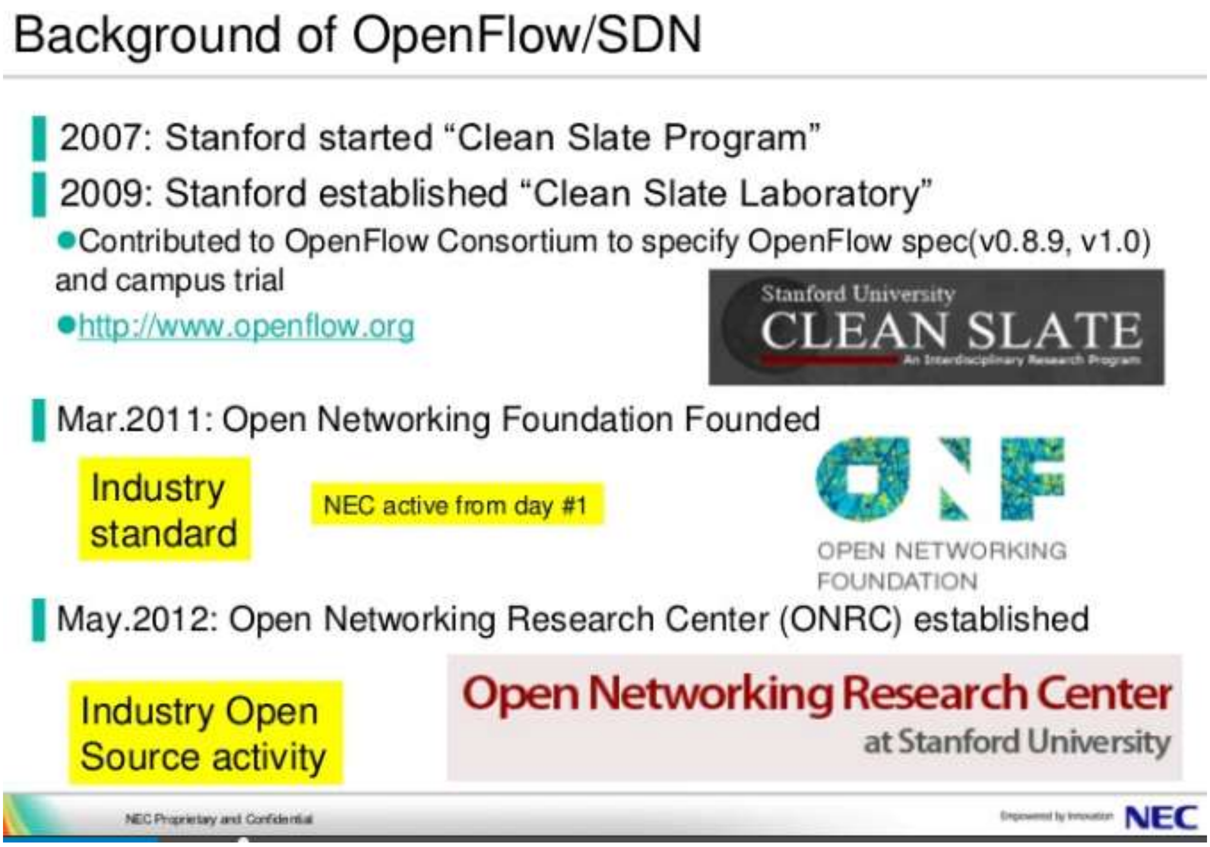
- **Decision plane:** This plane is responsible for creating a network configuration
- **Dissemination plane:** This plane is responsible for delivering information related to the view of the network to the decision plane
- **Discovery plane:** This plane allows network devices to discover their immediate neighbors
- **Data plane:** This plane is responsible for forwarding traffic.

4D Architecture continued with many experiments and systems, one such project was Tesseract. Tesseract was based on concept of the use of a single administrative domain to take the direct control of a network. The idea of logically centralized control of the network led to innovation of many projects related to the controller component of SDNs.

A final project which played an important role in pre-SDN era is SANE/Ethane. Ethane was a joint attempt by researchers in the universities of Stanford and Berkeley to create a new network architecture for the enterprise. Ethane was based on the main ideas expressed in 4D for a centralized control architecture. This was the project where security was consider to be incorporated, the argument was done to integrate security to network management as both require some sort of policy, the ability to observe network traffic and a means to control connectivity. Ethane incorporated this achievement by simply by including a flow-based Ethernet switches with a centralized controller which was responsible for managing the admittance and routing of flows by communicating with the conventional switches through a secure channel. The Ethane project was very important for further development of SDN, as the experiences gained by its design, implementation and deployment led the foundation of SDN. In particular, Ethane is also called the immediate predecessor of OpenFlow because of origination of simple flow-based switches which further formed the basis of the original OpenFlow API. Open Flow is discussed later but first let us see the emergence of SDN.

2.4 The emergence of Software Defined Networking [7, 9]

Early 2000's was the time when the real interest in network experimentation was done by funding agencies and researchers. The reason was clear, due to increasing user base of internet there was requirement of scalability and the need of automation was thought. That is when the real shift of interest towards SDN started. There was the need to deploy new protocols and services, targeting better performance and QoS in large enterprise networks and the Internet and this all motivated the research in SDN, and this motivation was further strengthened by back to back successful ventures innovating in SDN. There was huge success of experimental infrastructures like PlanetLab and by the emergence of various initiatives like the US National Science Foundation's GENI (Global Environment for Networking Innovations) that strengthened the interest in network innovation. Before this, it was really difficult to perform large scale experimentation as researchers were mostly limited in using simulation environments for evaluation. Experimentation on simulation environment could not be as effective and cannot capture all the important network-related parameters that a realistic test bed would do.



Background of OpenFlow/SDN

- 2007: Stanford started “Clean Slate Program”
- 2009: Stanford established “Clean Slate Laboratory”
  - Contributed to OpenFlow Consortium to specify OpenFlow spec(v0.8.9, v1.0) and campus trial
  - http://www.openflow.org
- Mar.2011: Open Networking Foundation Founded
- May.2012: Open Networking Research Center (ONRC) established

Stanford University
CLEAN SLATE
An Interdisciplinary Research Program
Industry standard
NEC active from day #1
ONF
OPEN NETWORKING FOUNDATION
Industry Open Source activity
Open Networking Research Center
at Stanford University
NEC Proprietary and Confidential
Empowered by Innovation
NEC

Figure 5- Some Important projects and their timeline for development of SDN and OpenFlow. [13]

During all the research and innovation effort in order to achieve simplified network management and network services deployment and to allow multiple experiments to run simultaneously at the same infrastructure using different set of forwarding rule for different cases there was one important requirement that needed to be addressed and that was the need for network programmability. Motivated by this idea, “Clean Slate Program” was created by a group of researchers at Stanford. This project, which was introduced as a mission to “reinvent the internet” lead to the proposal of the OpenFlow protocol as a means for researchers to run experimental protocols in everyday networking environments. OpenFlow was introduced and architected for a number of devices that contain only data planes. These data planes in OpenFlow are used to respond to commands sent to them from a centralized controller present in a single control plane for that network. This controller has responsibility for maintaining all of the network paths in a network. It also performs programming for each of the network devices it controlled. OpenFlow protocol describes the commands for these commands and responses. It is also important to notice that the Open Networking Foundation (ONF) supported the SDN effort commercially. Today, it is its central standardization authority and marketing organization. We just now described the basic architecture, On the basis of it, one can now infer how easy and fast it enabled to devise a new networking protocol by simply implementing it within a data centre. And it is cost efficient as well. Even better, one could also implement it in a virtual computing environment that is, virtual machine. Considering the concepts of ForCES, OpenFlow also included the principle of decoupling the control and forwarding plane, and standardized the information exchanges between the two using a simple communication protocol. OpenFlow provided the solution that created an architectural support for programming the network. This led to the creation of the term SDN to encapsulate all the networks following similar architectural principles. The Basic idea behind SDN is the creation of horizontally integrated systems through the separation of the control and the data plane while providing an increasingly sophisticated set of abstractions. If we have a look at the history of programmable network and projects presented in previous sections, we can conclude that the road to SDN was not a short one. It constitute of various ideas being proposed, tested and evaluated, driving research in this field over time and even further. SDN was never a new idea, it grew with many experiences and knowledge gained from projects over time.

With increasing user base and the services covered by network, there was an increasing list of requirement that need to be met and so there was a lot of research following up. That is when the Open Data Centre Alliance (ODCA) came up with a useful, concise list of requirements which should be met in updated network paradigm, which include the following [30]:

Adaptability: As there were all types of applications that were getting build with time and generated new set of requirement from network end considering the security perspective of the enterprise also. On the basis of application needs, business policy, and network conditions, it is important for Networks to adjust and respond dynamically.

Automation: As the number of users are increased, the infrastructure requirement increased rapidly and so, to configure them manually for each updating Policy, change was complex, costly, time consuming and it also caused errors. Each policy update must be automatically propagated so that manual work and errors can be reduced. Also SDN leads to automation in configuration and other perspectives of modern day networks.

Maintainability: This is a major requirement as introduction of new features and capabilities (software upgrades, patches) should not disrupt the operations and it must be seamless as maintenance parts a very high cost in traditional networking as the enterprise grows.

Model management: There must be management of the network at a model level by network management software. There should not be reconfiguration of individual network elements to implement conceptual changes.

Mobility: Considering the fact that there is an exponential increase in the number of mobile devices and virtual servers over the past few years. So the latest network technology should have included control functionality for mobility, including mobile user devices and virtual servers.

Security: Security is a major issue over past many years in networking and there has been many cases of losses due to network security in many corporates. There must be integration of seamless security as a core service instead of as an add-on solution in network applications.

Scaling: As enterprise grows there should be a possibility for the same network to grow rather than total change in infrastructure. Network implementations must be able to scale up or scale down. Also, its services should support the same feature to support on-demand requests. [30]

As compared to software-defined networks, software-driven networks is just a slightly different view of SDN. These are considerably different approaches. Software driven approach is important to implement in transition towards SDN approach. In the software-driven approach, one views OpenFlow and that architecture as a distinct subset of functionality that is possible. One views the world as more of a hybrid of the old and the new technology where it contains logically centralized control planes but the network devices are not brainless either. Moreover, the truth is that it is unrealistic to think that existing networks are going to be destroyed in abundance to make way for a new world of networking, as explained by the ONF and software-defined networks. It is also practically impossible to remove all of the network infrastructure working behind internet and the advances in network technology that exist today. Instead, it is practically suitable to opt for a hybrid approach which is combination of both approaches. Here, some portion of networks can be automated and operated by a logically centralized controller, whereas other parts of network that are costly or not appropriate to replace, would be run by the more traditional distributed control plane. This would also mean that both these approaches need to interwork with each other for a successful transition.

However it is also a fact that SDN and OpenFlow proponents are achieving more flexible network device programmability. However, this is more concerned about how they are programmed and not the location of the network control. It should not be forgotten that the major motivations for creating SDN and OpenFlow was the flexibility. Flexibility of how to program a network device, and not just where it is programmed. Both of these questions are solved, if one concentrate on the SDN architecture described above. The question is, what is the most optimal choice? Is it the programmability aspect or not.

SDN is an idea coming from the ideas discussed in previous sections. If we question what SDN managed to do differently compared to these ideas? The answer is that SDN is not much different than these ideas. SDN has an integration of its architecture into the programmability concepts that are discussed by these ideas, this leads to many interesting use cases that see its practicality by many parties interested in it. SDN is still improving and making space for itself to be the next major innovation in the world of networking.

To perform this programmability constraints and other features of SDN, many vendors and service providers such as Juniper, Cisco, and many more introduced I2Rs that is, Interface to the Routing System (I2RS). Which is considered to be a prominent step towards network programmability. Many representative from the above sources have contributed to several IETF drafts towards this concepts for example frame-work drafts contributed by Alia Atlas, David Ward, and Tom. There should be more upcoming drafts around this topic in the near future as there has been a great interest in this effort. The basic idea around I2RS is to create a protocol and components to act as a means of programming a network device’s routing information base (RIB) that uses a fast path protocol to allow a quick solution of provisioning operations instead of allowing for real-time interaction with the RIB and the RIB manager that controls it. Previously, the only access one had to the RIB was via the device’s configuration system (in case of Juniper it was Netconf or SNMP).

To better understand I2RS, the key is that it is definitely more than just another provisioning protocol and that is because other problems are taken care by many solutions. Examples of other problems are slow feedback loop between network devices, their programming state and the analytics processes after there processing is done. Those corporates involved in I2RS believe that optimizing this loop of l2RS is the key to the future of programmable networks. I2RS provides various levels of abstraction in terms of programmability of network paths, policies, and port configuration. In all cases, it has the advantage that it allows adult supervision of the programming done, in order to check the commands before committing them. For example, some protocols are used for programming at the hardware abstraction layer (HAL), which places an undue burden on its operational systems as it is far too granular or detailed for the network's efficiency. There is another example of it is that it optimize the network by fast reprogramming whenever it witness any changes in program before it check the results and that is because of the fast and optimal access by operational support systems (OSS) applications of the RIB. One key aspect around all of these examples is that the RIB manager helps in occurrence of discourse between the applications and the RIB. This is important, as many operators would like to leverage this new and useful programmability paradigm to allow additional levels of optimization in their networks and also preserve their operational investments in routing protocol intelligence that usually exists in device operating systems such as Juniper's Junos or Cisco's IOS-XR.

I2RS also lends itself in order to perform logical centralization of routing, making decisions for paths and network programmability. This helps to perform distributed controller functionality by allowing cases such as allowing devices to perform network functions by being in the network or by physically being away from network areas.

where it is desired. However, we are also able to support in case more classic distributed control is desired.

Finally, another key subcomponent of I2RS is normalized and abstracted topology.

To represent this topology, one need to define a common and extensible object model. This service also exposes multiple abstractions of topological representation.

A key aspect of this model is that non-routers (or routing protocol speakers) can more easily manipulate and change the RIB state going forward. Today, non-routers have a major difficulty getting at this information at best. Going forward, components of a network management/OSS, analytics, or other applications that we cannot yet envision will be able to interact quickly and efficiently with routing state and network topology.

So, it is important that we define SDN for what we think it is and its future prospects. [27]

Chapter 3

SDN- Introduction and Concepts

As we have already discussed the history of SDN and programmable network. We infer from our previous sections that SDN is a part of evolution of network and programmable networks and this was a continuous process. Now let us consider the basic idea of SDN and in upcoming sections we will focus on the key ideas underlying the SDN paradigm, the most recent instance in the evolution of programmable networks. We need to examine SDN both macro- and microscopically to understand the SDN concepts and to comprehend the benefits that this paradigm promises to deliver. There can be a proper study on the way in which computer systems and computing applications evolved into an open approach to computing from a closed, vertically integrated, proprietary systems. This evolution was similar to the evolution coming with SDN. Let us look at the figure shown below:

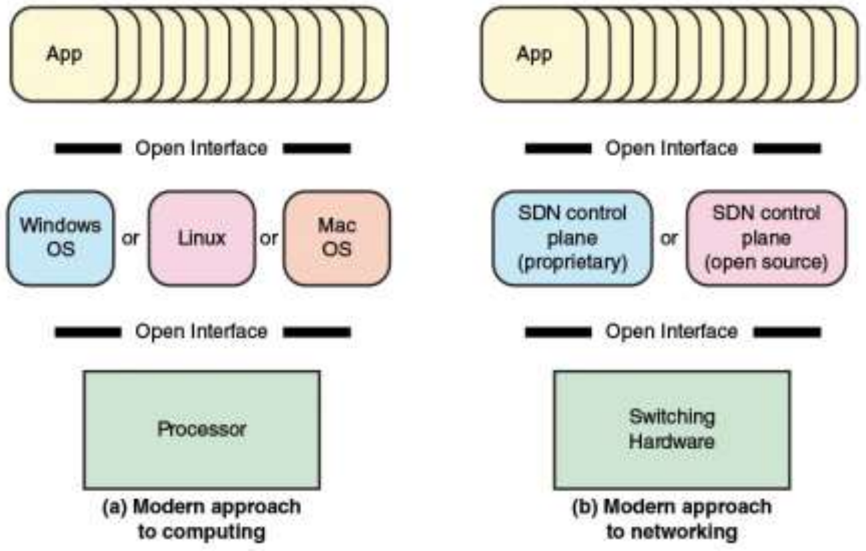

App
Open Interface
Windows OS
or
Linux
or
Mac OS
Open Interface
Processor
(a) Modern approach to computing
App
Open Interface
SDN control plane (proprietary)
or
SDN control plane (open source)
Open Interface
Switching Hardware
(b) Modern approach to networking

Figure 6: Evolved Approach to computing and networking [14]

In the early days of computing, there was no such thing as standardization or compatibility in different vendors. There were vendors such as IBM and DEC which used to provide a fully integrated product. If not all of the application software, it used to provide at least a proprietary processor hardware, unique assembly language, unique operating system (OS), and the bulk. In this environment, there was restriction to customers, especially large customers. This system tends to restrict customers to one vendor depending on the applications offered by that vendor. It was difficult to migrate to another vendor's hardware platform as it would result in major changes in support infrastructure and at the application level. In modern days, the computing environment has changed a lot with high openness and great customer flexibility. These days, the computing hardware are advanced and it consists of heavy processor like x86 and x86-compatible processors for standalone systems and ARM processors for embedded systems to port operating systems implemented in C, C++, Java, and the like easily. Even exclusive hardware architectures, such as IBM's Enterprise line were more of moving towards this open source operating system environment. This enterprise line provides standardized compilers and programming environments and it can easily run Linux based operating system and other open sources operating systems. Hence, applications written for Linux or other open operating systems are more flexible in terms of vendor platform as it can easily be moved from one vendor platform to another. It also helped in development of virtual machines that enables the movement from one server to another, across different hardware platforms and operating systems. Similar to the evolution of computing technology, the networking environment also faced the same limitations to the pre-open era of computing. The major issue is the lack of integration between applications and network infrastructure, it is not developing applications that can run on multiple platforms.

The central concept behind SDN is same as discussed in above paragraph. It helps to enable developers and network managers to have the similar control over network equipment that they have had over x86 servers. This will enhance the features of SDN and it would be beneficial for Business. The SDN approach splits the switching functions between a data plane and a control plane. In this approach, the control and data plane are on separate devices. The responsibility of the control plane is to provide the intelligence in designing routes by making route decisions at real time, setting priority and routing policy parameters to meet QoS and QoE requirements and to cope with the shifting traffic patterns whereas, the role of data plane is to forward data packets. The switching hardware should present a uniform interface independent of the details of internal implementation to define open interfaces. Similar to that, SDN has open interfaces that are defined to help networking applications to communicate with the SDN controllers.

We will discuss a general overview of its architecture before going into an in-depth analysis of its building blocks.

3.1 Overview of SDN Building Blocks

As already mentioned, throughout the history the management of network has been done by manually logging into network devices like routers and switches and then issuing configurations and getting the desired output. But in long run researchers realized this is not the most appropriate way to do because when large organization scale, there are tens of thousands of network devices to be configured. SDN (Software Defined Networking) helps to address this by abstracting the control of networking (control plane) from the execution of networking (data plane). The SDN approach helps in the management of network services through the abstraction of lower level functionality. Network administrators now only need to use the abstractions available in the SDN architecture instead of dealing with low level details of network devices regarding the way that packets and flows are managed. This approach has brought many other technological advancement like Data Centers, SDWAN and many more that we will discuss in further sections.

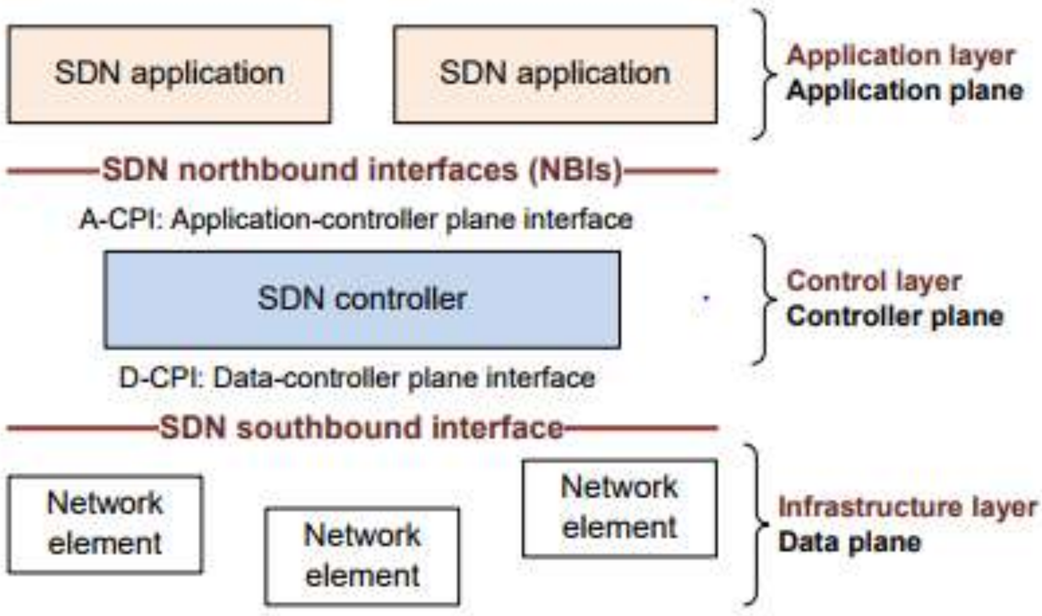


Figure 7: Basic Building Blocks of SDN [14]

As shown in the above image, at the bottom layer we can observe the Infrastructure layer which is the data plane, where the network infrastructure (switches, routers, wireless access points etc.) lies. According to the concept of SDN, all the control logic (e.g., routing algorithms like BGP) has been removed of these devices. They simply implement a set of forwarding operations for manipulating network data packets and flows, providing an abstract open interface for the communication with the upper layers. According to the SDN terminology the devices on data plane are commonly known as **network switches**.

Considering the next layer in the figure which is the control plane, where an entity called as the **controller** lies. Controller is the most important entity in SDN as it encapsulates the networking logic and is responsible for providing a programmatic interface to the network. This interface is used to implement new functionality and perform various management tasks. Considering the disadvantages of previous approaches like ForCES, the control plane of SDN is considered to be centralized logically and is separated entirely from the network device that is data plane, whereas physically it can be either centralized or decentralized residing in one or more servers or devices, which control the network infrastructure as a whole. The most important difference between SDN and previous programmable network attempts is that SDN has introduced the notion of the network operating system abstraction that provides an application programming interface (API) to an abstract operating system. Hence, developing code for multiple software or hardware platforms becomes easier and faster. If we revisit previous sections we infer that efforts like active networking proposed some sort of node operating system (e.g., NodeOS) for controlling the underlying hardware.

The Network Operating System can be called as controller as well play an important role as an intermediate layer between Data layer and Control layer. It is responsible for maintaining a consistent view of network state, which is then used by control logic to provide various networking services for topological discovery, routing, management of mobility and statistics etc. A network operating system reveals a simple interface for controlling the network and thus it offers a more general abstraction of network state in switches. This abstraction is like a logically centralized control model, in which the application's API view the network as a single system. In simple words, the services provided by the controller to perform network-related tasks, like load balancing, network virtualization etc. is performed by the applications that are part of application layer which lies on the top of the SDN Stack. One of the most important features of SDN is the ability of it to let the third party developer define the development and deployment of new applications through the abstraction. It allows the developer develop and deploy new applications in various networked environments from data centres and WANs to wireless and cellular networks. Also, the SDN architecture eliminates the need for dedicated middle boxes like firewalls, Intrusion prevention systems (IPS) and Intrusion Detection Systems (IDS) in the network topology, as SDN made it possible to implement their functionality in the form of software applications that monitor and modify the network state through the network operating system services. The abstraction layer has great importance to SDN as it motivates the innovation, making SDN an important solution both for researchers and the industry. Therefore, the communication of the controller to the data plane and the application layer can be achieved through well-defined interfaces (APIs).

There are two main APIs in the SDN architecture:

i) **A southbound API** : This API is important for the communication between the controller and the network infrastructure

ii) **A northbound API**: This API is used to define an interface between the network applications and the controller.

This is similar to the way communication is achieved among the hardware, the operating system and the user space in most computer systems. We have discussed the SDN Architecture Briefly as shown in fig below. In further discussion we will consider some building blocks in detail. Some of the examples of SDN applications will also be discussed in the next section.

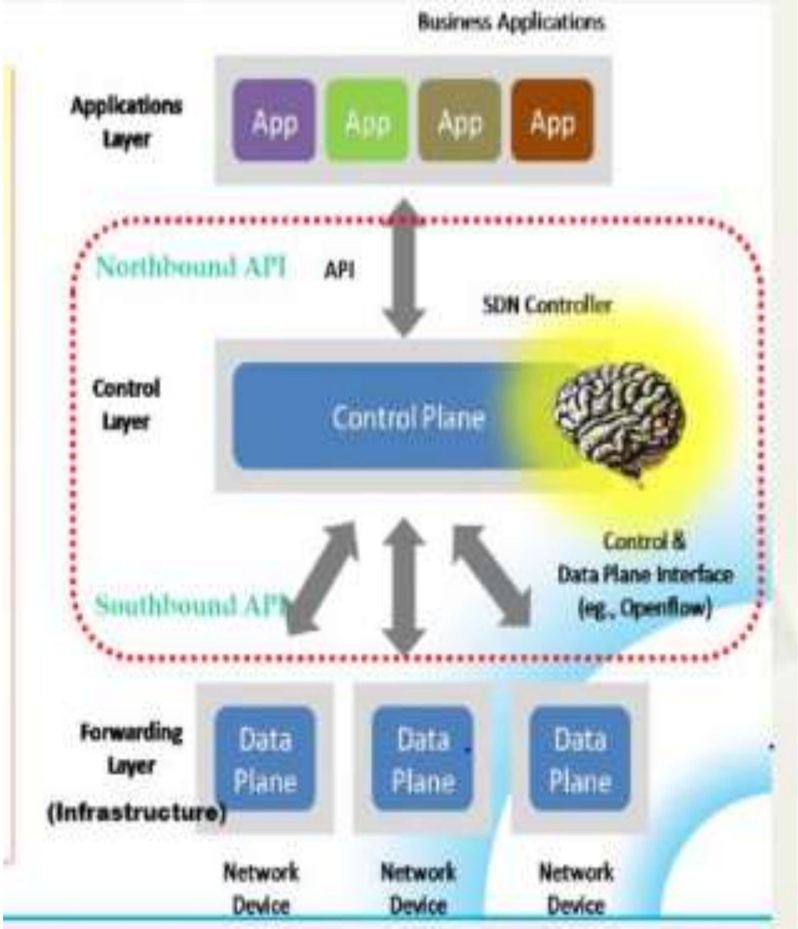


Figure 8- SDN Architecture and basic building blocks [15]

3.1.1 SDN Switches

In the traditional networking paradigm the network infrastructure is considered to be the most important part of the network. Each network device has a role in performing the functionality that would be required for the operation of the network. For instance, a router is used to forward packets between networks considering different routing protocol configured on it. It needs to provide the proper hardware like a Ternary Content Addressable Memory (TCAM) for quickly forwarding packets, as well as a software to execute distributed routing protocols like BGP. Similarly, other devices like a switch also need to have both hardware and software and also wireless access point needs to have the proper hardware for wireless connectivity as well as software for forwarding packets, enforcing access control etc. Although, changing the behavior of a network device dynamically is not an easy task because of their closed nature.

As shown in the figure above the three-layered SDN architecture makes it possible to change the behavior of networking devices by decoupling the control from the forwarding operations, simplifying the management of network devices. As already discussed, all forwarding devices keep possession of the hardware that is responsible for storing the forwarding tables (e.g., Application-specific integrated circuits - ASICs - with a TCAM), but remove their logic. The controller commands the switches how packets should be forwarded by installing new forwarding rules through an abstract interface. Each time a packet arrives to a switch its forwarding table is checked and according to the forwarding information the packet is forwarded.

Even after giving an overview of SDN clearly defining its three-layered architecture, it's still not clear about the clear boundary between the control and the data plane should be because there are still some functions performed by the layers which should be performed by the other one for example, Active Queue Management (AQM) and scheduling configuration are operations that are supposed to be a part of control plane but are considered to be part of the data plane even in the case of SDN switches. Although, there is not much problem preventing these functions from becoming part of the control plane by introducing some sort of abstraction allowing the control of low level behavior in switching devices. This type of approach could turn out to be valuable because it would result in simplification of the deployment of new more efficient schemes and more innovation for low level switch operations [16].

Considering other aspects of moving all control operations to a logically centralized controller. While it has the advantage of easier network management, it can also raise scalability issues if physical implementation of the controller is also centralized. Therefore, it might be important to keep some of the logic in the switches. This is taken care in some of the models like in the case of DevoFlow [17], which is a modification of the OpenFlow model. In DevFlow, the packet flows are distinguished into two categories: small ("mice") flows handled directly by the switches and large ("elephant") flows requiring the intervention of the controller. Similar model that consider that concept was the DIFANE [18] in which, the controller intermediate switches are used for storing the necessary rules and the controller is reassigned to the simple task of partitioning the rules over the switches.

Another issue of SDN switches that need to be considered is that the forwarding rules used in the case of SDN are more complex than those of traditional networks. There are lot of rules in SDN that makes it complex like using wildcards for forwarding packets, considering multiple fields of the packet like source and destination addresses, ports, application etc. As a result of these complex rules, the switching hardware cannot easily cope with the management of packets and flows. For the following complex forwarding operation to be fast there is requirement of ASICs using TCAM but such specialized hardware is expensive and power-consuming. As a result only a limited number of forwarding entries for flow-based forwarding schemes can be supported in each switch, which hinders network scalability. A way to subsist with this situation would be to introduce the switch with an assisting CPU to perform not only control plane but also data plane functionalities, e.g., let the CPU forward the “mice” flows or to introduce new architectures which would be more expressive and would allow more actions related to packet processing to be performed.

If we look at the issues associated with hardware limitations, it is not only restricted to fixed networks but is extended to the wireless and mobile domains as well. Similar to the data plane of fixed networks, the data plane of wireless network also needs to be redesigned in order to offer more useful abstractions. Despite the fact that the data plane abstractions offered by protocols like OpenFlow support the idea of decoupling the control from the data plane, the same concept cannot be extended to the wireless and mobile field unless the hardware (e.g., switches in backhaul cellular networks and wireless access points) starts providing equally sophisticated and useful abstractions [6].

In order for the new paradigm of SDN to gain popularity, backward compatibility is a very important factor Regardless of the way that SDN switches are implemented. Despite the fact that there exist pure SDN switches that completely lack integrated control but the most successful approach at the early step of SDN is the hybrid approach which support of SDN along with traditional operation and protocols [9]. The reason for this is that the infrastructure in most enterprise networks still follows the conventional approach. Knowing that the features of SDN present a compelling solution for many realistic scenarios but still switching to SDN switches as infrastructure completely is costly and inappropriate from industrial point of view. Therefore, there is a requirement of an intermediate hybrid network form which can ease the transition to SDN.

3.1.2 SDN Controllers

As discussed in previous topics, the most important and basic component in SDN is the existence of network operating system placed between the network infrastructure and the application layer. The major achievement of SDN is because of the idea of coordinating and managing the resources of the whole network and for revealing an abstract unified view of all components to the applications executed on top of the network operating system. This idea is similar to the one followed in a typical computer system, in which the operating system lies between the hardware and the application and is responsible for managing the hardware resources and providing common services for user programs. Similarly, these days there is a homogeneous environment present for network administrators and developers which is easier to program and configure much like a typical computer program developer would. This functionality of logically centralized control and the generalized network abstraction is offered by controller in SDN. This makes the SDN model applicable to a wider range of applications and heterogeneous network technologies compared to the conventional networking paradigm.

We know from the SDN architecture that the controller is the key entity. Logically, controller is the top of the architecture hierarchy, which serves as the brain of the system. However, Still we need to discuss is how that controller functions or even fits into our network topology. This section concentrate on the functionality of controller and how they play an integral part in regulating the data flow between the application layer and the physical layer, where application layer comprise of the applications we use and the physical layer represent the routers and switches our data has to cross. You must have another means of centralized control, if you are going to replace the control layer of every device (router or switch). That is where the central controller comes into play.

To understand the importance of SDN Controller let us consider a heterogeneous environment composed of a fixed and a wireless network comprised by a large number of related network devices (routers, switches, wireless access points, middle boxes etc.). If the case is traditional networking paradigm then each network device would need an individual low level configuration by the network administrator in order to operate effectively. Also, if each device targets a different networking technology then it would have its own specific management and configuration requirements, meaning that would require an extra effort by the administrator to make the whole network operate as intended. On the other hand, if we consider the same case in terms of the logically centralized control of SDN, the administrator does not have to think much about low level details but, there is a need for the administrator to define high level polices as they would perform the network management. So, in SDN, the major responsibility of configuring the operation of network devices is performed by the network operating system.

As we have discussed the basic idea behind the SDN controller. In the following subsections, let us take a closer look at specific design decisions and implementation choices made at this core component that can prove to be critical for the overall performance and scalability of the network.

3.1.2.1 Centralized control in SDN

The controller is a centralized unit that work as a ‘brain’ in the network. It will have a global view of all the network devices, their interconnections, and the best paths between hosts. Having this single global map of the network enables the controller to make swift, intelligent, and agile decisions with regard to flow direction, control, and speedy network reconciliation when a link fails. As shown in following figure, the network no longer has to converge, through multiple devices in the network swapping routing tables, running an algorithm, before they update their routing tables and then recalculate the preferred routes.

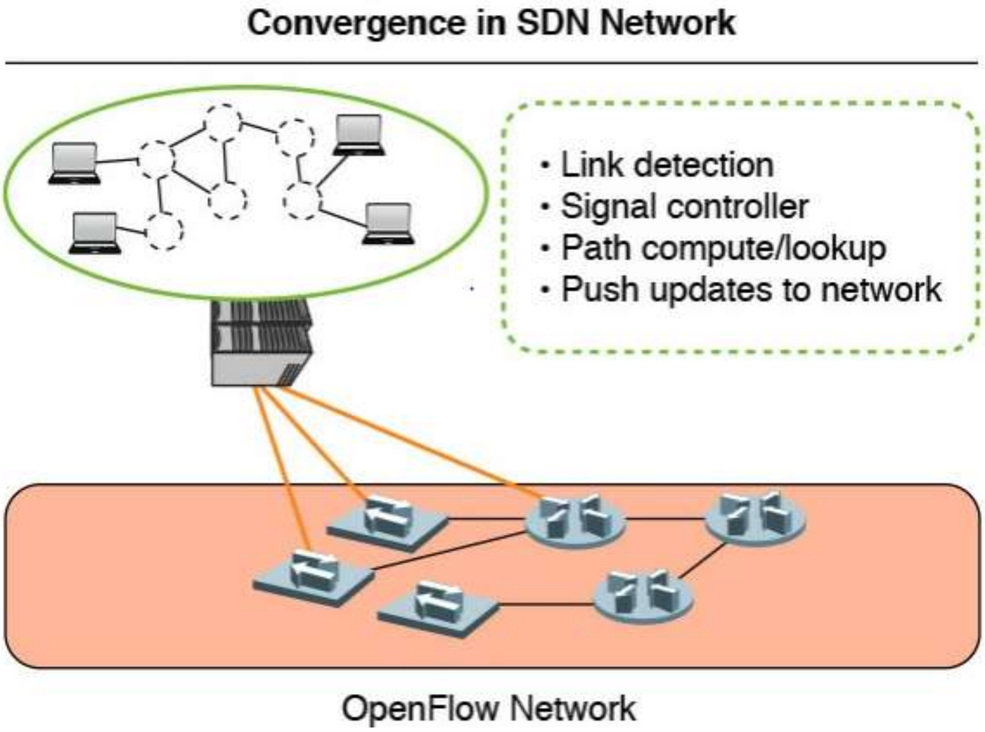


Figure 9- Network convergence is much more efficient in SDN. [9]

After all, the time to converge in a network is the time to detect a failure + the time to announce a failure to all parties + the time to run the algorithm + the time to update the databases in each device. However, in a software-defined network (SDN), the controller already has a global view of the entire network and a selection of predefined alternative routes (flows) for every link so that it can fail over to an alternative route quicker and more gracefully than a traditional routing protocol such as Open Shortest Path First (OSPF) or Enhanced Interior Gateway Routing Protocol (EIGRP). After all, it doesn't have to re-compute the shortest path for every link. It already knows shortest paths and convenient path, and therefore there is no time required to run a routing protocol algorithm or update the routing tables—which is not an inconsiderable time during network convergence.

As discussed in previous sections, SDN architecture explains that there is a central entity that controls the network infrastructure logically. This central entity is responsible for management and policy enforcement and that central entity is controller. Also the point to understand is that logically centralized control does not necessarily also imply physical centralization. Although, there have also been various proposals for physically centralized controllers, like for instance NOX and Maestro [9]. A physically centralized control design simplifies the controller implementation. In SDN, the network is not subject to consistency related issues, with all the applications seeing the same network state (which comes from the same controller) because all switches are controlled by the same physical entity. This Approach of centralized control of network has one disadvantage that there can be a single point of failure for the whole network and that is, Controller. In order to mitigate this issue, one can include backup controllers or more than one controller to work in case of the failure of major controller. There can be other drawbacks of the centralized approach, one of them can be that it can raise scalability concerns, since all network devices need to be managed by the same entity. One approach to tackle these drawbacks is to use multiple controller over a network by maintaining a logically centralized but physically decentralized control plane. Then in that case, many controller together manage a large network as they all communicate with each other efficiently to perform their functions by having a common network view. This becomes difficult for one controller as it can perform its function for only one part of the network. This enable the control operations to be performed by a distributed system but, applications view this distributed control system as a single entity. There are many advantages of this approach.

1. This approach does not have a single point of failure anymore.
2. This approach increases performance and scalability as each individual controller need to manage only a part of the network.

Some well-known examples of the controllers that belong to this category are Onix [21] and HyperFlow [22]. One potential downside of decentralized control is once more related to the consistency of the network state among controller components. There can also be a case to address where applications served by different controller could have different view of network thereby giving different performance and also which might make them operate improperly because the state of the network is distributed.

A hybrid solution that tries to encompass both scalability and consistency is to use two layers of controllers like the Kandoo [23] controller does. The controllers present at the bottom layer do not have knowledge of the whole network state. These controllers only run control operations for local network of that switch and so they require knowing the state of a single switch (local information only). Whereas, the controllers which operate at the top layer are logically centralized controller and thus responsible for performing network-wide operations that require knowledge of the whole network state. The whole idea of this approach is that if there is requirement of local operations to be performed, they can be done faster this way and do not incur any additional load to the high-level central controller, effectively increasing the performance and tackling the scalability issue of the network.

There have also been other approaches proposed where the idea of logical decentralization have been proposed apart from the ideas related to the level of physical centralization of controllers. This idea of logical decentralization comes directly from the early era of programmable networks and from the Tempest project. In that project, the Tempest architecture allowed multiple virtual ATM networks to operate on top of the same set of physical switches.

Despite all the discussion about the SDN and the level of centralization with SDN controller. There is a need of discussion regarding the quality of service and their effect on performance and applicability over large networking environments. One of the most commonly raised concerns by SDN doubters is the ability of SDN networks to scale and be responsive in cases of high network load. This concern is backed by the fact in the new paradigm i.e. SDN is that the control moves out of network devices and goes in a single entity which we call controller is responsible for managing the whole network traffic.

Motivated by this concern, performance studies of SDN controller implementations [23] were done. These studies revealed that even physically centralized controllers can perform really well, having very low response times. This argument was made with proofing done from different approaches. For instance, during these studies it has been proved that even primitive single-threaded controllers like NOX can handle an average workload of up to 200 thousand new flows per second with a maximum latency of 600ms for networks composed of up to 256 switches. Then the New multi-threaded controller implementations were checked and they showed even better results. For example, NOXMT can handle 1.6 million new flows per second in a 256-switch network with an average response time of 2ms in a commodity eight-core machine of 2GHz CPUs. Newer controller designs promised to improve the performance even further because they were targeting large industrial servers. One of them is the example of the “McNettle” controller. It claims to be able to serve networks of up to 5000 switches using a single controller of 46 cores with a throughput of over 14 million flows per second and latency under 10ms.

One more important concern in this paradigm to be discussed is the approach of physically decentralized control plane [24]. This depends on the physical position of the controllers within the network. There can be many factors that can affect network performance, some of them are the number of controller and the physical location of controllers, also by the algorithms used for their coordination. So, seeing the importance of this situation, there have been various solutions proposed. These solution lies between the idea of viewing the placement of controllers as an optimization problem and establishing connections of this problem to the fields of local algorithms and distributed computing for developing efficient controller coordination protocols.

A final concern raised in this section is the error caused due to concurrency issues because of multiple SDN Controller in physically distributed environment and their distributed nature. The solution to this problem can be performing either complete commitment of a policy update or aborting like in case of transactional interfaces or transactional databases. [25] [26].

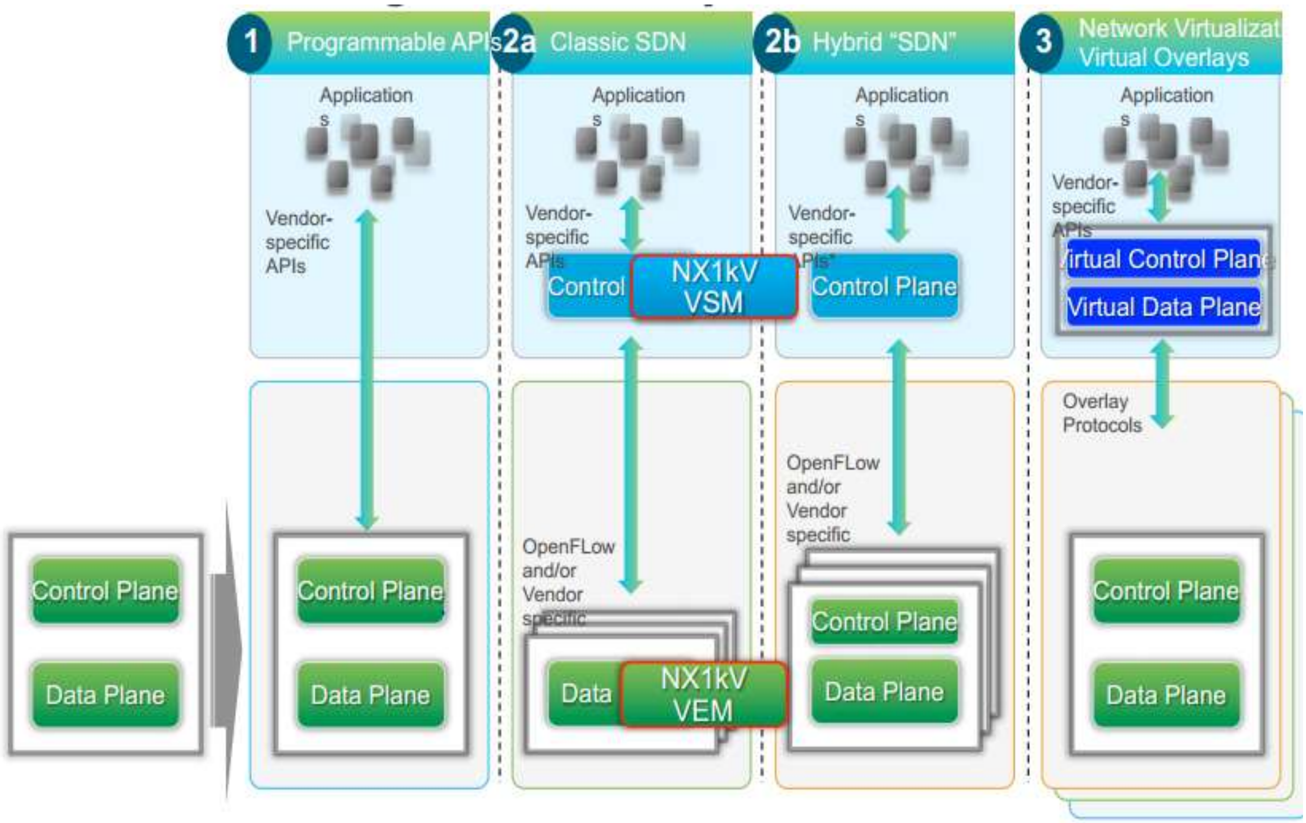


Figure 10: Different SDN Approach model [28]

Chapter- 4

Concepts of Virtualization

4.1 History

Virtualization can be defined as the technology used to run multiple operating systems (OSs) or applications on top of a single physical infrastructure by providing each of them an abstract view of the hardware. It enables these applications or OSs to share the same hardware resources while running in isolation.

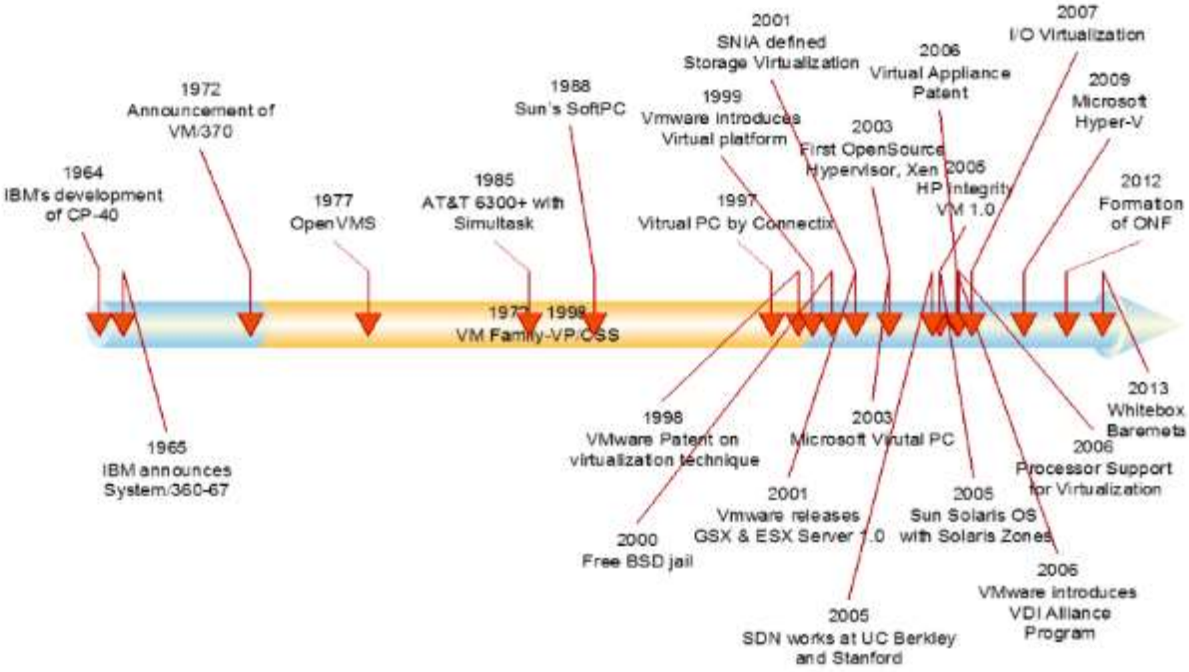
Virtualization is not a new concept, it is there since decades. It is a part of the IT revolution from infrastructure perspective. Virtualization concepts has its origin back in 1960s when IBM had the goals of implementing time and memory sharing across users and applications and they developed the CP-40 Operating System. Though CP-40 and its successor, CP-67 were not that popular but they laid the foundation for virtualization concepts that exist today.

The idea of virtualization didn’t get much attention and popularity for quite some years despite some early development in the 1960s. Until the end of the 1970s, mainframe systems were dominating the industry and were predominant computing resource available. It was sensible to share the mainframe’s computing horsepower between multiple users and applications, this is what virtualization was promising to offer. However, with the emergence of personal computers (PCs), it became possible for organizations to set up and manage their own computing infrastructure as they were relatively inexpensive. There was a rapid adoption due to this revolutionary technology. There were many advantages and benefits of this technology such as- the ability to acquire the hardware, cost of ownership, and operating system were much better compared to the computer technologies that preceded the PC. However, there were few drawback initially due to lack of development the initial applications and operating systems on these PCs offered single-user environments, and to run several applications simultaneously, effective multitasking was not efficient due to the lack of computing resources in the hardware. This created the culture of “one application per server”—or single tenant server. Additionally, there is a requirement of an isolation between departments and sub-teams within the organizations (for example, sales and marketing wouldn’t have wanted to share its data with engineering), and that that is how the realization of the need of the idea of isolated and separate computer systems that were running separate applications and used by different teams. Virtualization was far away from becoming the mainstream idea because of this huge shift in the way computing resources were acquired and used. Over the next decades, the Internet revolution in the 1990s created a demand for server farms which host various applications and databases, performing diverse functions needed to offer the Internet-based services such as Web browsing, email, and file hosting.

We can define the term server farm as a collection of many servers that are deployed and managed by an organization for delivering specific computing functions and services that are beyond the capability of individual servers. Along with these innovations there was also innovations in computer hardware which resulted in much more efficient and powerful central processing units (CPUs), faster memory access, lower prices, high capacity storage, and high-speed networks offering better throughput. All these advancements in hardware, use of dedicated servers for applications, and the increasing demand for these applications lead to development of server farms that were using many separate servers for the applications, but still these servers were heavily underutilized. As these server farms required running hundreds or thousands of servers, so they usually consumed large amount of power and take up large amounts of physical space. This results in wastage in space and power, management overhead, and the challenge of maintaining these servers translated into higher operating and procurement costs.

Once again, it was sensible to improve hardware utilization, decrease power consumption, save space, and reduce cabling requirements by consolidating these applications on shared servers. Virtualization came out of a long time of inactivity and hibernation. It could meet all the needs for these cost savings without rewriting applications or changing the end-user experience. This time around there were additional requirements from application perspective, such as a need for stricter isolation and segregation between applications, load balancing based on traffic, resiliency and high availability of applications. Virtualization technology didn’t take long to catch up with these new requirements.

The first commercial product to enable virtualization on x86 platforms was released by VMWare in 1999 and called VMWare Workstation. In 2001, it was soon followed by “VMWare ESX” for the server market. Then there were many other implementations soon after, such as Hyper-V (by Microsoft), VirtualBox (by Oracle), and open source virtualization solutions such as Xen and KVM. In 2005, Intel and AMD announced processor capabilities that would offer CPU support for hardware-assisted virtualization. Intel’s VT-x and AMD’s AMD-V played an important role in the increased adaptation towards virtualization. It took adoption of virtualization to a new level. Figure 2-1 summarizes this history behind the re-emergence of virtualization technology in the context of demand for and availability of computing resources.


1964
IBM's development of CP-40
1965
IBM announces System/360-67
1972
Announcement of VM/370
1977
OpenVMS
1985
AT&T 6300+ with Simultask
1972 1999
VM Family-VP/CSS
1988
Sun's SoftPC
1997
Vitrual PC by Connectix
1998
VMware Patent on virtualization technique
1999
Vmware introduces Virtual platform
2000
Free BSD jail
2001
SNIA defined Storage Virtualization
2001
Vmware releases GSX & ESX Server 1.0
2003
First OpenSource Hypervisor, Xen
2003
Microsoft Virutal PC
2005
SDN works at UC Berkley and Stanford
2005
Sun Solaris OS with Solaris Zones
2006
Virtual Appliance Patent
2006
HP Integrity VM 1.0
2006
Processor Support for Virtualization
2006
VMware introduces VDI Alliance Program
2007
I/O Virtualization
2009
Microsoft Hyper-V
2012
Formaton of ONF
2013
Whitebox Baremets

<u>Figure 11: Virtualization Timeline</u> [34]

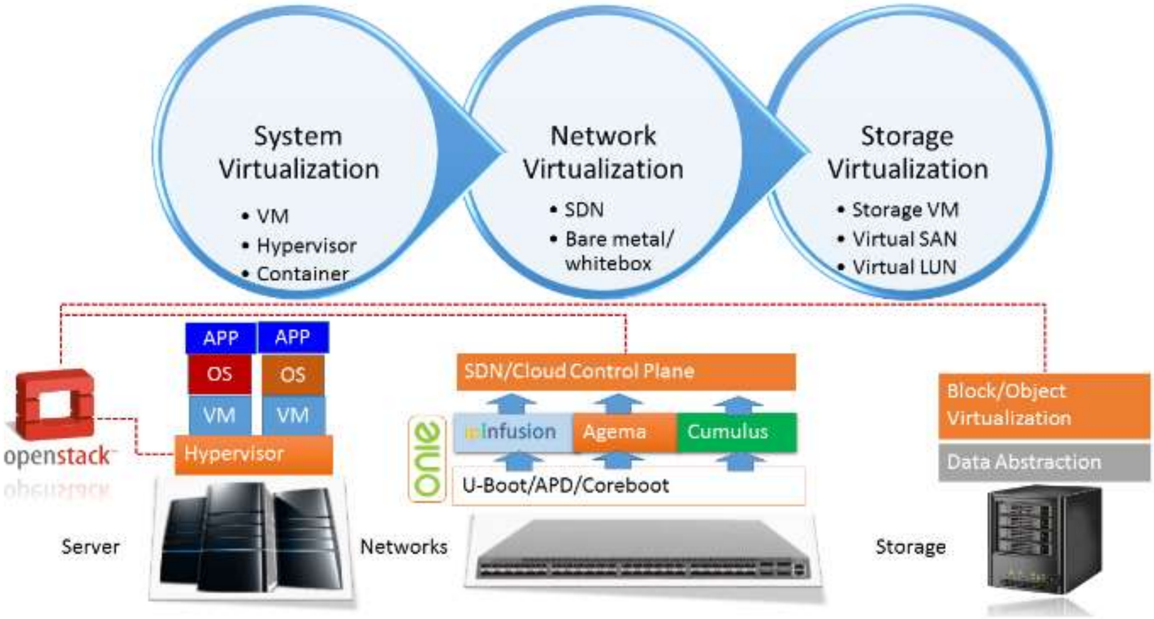
The goal behind virtualization is to offer a mechanism to share an operating system and hardware resource pool to run multiple applications without dependency on or knowledge of each other. Each application is not necessarily aware that this hardware is a subset abstracted from a bigger hardware pool, rather it is made to believe that it owns the hardware resources. The applications have segregation between their processes, disk and file-system usage, user management and networking.

With virtualization, the smaller number of servers doing the same job and hosting multiple applications simultaneously which resulted in the drastic reduction in number of servers in a data centre or server farm.

Virtualization resulted in many benefits which made it compelling technology to be adopted, benefits observed were as follows: savings of power, space, and operational costs, reduced total cost of ownership and business resiliency.

4.2 Server Virtualization, Network Virtualization and NFV

Until now we have majorly discussed only server virtualization. Rather, there are virtualization concepts in broad sections of technology as shown in figure- System Virtualization, Network virtualization and Storage Virtualization. So we can infer that the concept of virtualization was implemented in places other than servers, like networks and storage devices. Each domain has its own meaning on context and use cases.



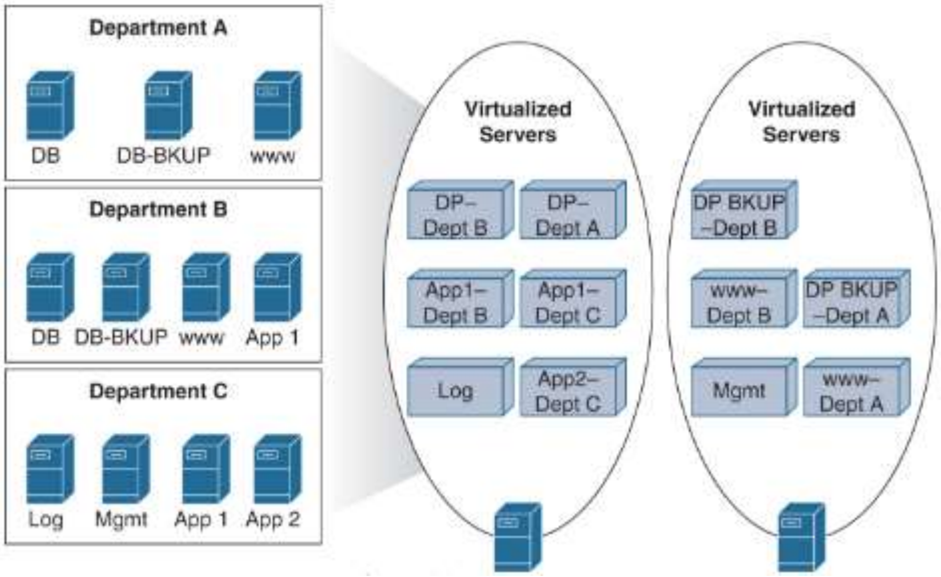
<u>Figure 12: Virtualization Concepts </u>[34]

We will majorly discuss the following three broader areas of virtualization:

- Server virtualization
- Network virtualization
- Network functions virtualization (NFV)

4.2.1 Server Virtualization

Server virtualization was discussed in detail earlier in previous topic. Virtualization of multiple physical servers providing email, database, management and web services into a much smaller number of physical servers. Design and deployment still need to be verified of any redundancy as it wouldn't be considered a good design and deployment to virtualize the primary and backup database servers on the same physical hardware. So, the physical aspect of redundancy still needs to be taken into account. The figure below shows the appropriate way to virtualization considering the redundancy aspect.



<u>Figure 13: Server Virtualization</u> [33]

Server virtualization has proven to be a very successful and efficient way to consolidate and manage resources and is a fairly mature technology now. A large number of software tools have been developed with the goal to provide fast and easy deployment of virtualized servers and also to give administrators the ability to manage and monitor their performance and optimize their utilization.

4.2.2 Network Virtualization

Network virtualization is a concept that is often confused with NFV. In reality, it preexist NFV and has little relevance to it. It does not have a relationship with the virtualization ideas that have been discussed so far. Rather, it uses the word “virtual” in a different context than server virtualization. Network virtualization is an approach in which a single physical network is logically split into multiple logical networks. These networks share the underlying infrastructure, but to the end user this sharing is not visible, and the protocols and technologies used make these networks appear to be completely independent and separate networks running on a dedicated infrastructure. The logical networks (or virtual networks), provide isolation, privacy, and network-level segregation between the networks.

An initial example of network virtualization is perhaps use of virtual LAN (VLAN). As shown in Figure below, this offers a way for a campus or office network to be split into virtual segments that share a portion of the switching and data path with each other.

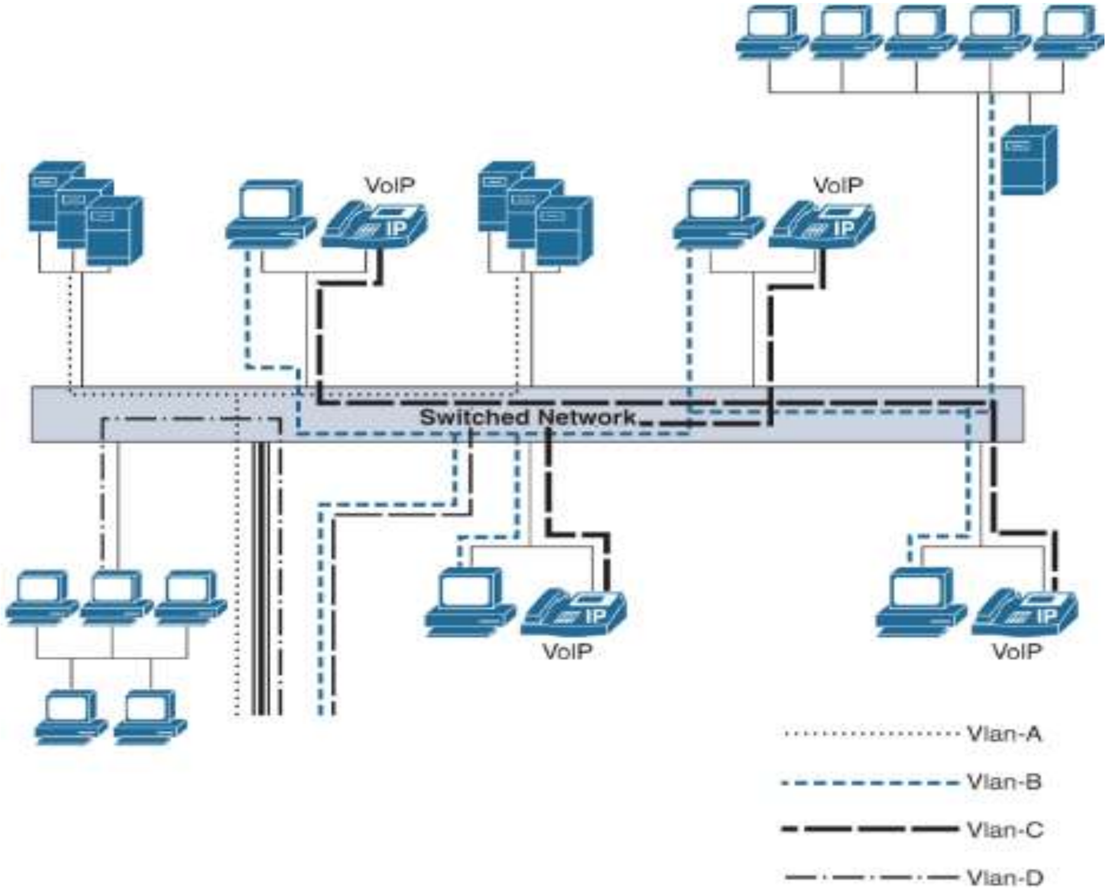


Figure 14: Network Virtualization using VLANs [33]

Some other examples of such virtual networking technologies are IP Layer 3 virtual private network (L3VPN), Virtual Extensible LAN (VXLAN), ATM switch and permanent virtual circuits (SVC/PVC). Notice the use of the word virtual in each of these technologies, because each of them was offering a way to implement a virtual network overlay on a physical network.

For Internet service providers (ISPs) this ability to overlay multiple networks on a shared infrastructure made it possible to offer many different services without the need to deploy separate physical networks for each of them. Network virtualization techniques made it possible to simultaneously run services such as broadband Internet, video streaming, and voice over IP as logically separate networks over same physical network. The major benefits were a significant savings on deployment, management, and maintenance costs of the infrastructure.

For businesses, network virtualization offered cost-effective ways to interconnect their intranet segments, small offices, and remote workers through the Internet or over the virtual private network services that they could receive through ISPs.

A typical ISP today has many separate overlays running on the network infrastructure. To carry the traffic (predominantly data traffic) for these multiple virtual networks, the physical networks infrastructure has to be provisioned with enough bandwidth as well as designed with measures to prioritize different services in case of congestion or failures. This led to many developments in quality-of-service (QoS) implementations, routing protocols, traffic engineering, etc. The details of these are outside the scope this book, but it is important to distinguish between network virtualization and NFV. Network virtualization has heavily influenced network protocols and network growth. NFV too is influencing network protocols and growth—but in a different way.

4.2.3 Network Functions Virtualization

Network functions virtualization (NFV) extends the idea of server virtualization to network devices meant to perform specific functions in the network. It's the success of server virtualization that has attracted network operators to look at NFV, and consequently push the equipment vendors and manufacturers to break away from the "one device, one function on custom hardware" and start allowing their network operating systems to run in virtualized environment. The initial white paper presented on NFV, directly referenced the success that servers virtualization has had and proposed to do the same with network functions with the goal of achieving similar benefits.

The software on network devices has historically been proprietary and custom built, while the hardware anatomy consists of a low-to-medium processing engine, disk storage, and a large number of physical interfaces for data input and output (I/O). These devices also use dedicated CPUs for processing and forwarding network traffic, and specialized memory for address lookup, such as a ternary content-addressable memory (TCAM). These packet processing CPUs are highly customized for implementing network functions such as forwarding, classification, queuing, and access control and are often implemented as Application Specific Integrated Circuit (ASIC) in the traditional networking devices.

Custom off the shelf (COTS) hardware lacks these dedicated packet processing and forwarding CPUs needed for high throughput, the specialized memory for fast lookup, and the special software or operating system that implements the network functions. With NFV, the operating system is virtualized and runs on COTS servers—the amount of computing, storage, and interface resources required for running multiple instances of these operating systems on a single server can easily be accommodated. To make up for the lack of forwarding CPUs and fast-access cache memory, special software techniques have been developed to achieve high performance while using general-purpose CPUs. Some of these techniques such as Intel's Distributed Packet Development Kit (DPDK) and Cisco's Vector Packet Processing (VPP) are discussed in later chapters.

One consequence of massive server farm growth with virtualized servers was the emergence of new architectures designed to be fault tolerant. Because these servers are relativity inexpensive, the applications running on them require 24-hour availability. Software-based management and deployment tools are used to dynamically provision, teardown, or move the virtual servers. The resulting architecture was built to expect and manage failure, and work around those failures (through re-provisioning, moving, or reconnecting the impacted services). In contrast, traditional networks using physical network devices were designed for high availability by trying to ensure uptimes through physical redundancy and overprovisioned or redundant data links. With NFV, the network designs and architecture can adapt the same ways as IT virtualization.

Since virtualization of network functions is following the footsteps of relatively mature server virtualization, it leverages a number of tools developed for that. Some examples of server virtualization tools being adapted for NFV deployments are Openstack, VMWare's vSphere, and Kubernetes.[33]

Chapter- 5

Network Function Virtualization (NFV)

Network functions virtualization (NFV) is a technology area which aims to virtualize network services that traditionally run on exclusive, dedicated hardware. By implementing NFV, network functions like routing, load balancing and firewalls are virtualized and we can say that they are packaged as virtual machines (VMs) on commodity hardware. One of the most important components of NFV architecture are Individual virtual network functions, or VNFs. Network functions virtualization (NFV) is heavily influencing the world of networking. It is transforming the networking industry towards a virtualization approach. It is transforming the way networks are designed, deployed, and managed. It is moving away the network industry from customized hardware with pre-packaged software.

Each dedicated device, when comparing to NFV.s commodity hardware, needs to be manually cabled together according to functional requirement, which is a time-consuming process. Hence, this justifies the NFV's mission to use commodity hardware. Which is important because it would be expensive for network managers to purchase and manually configure dedicated hardware devices and it wouldn’t be a logical decision, that too, for building a service chain that can link certain functions to perform a desired sequence or a program. NFV has made this easier. In NFV, Network managers can add, move or change network functions at the server level using a simple provisioned process because NFV architecture eliminates specific hardware by virtualizing network functions.

NFV allows flexibility in the process. For example, let’s say a VNF running on a virtual machine requires more bandwidth then, the administrator can either use another virtual machine on the same server to handle part of the load with its bandwidth or move the VM to another physical server. Hence, this approach helps IT department to work with agility to meet the changing business needs and goals and network service demands.

The NFV concept was originally presented at the SDN and OpenFlow World Congress in October 2012. It was presented by a group of network service providers. The aim of the service providers was simplification the process of adding new network functions or applications and making the process fast. An Industry Specification Group for Network Functions Virtualization took forward the NFV development and standards, named as The

European Telecommunications Standards Institute (ETSI). They could see many advantages of NFV. NFV was for sure beneficial for enterprises, service providers and they could see the immediate use case for it and they supported its development. Also, many corporates could see that, NFV has potential to improve scalability and better utilization of network resources. Let's see a case which was quite often request from customer to a service provider for a new function, it can be easily done in case of NFV as NFV helps the service provider to add that service by not upgrading or buying a new hardware on the customer end but by adding the service in the form of a virtual machine.

Some more basic advantages of NFV are that it consumes less power and it increases the physical space for an enterprise because NFV eliminates most traditional hardware appliances. Therefore, it can help reduce both operational and capital expenditures.

NFV is closely related to SDN. They are two closely related technology but still different. They are often used together but not always. Both these technology aims towards network virtualization and automation. We will discuss about this in further topics. [33]

5.1 Introducing NFV as part of Evolution of Networks

The history of networking and the challenges that the network industry faced played a vital role in the motivation and need behind the networking industry's fast adoption of NFV. There has been a significant evolution and improvement in data communication networks and devices. Network still struggle to cope with the demands of the changing market no matter they have become faster and more resilient with higher capacity. Due to introduction of cloud-based services, there are new set of requirements and challenges brought forward in networking industry such as the need to introduce infrastructure for those demands and to support those services to make them work efficiently. There are many examples of areas that need to be addressed, such as throughput and latency need to be mitigated in existing networks. Few of them are Mega-scale data centres hosting computing and storage, an exponential increase in data-enabled devices, and Internet of Things (IoT) applications.

Due to increasing use of video, mobile, and IoT applications, the demand for bandwidth has increased rapidly. Hence, the service providers are trying for cost effective ways to expand and scale their network services. There are many characteristics of traditional devices which makes it inefficient and create many constraints, where it has the limitation of scalability, deployment costs, and operational efficiency of the network. Hence, the operators are forced to consider alternatives that can remove the limitations.

Initially it all started with server virtualization in data centres, where this approach is already proven technology. In data centres, there are virtualized servers running on shared hardware which replaced the traditional approach of stacks of independent server hardware systems.

NFV builds on this concept of server virtualization. It just increased the scope of virtualization beyond servers that is including network devices. It is really important for networks to have important features like scalability, flexibility, manageability issues, operational cost, Interoperability and many more, Also, NFV allows the ecosystem to manage, provision, monitor, and deploy these virtualized network entities to gain these features.

The Network Function Virtualizations (NFV) is used as a blanket term which covers a lot of topics in itself, it reference the overall ecosystem that comprises the virtual network devices, infrastructure and the management tools that helps in integration of these software pieces with computer hardware. To define NFV more accurately, it is the method and technology that helps you to replace physical network devices with one or more software programs executing the same network functions that was performed by that physical device, while running on generic computer hardware. One of the example is using a software-based virtual machine in order to replace a physical firewall appliance. This virtual machine performs the same function as firewall. It provides the firewall functions on non-dedicated, shared, and generic hardware, it also runs the same operating system.

NFV helps us to implement the network functions on any generic hardware, only requirement is that it should offer the basic resources for processing, storage, and data transmission. Virtualization has developed to a great extent. Nowadays it can also mask the physical device and making it possible to use commercial off the shelf (COTS) hardware and making a more flexible choice for the infrastructure for NFV. Where, Commercial off the shelf (COTS) refers to the products or services that are developed and marketed commercially. COTS hardware are built and sold for any use case that requires these resources, for example- general-purpose computing, storage, and networking gear. It doesn’t enforce usage of an exclusive hardware or software.

In traditional network architecture, for the specific network functions, there is a hardware that is developed, customized, and deployed as dedicated equipment for that specific function.

This is why vendors are not concerned about the hardware on which their code will run. Both, the hardware and the software running on the device are fully controlled by vendors. This gives the flexibility to vendors to design the hardware and its performance factors according to the roles these devices will play in the network. For example, a device designed for the network edge need to be of lower cost so, it will be kept simpler and it will offer low availability, whereas, the other device that is designed for the network core need to be of high quality with less errors and high availability and need not be cheap and it have carrier-class resiliency built into it. So in traditional approach, tight integration of hardware and software is done to achieve most of the features for these devices. The scenario is different with NFV.

In the case of virtualized network functions, there should not be any assumptions about the capabilities of the hardware. Also, tight integration of hardware and software is not possible in NFV. Rather, NFV decouples and separates the software from hardware and try to operate hardware using software. Also, it offers the ability to implement the virtualized functionality of very specific network functions using any commercially available hardware. Virtualization of networks creates a lot new possibilities in the way networks can be deployed and managed. NFV opens up new innovation, design options. It also enables new network architectures and offers great flexibility, agility, capital and operational cost savings and scalability. [33, 29]

5.2 Architectural Framework of NFV

The architecture of NFV creates multiple touch points for management and gives them better control over hardware by allowing software to run on generic shared hardware. These software are developed by the vendors. It is contradicting traditional architecture and overcoming the disadvantages of traditional networks. The network devices in traditional approach are comparatively basic because both the hardware and software tightly integrated. The NFV architectural framework is developed to promote flexibility and manageability. It also ensure the standardization and compatibility of these management touch points created in NFV. The compatibility should be between the implementations of different vendors. In following sections we will do a comprehensive discussion on the NFV framework and the rationale behind its blocks. Understanding the framework will help us to envision the aim of NFV, flexibility and freedom of choice that NFV has to offer.

5.2.1 The need for NFV Framework

The architecture of NFV creates multiple touch points for management and gives them better control over hardware by allowing software to run on generic shared hardware. These software are developed by the vendors. It is contradicting traditional architecture and overcoming the disadvantages of traditional networks. The network devices in traditional approach are comparatively basic because both the hardware and software tightly integrated. NFV creates multiple touch points for management by allowing software developed by the vendors to run on generic shared hardware. In the NFV jargon, Virtualized network function (VNF) plays most important role. **Virtualized network function (VNF**) can be referred as a virtually implementing the network functions using virtual machines. In order to virtualize the complete network segment, there is a need for implementation of a combination of these VNFs. A particular network function is performed by each VNF. Examples of network functions are router, switch, firewall, load-balancer, etc. VNF (virtualized network function) helps to replace a specialized hardware by performing its network function using the systems or software. This function is run on a generic hardware systems.

The service providers has the ability and independence to choose an appropriate combination of vendors and functions according to their needs. Various vendors provides these VNFs. There is a need for a standardized method of communication between the VNFs. Also, the ways to manage these VNFs in the virtual environment should be standardized. All this standardization is required because of high level of flexibility and freedom of choice provided by NFV. The management of NFV needs to consider few points. They are as follows:

- VNFs can be implemented by multiple vendors.
- The need to manage the network functions- their life cycles and their interactions
- Hardware resource allocations need to be managed
- The utilization need to be monitored properly.
- VNFs need to be configured according to their functions.
- Implementation of service by interconnection of the virtualized functions
- It is important to interact between billing and operational support systems

In order to implement these management roles, a standardized framework should be defined as well as the system should be open. This standard framework is important to ensure that the VNF deployed is not restricted to specific hardware and it need not be tailored especially for any environment. A reference architecture should be offered to vendors that they can follow while implementing any VNF, to check consistency and uniformity in the deployment methodologies. Also, it needs to ensure that there should be no dependency on any vendor for the management of these VNFs and the hardware they run. It should be make sure that there need not any special adjustments required to implement the network functions in this heterogeneous ecosystem. It is important that the VNFs, hardware, and the management systems should work effectively and seamlessly within the well-defined boundaries and, this framework must provide the architectural foundations for that seamless functioning.

5.2.2 ETSI Framework for NFV

NFV was first introduced by an alliance of key service providers in 2012. It was introduced at the SDN OpenFlow World Congress. The major motivation was to reference the challenges faced by network operators while using traditional approach. One of the major challenge was to enable innovative services to their customers, which was dependent on introducing the new hardware and was costly. According to the group, the challenges associated with the following concepts are as follows: [33]

- The new equipment need to be introduced with the design changes.
- Cost associated with deployment and physical constraints of each hardware need to be considered.
- Managing and operating the new proprietary hardware and software requires proper expertise which is lacking.
- Each new proprietary equipment comes with a hardware complexity.
- This equipment comes with a short lifecycle that makes it obsolete.
- Revisiting the cycle before the realization of returns from the capital expenses and investments.

NFV was proposed by the group as a way to tackle these challenges and improve efficiency, in other words “by utilizing standard IT virtualization technology to cover as many network equipment types and consolidate them to virtualized environment. This include high volume servers, switches and storage, and locating them to Datacentres, Network Nodes and in the end user's area of functioning."[30]

There is an Internet specification group (ISG) formed under an independent standardization organization called the European Telecommunications Standards Institute (ETSI). This group was formed by seven leading telecom operators. The main aim of this organisation was to define a set of specifications that can help in bringing NFV-based network to the industry and to possibly move from the traditional network centric approach to an advanced virtualization approach. [31]

This group was formed in 2012 but formally started functioning in early 2013. They started working towards requirements definition. They started making an architectural framework to enhance and support virtualized implementation of network functions. Previously these functions were performed by custom hardware devices from vendors.

ETSI devised three key criteria for coining the recommendations:

- **Decoupling:** It refers to complete separation of hardware and software

- **Flexibility:** It enables deployment of the network functions to be automated and scalable.

- **Dynamic operations:** It provide granular control and monitoring of the state of network and use that to control the operational parameters of the network functions.

Based on the three criteria discussed above, a high-level architectural framework of NFV was established. Initially it defined distinct areas of focus. This architectural framework is commonly referred to as the ETSI NFV framework. It forms the basis of the standardization and development work for network virtualization.

At a high level, the framework supported management of VNFs, relationships and interdependencies. This framework also helps in controlling the flow of data between VNFs, and appropriate resource allocation. These roles were categorized into three high-level blocks by ETSI ISG. These blocks were named as the infrastructure block, virtualized functions block, and management block. We will discuss their formal definition below:

- **Network Functions Virtualization Infrastructure (NFVI) block:** This forms the base of the overall architecture. It consist of the hardware to host the virtual machines like computer to storage hardware and network hardware, the software to make virtualization possible as virtualization layer, and the Virtualized resources like virtual storage, virtual network and virtual computer as shown in the figure below.

- **Virtualized Network Function (VNF) block:** The virtual machines offered by NFVI are used by this VNF block to perform different virtualized network functions by the software implementation.

- **Management and Orchestration (MANO) block:** If we look at the figure below, MANO is defined as a separate block in the architecture. It interacts with both the NFVI and VNF blocks. The framework makes the MANO layer responsible for the management of all the resources in the infrastructure layer. Also, this layer creates and deletes resources and manages their allocation of the VNFs.

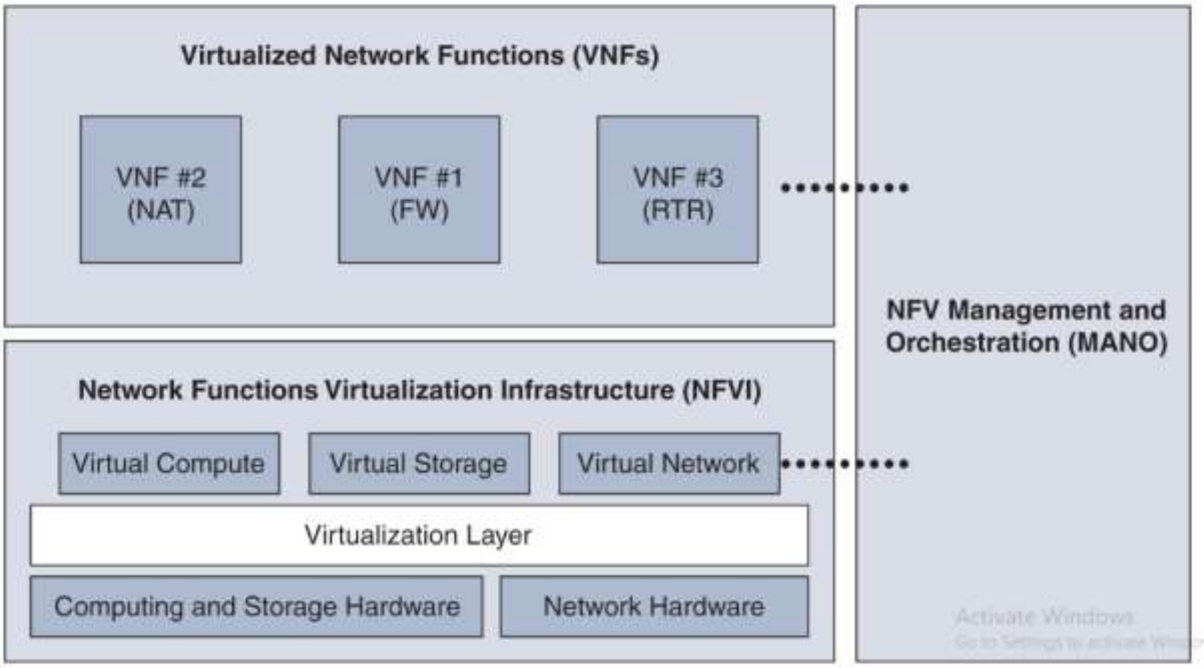

Virtualized Network Functions (VNFs)
VNF #2 (NAT)
VNF #1 (FW)
VNF #3 (RTR)
NFV Management and Orchestration (MANO)
Network Functions Virtualization Infrastructure (NFVI)
Virtual Compute
Virtual Storage
Virtual Network
Virtualization Layer
Computing and Storage Hardware
Network Hardware
Activate Windows
Go to Settings to activate Windows

Figure 15: High Level ETSI NFV Framework [33]

5.2.3 ETSI Framework Explained

In this section we will discuss and try to understand the ETSI framework and the thought process behind its high-level blocks in a better way. Firstly, we need to understand the building process that led the foundation to this framework. In the initial sections, we can understand the fundamental concept of NFV, such as virtualizing the function of a network device. VNFs plays a major role in NFV.

There is a need to deploy VNFs either as standalone entities or as a combination of multiple VNFs to implement network services. There is no need for the protocols associated with the function that is being virtualized to be aware of the virtualized implementation. As shown in the ETSI NFV framework diagram above, The VNF implementing various network functions like firewall service (FW), NAT device (NAT), and routing (RTR) in VNF layer communicate to each other without knowing their physically connection, even it need not know that they are not physically connected or are running on a virtualized technology rather than dedicated physical devices.

To run these VNFs efficiently, there is a need for a general-purpose hardware device with generic hardware resources such as a processor (CPU), storage, memory, and network interfaces. This is cost efficient as there is no need for any dedicated or custom hardware designed to run these VNF. This is done using COTS hardware. It can be comprised of any combination of the required hardware resources to run the VNFs integrated hardware solution providing rather than a single COTS device. Hardware can be shared among multiple VNFs using the Virtualization technology. Virtualization technologies have been used in data centres for some time and it has already evolved too much and is reliable. Some examples of these technologies are hypervisor-based virtualization or container-based virtualization.

Virtualization of hardware offers an infrastructure for the VNF to run upon. COTS hardware can be used by NFV infrastructure (NFVI) as a common pool of resources and create separate subsets of these resources, these subsets perform virtualized compute and allocation of storage, and network pools as needed by the VNFs.

The vendors for the VNF recommends a minimum requirement of the resources to be available for its implementation, but these hardware parameters can’t be controlled or optimized by the vendors. There are multiple resources to manage and operate at various levels in this virtualized network’s architecture as compared to the previous network architecture where the management of network is vendor specific. It has limited knobs and data points offered by vendors. It is difficult to add any new requirements or enhancements in management capabilities as it is vendors specific and hence it is possible only with vendor support. NFV has made it possible to manage the entities at a more granular and individual level. Therefore to define NFV architecture completely we need to define the methodologies to manage and interconnect these layers, automate the process and coordinate between these layers and functional blocks in an agile, scalable, and automated way.

Due to this requirement, we need to add another functional block to the framework named as MANO (NFV management and orchestration). This block communicates with and manages both the VNF and NFVI blocks. This block allocates the hardware resources to these VNFs deployment and manages the interconnections of the VNFs on the COTS hardware.

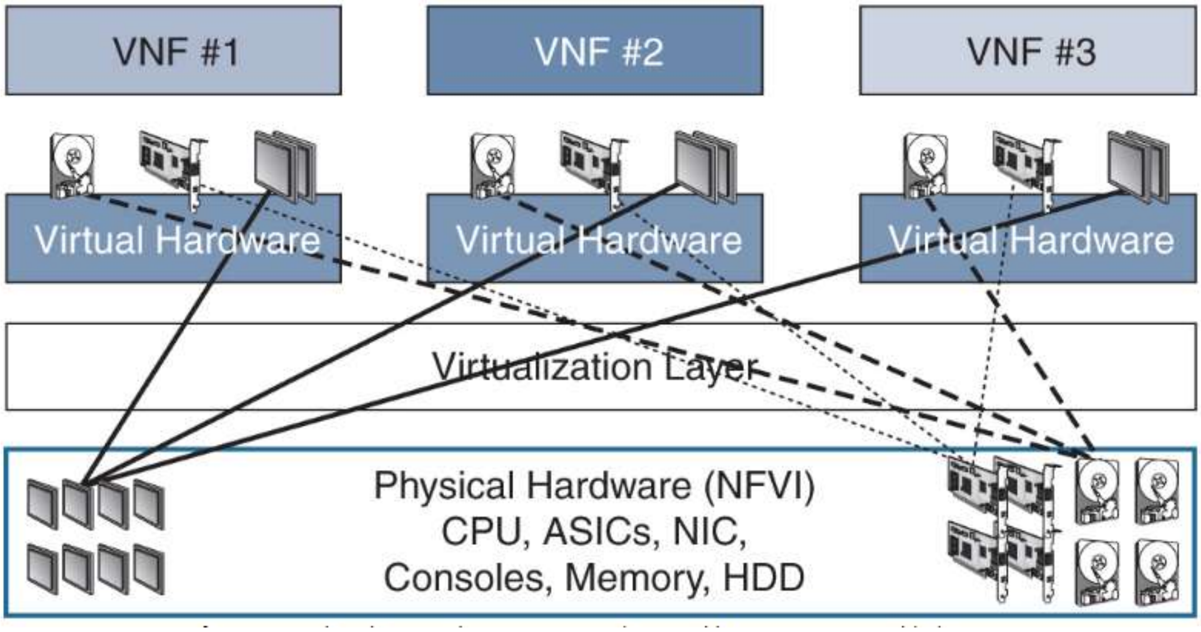


Figure 16: Virtual Computing, Storage, and Networking Resources Provided to VNF [33]

MANO block is fully aware of the utilization, operational state, and usage statistics of the network entities because it has full visibility of the entities and is responsible for managing them. Because of these roles and responsibilities, MANO is the most appropriate interface to perform these roles of gathering the utilization data for the operational and billing systems.

There is a deeper functionality and role of each of these blocks in the framework defined by ETSI. There is a defining role for each individual functional blocks has a distinct role and responsibility. There is a reason that the high-level blocks comprise multiple functional blocks like the management block (MANO). MANO block can be defined as a combination of three functional blocks: the Virtualized Infrastructure Manager (VIM), Virtualized Network Function Manager (VNFM), and NFV Orchestrator (NFVO). The ETSI framework also defines the reference points for the functional blocks as these reference points enable the interaction, communication and their functioning with each other.

Let us discuss each layer in the framework diagram in detail: [33]

- Infrastructure Layer: In the ETSI NFV framework, infrastructure block (NFVI) helps the VNF by providing the virtual hardware available as in NFV, the VNFs rely on the virtual hardware that is available to perform the network function, and these virtual hardware are made functional and controlled by software resources running on physical hardware. As shown in the figure below, this infrastructure block contains physical hardware resources like computing hardware, storage hardware and network hardware, in the middle is the virtualization layer, and at the top are the virtual resources like virtual computer, virtual storage and virtual network.

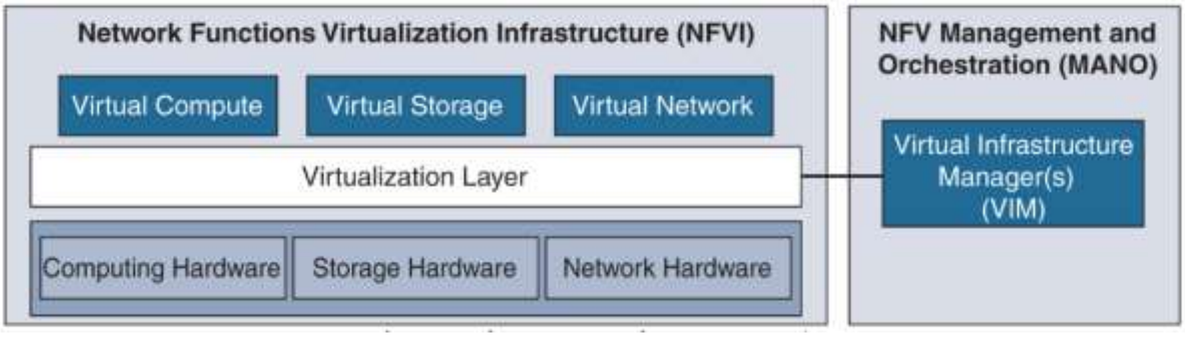

Network Functions Virtualization Infrastructure (NFVI)
Virtual Compute
Virtual Storage
Virtual Network
Virtualization Layer
Computing Hardware
Storage Hardware
Network Hardware
NFV Management and Orchestration (MANO)
Virtual Infrastructure Manager(s) (VIM)

Figure 17: Infrastructure layer of ETSI NFV Framework [33]

The physical hardware resources in ETSI framework are divided into three main categories – computing hardware, storage hardware, and network hardware. The CPU and memory are included in the computing hardware. Cluster-Computing techniques are used to group them between hosts. The second category is storage hardware. Examples of distributed storage are network-attached storage (NAS) or using SAN technologies to connect it to devices. The third category is networking hardware which include pools of NIC cards and ports that can be used by the VNFs. All these hardware are generic hardware devices and not the specific network function hardware. They are available in customs off the shelf hardware (COTS). These functional blocks are not confined to a single physical host, location or point of presence (POP) rather they can span and scale across multiple devices and interconnected locations.

The networking hardware such as switches, routers, optical transponders, wireless communication equipment, etc. are also considered a part of NFVI. The criteria for it to be part of NFVI is that they should be present within the physical location interconnecting the storage and compute devices, or interconnecting multiple locations. However, there is a pool of virtual resource that is allocated to VNF but these hardware devices are not part of it

There is another important function block that is part of NFVI and that is virtualization layer. It helps to make sure availability of pool of hardware to VNFs as a virtual machine by interacting with the pool of hardware devices directly.

According to ETSI, there is a major management functional block in the framework called as Virtualized Infrastructure Manager (VIM). It is used to manage NFVI. VIM is part of MANO (Management and Orchestration blocks). There are many responsibilities of VIM as It is has to manage the computing, storage, and networking hardware. It also has to manage the virtualized hardware as well as the software used in virtualization layer. VIM has a full visibility of these resources and the whole network. It also has to monitor their performance attributes like utilization reports and their operational attributes like power consumption, health status, and available resources. VIM directly manages the hardware resources.

It is a possibility that the instance of VIM may not be restricted to a single NFVI layer that is maybe a single VIM implementation controls multiple NFVI blocks. Also, the ETSI framework brings a lot of flexibility by allowing multiple VIMs to function in parallel and thus, they can control several separate hardware devices at one time. The physical location of these VIMs is not the issue as they can be either in a single location or different locations.

- **Virtualized Network Functions (VNF) Layer:** In ETSI NFV Framework the role of VNF layer is to implement virtualization of network function. It comprises of the VNF-Manager (VNFM) and the VNF-block which is a combination of VNF and Element Management (EM) blocks. VNF-Manager (VNFM) is the functional block that manages VNFs.

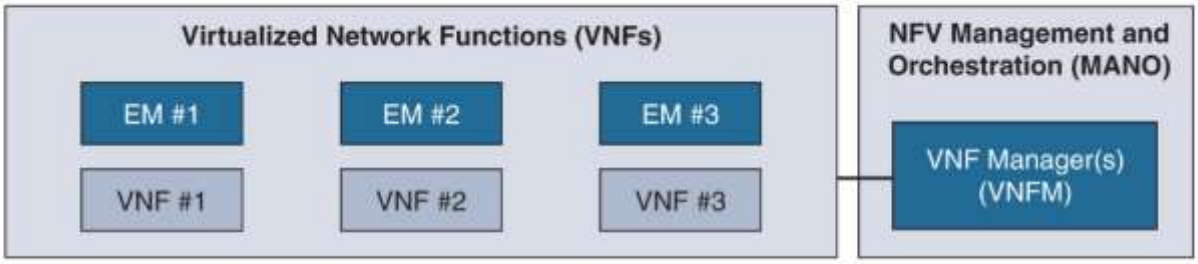

Virtualized Network Functions (VNFs)
EM #1
EM #2
EM #3
VNF #1
VNF #2
VNF #3
NFV Management and Orchestration (MANO)
VNF Manager(s) (VNFM)

Figure 18 Virtualized Network Function (VNF) Layer. [33]

There is a need to develop a virtualized implementation of a network function for flexibility purposes, so it can run on any hardware rather than being too selective about choosing a hardware. Only requirement from a hardware should be that it has sufficient computing, storage, and network interfaces that are required in a generic hardware. It is expected from a virtualized network to be unaware that the generic hardware it is running on is actually a virtual machine but the details of the virtualized environment are transparent to the VNF. It is expected that the behavior and external interface of the VNF is similar to the physical implementation of the network function and device that it is virtualizing.

There may be a requirement of a single VNF or multiple VHFs to implement the service that is to be virtualized. It is a possibility that some of the functions that are collectively implemented in a group, have dependencies on each other. This happens mostly in case while implementing a network service as a group of VNF, then the VNF needs to process the data in a specific sequence. The group is referred to as a VNF set, when a group of VNFs doesn’t have any interdependency. One of its example is mobile virtual Evolved Packet Core (vEPC) where multiple VNFs like MME (Mobile Management Entity) and SGW (Service Gateway) works together to offer their role of the functionality of vEPC but are working independently implementing their functions.

In case the network service requires VNFs to process the data in a specific sequence, then the there is a need to define the connectivity between the VNFs needs and deploy it to ensure it. This process of defining and deploying the connectivity is called as VNF-Forwarding Graph (VNF-FG). It is also called service chaining. The idea of service chaining is important in the NFV world.

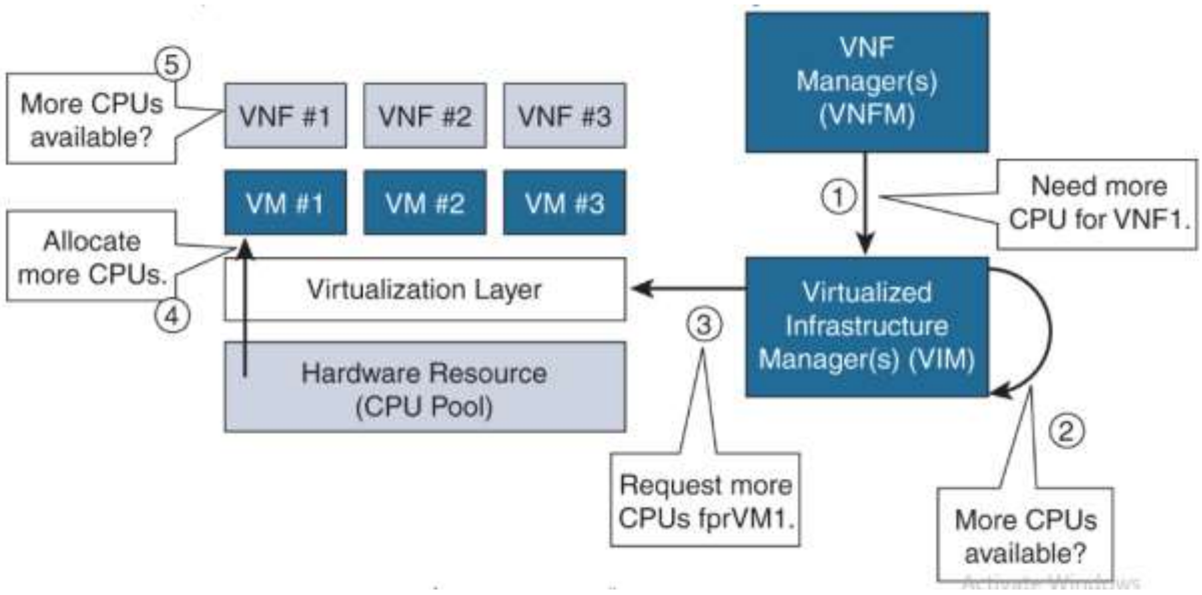


Figure 19: VNF Resources being scaled up by VNFM [33]

It is the VNFM's responsibility to bring up the VNF and manage the scaling of its resources. Whenever there is a need for the VNFM to initiate a new VNF to the current system or any changes made in VNF, like adding or modifying the resources available to a VNF, it is the VNFMs responsibility to communicate that requirement to the VIM. Also, it requests to VIM that the resources allocated to the VM that is hosting the VNF are modified by the virtualization layer. VIM can also determine if any additional needs can be serviced by the current hardware because the VIM has visibility into the inventory.

The VNFM also has the responsibility for the FCAPS of the VNFs. FCAPS stands for: fault, configuration, accounting, performance, and security. These are the five main management parameters. It is an ISO telecommunications management network mode. VNFM uses the Element Management (EM) functional block or communicate with the VNFs to manage FCAPS.

In ETSI Framework, Element Management (EM) is another functional block which is also used for implementing the management functions. Its management scope element management system (EMS) in traditional approach. EM manages the VNFs by interacting with them using exclusive methods while employing open standards to communicate with the VNFM. The FCAPS are still managed by VNFM directly, but not by EM so it can take support from the EM to interact with the VNF.

According to the framework it is not mandatory to implement a single VNFM to manage all the VNFs. There can be a possibility where any number of VNFM are managing any number of VNFs during NFV deployment because the vendor that owns the VNF may requires its own VNFM to manage that VNF.

- **Operational and Orchestration Layer:** Network operators would want to move from physical to virtual devices but during the process they don’t want to remove or lose management tools and operational and business support systems (OSS/BSS) applications. The ETSI framework doesn’t require these tools to change as part of transformation to NFV. Even though the devices are replaced by VNFs, the framework allows them to continue to use management tools and operational and business support systems (OSS/BSS), also work with the same devices that they don’t want to replace. While this is meeting the expectation and what is desired of it. There are some drawbacks to existing system and that are that it doesn’t fully benefit from NFV. To resolve this situation, steps need to be taken by service providers to enhance and evolve the existing tools and systems and further advance them to use NFV management functional blocks. By this way one can migrate the NFV benefits (like elasticity, agility, etc.) in the existing system. While this is a viable option for some providers but still this does not seems like a feasible option for other providers because these systems are built in-house traditionally that do not support the management by NFV like open platforms.

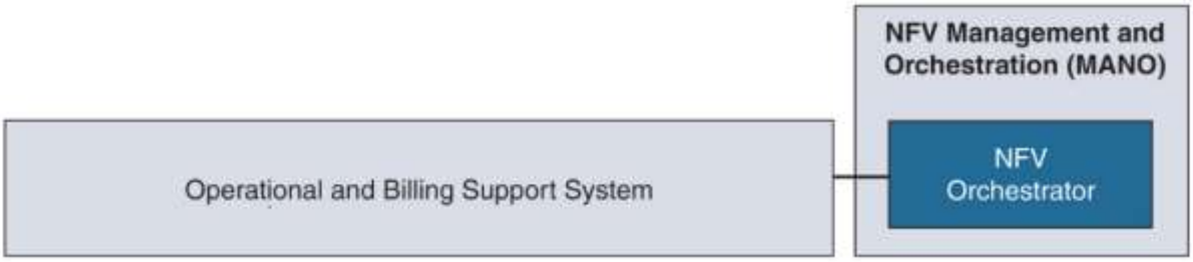

NFV Management and Orchestration (MANO)
Operational and Billing Support System
NFV Orchestrator

Figure 20: Operational and orchestration layer [33]

The solution for this offered by the ETSI framework is to use another functional block that is NFV Orchestrator (NFVO). It extends the functionality of current OSS/BSS, Its man role is to manage the operational aspects and deployment of NFVI and VNF.

The role of NFVO is not as direct as other blocks rather it functions like an additional block that is compatible with both current operating tools and VIM and VFNM. It buffers between these systems. Still NFVO has a critical and important role in the framework and that is to communicate the needed pieces of information to VIM and VNFM for implementing required service by not concentrating on the end-to-end service deployment and avoiding the bigger picture or eagle eye view of service virtualization. NFVO also works with the VIM(s). It has the full view of the resources that are being managed by them. As discussed already, there can be multiple VIMs in the system and each VIM concentrate on the functions of the NFVI resources that it is managing. NVFO can coordinate the resource allocation through the VIMs because it has the collective information from these VIMs.

The VNFM is unaware of any connection of the services between the VNFs and also it doesn't have visibility of how the VNFs combine to form the end to the service path rather VNFM is managing the VNFs independently. NVFO contains this information of the connections of the services and how the VNFs combine to form the end to the service path.

The existing OSS/BSS have a place in the framework to provide the management, but it is not a part of the NFV transformation. It is important to define the reference points between the existing OSS/BSS and NFVO and it is done by ETSI NFV framework. NFVO can be defined as an extension of the OSS/BSS. NVFO helps to manage the NFV deployment without affecting OSS/BSS.

- **NFV Reference Points:** To know the communication that must occur between the functional blocks, NFV reference points are defined by NFV. It is important to identify and define these to make sure that the flow of information is consistent across the vendor implementation for functional blocks. It helps to establish common way to communicate and share information between the functional blocks.

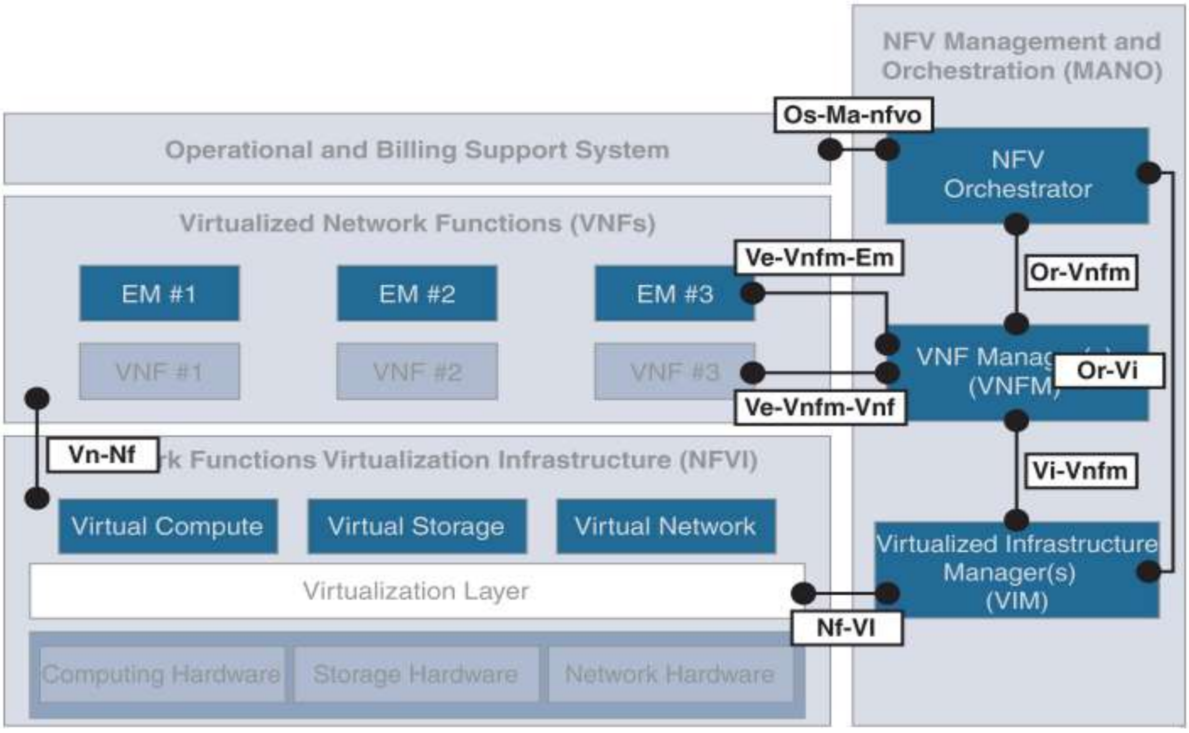


Figure 21: Types of reference points in the NFV Framework [33]

Above diagram show the types of reference points defined by the framework. Let us discuss these reference points: [33]

- **Os-Ma-nfvo:** This reference point is meant to define the communication between NVFO and OSS/BSS and this is only one reference points between NVF (MANO) and OSS/BSS. It was originally labeled as Os-Ma
- **Ve-Vnfm-vnf:** This reference point is meant to define the communication between VNFM and VNF.
- **Ve-Vnfm-em:** This reference point main roles are management of VNF lifecycle, managing faults and managing configuration and other roles.
- **Nf-Vi:** This reference point is meant for exchanging information between VIM and NFVI’s functional blocks.
- **Or-Vnfm:** This reference point main role is communication between NFVO and VNFM. It also helps in functions such as VNF instantiation and other VNF lifecycle-related information flow.
- **Or-Vi:** This reference point helps NFV orchestrator (NFVO) to communicate directly to VIM, It helps in management of the infrastructure used in the NFVI.

- **Vi-Vnfm:** This reference point meant for standardization of exchanged information process between VIM and VNFM.
- **Vn-Nf:** This reference point functions to communicate performance and portability needs of the VNF to the infrastructure block. This reference point doesn’t have a management functional block as one of its boundaries and this is the only such reference point.

5.2.4 NFV Framework Summary

The goal of defining the ETSI framework is because it plays an important role in network virtualization. We discussed specifically the individual functional blocks and the reference points as the role of this framework is standardized implementation. Similarly, the reference-points helps to define the interdependencies and communications paths. Also, they are meant to be open and standard methods.

This framework has bring in the flexibility and sense of independence as vendors can independently develop these functions and deploy them to other functional blocks developed by other vendors and it will work smoothly. The network can have a heterogeneous deployment of NFV as much as the framework allows the implementations scope and role. This will result in the flexibility for service providers to choose between vendors for different functional blocks. This has changed the way traditional networks were deployed, where service providers were restricted to the hardware and software provided by their vendors which had many limitations attached to it. They were actually challenging the operational needs also and it had interoperability concerns. NFV comes with the solution to these limitations and offers service providers to use NFV functional blocks and hardware with any combination of vendors. This way they can deploy a scalable and agile network. There may still be some interoperability issues related to higher-level protocol that may arise during implementation by various vendors. For example, there may be some issues in BGP implementation by a vendor of one VNF while it is peering with another VNF developed by a different vendor. A standardization process exists and will continue to play a role for these types of interoperability issues. The standardization of NFV building blocks is still an ongoing effort.

5.2.4.1 Virtualization and NFV

NFV is the result of adaption of server virtualization to networking. A considerable amount of focus was given here to go through the fundamentals of virtualization, as this forms the core of the NFV architecture. Looking back to the ESTI architecture, the NFVI block has the virtualization layer, which is implemented via a hypervisor (if using VNF as virtual machines), or LXC/Docker (when using containers for VNF). The concept learned in this chapter is applicable to this virtualization layer of NFVI Block.

The evolution in server virtualization has a direct influence on the way VNF are deployed and implemented. For example micro-services with containers is being explored as an option to further optimize NFV deployment efficiency, since container reload and deletion times are much shorter. In the long run, this option may justify a trend towards use of containers in NFV rather than a virtual machine.

Similarly, some enhancements are being pushed into virtualization due to its use by NFV. For example the VNF interconnectivity uses the same tools that Virtualization uses to interconnect the virtual machines or Containers. This has pushed for development of more efficient virtual switches, as well as optimized packet-processing techniques at the kernel level and network interface card (NIC) level, such as Intel’s Data Plane Development Kit (DPDK).

5.2.5 Designing NFV Networks

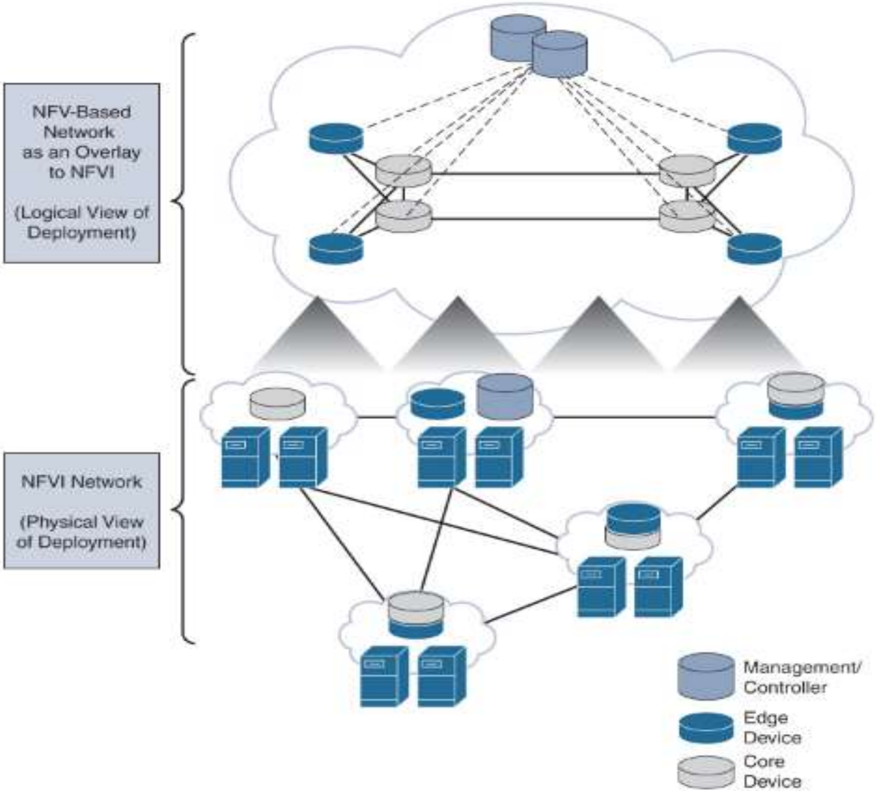
In traditional network designs, the requirements limit the scope of the design to a limited range of devices as their design hardware centric, and the designs need to be adjusted to work with the limited options available in case the hardware that is available from vendors doesn’t meet the design requirements (lack of features, scale, or capacity). The resulting network design is tightly wrapped around the hardware devices and their capabilities. Hence, making the network rigid and difficult adapt to any future changes that may be required for introducing new services.

NFV-based design helps us to remove this restriction of hardware centric design and provide flexibility that is not restricted by networking hardware. Additionally an NFV design has incorporating elasticity, scalability, and software-centric approach which helps to meet changing network requirements. NFV helps the resulting network to avoid longer lead times which has been plaguing new service adoption in the traditional networks as it has the ability to offer speedy transition and agility.

A different approach is required for the design and deployment for NFV networks to utilize the capacities of NFV. The network functions are decoupled from the hardware, so the choice of VNF types and vendors doesn't have a correlation with the design of the physical infrastructure. Vice versa is true for the physical infrastructure, as it can be designed without influence from the VNFs it will host and run. Another design dimension is added by the considerations for management and deployment of the network functions. Each of these blocks are independent of other block i.e. they are designed individually and fairly independently since they are influenced by different factors and involve a different thought process.

Designing Networks using Network Functions

The network and its functional blocks such as VNF can be viewed as an overlay on top of this infrastructure after the NFV infrastructure is in place, as shown in Figure below. The design of the network is therefore independent, flexible, and free from constraints of the physical hardware. The network design could be focused purely on the services that need to be implemented and offered by the VNF, and any needed computing, storage, or networking resources can be assumed to meet the design requirements.


NFV-Based Network as an Overlay to NFVI
(Logical View of Deployment)
NFVI Network
(Physical View of Deployment)
Management/ Controller
Edge Device
Core Device

<u>Figure 22: Network Overlay on top of Infrastructure.</u> [33]

In case of VNF, the design and deployment takes a software-centric approach so the core functionality of the network is implemented in software which helps the network functions (VNF) can be add, scale up and down, remove, and relocate purely in software. The VNFs are also expected to support open APIs, which allows any third-party orchestration and management tools to control the placement and life cycles of the VNFs. The orchestration tool instructs the virtualization layer to interconnect the VNF in any desired order, and new VNFs can be instantiated and added to the data or control traffic's path on the fly. Programmability of the VNF via the open API leads to software-defined networking (SDN).

First let us understand the role of Open API. Open API support by VNFs implies that the VNF supports and documents APIs that provide a way for configuration, monitoring, and management needs in addition to the traditional way of configuration and management (typically using a text-based command line interface or CLI).

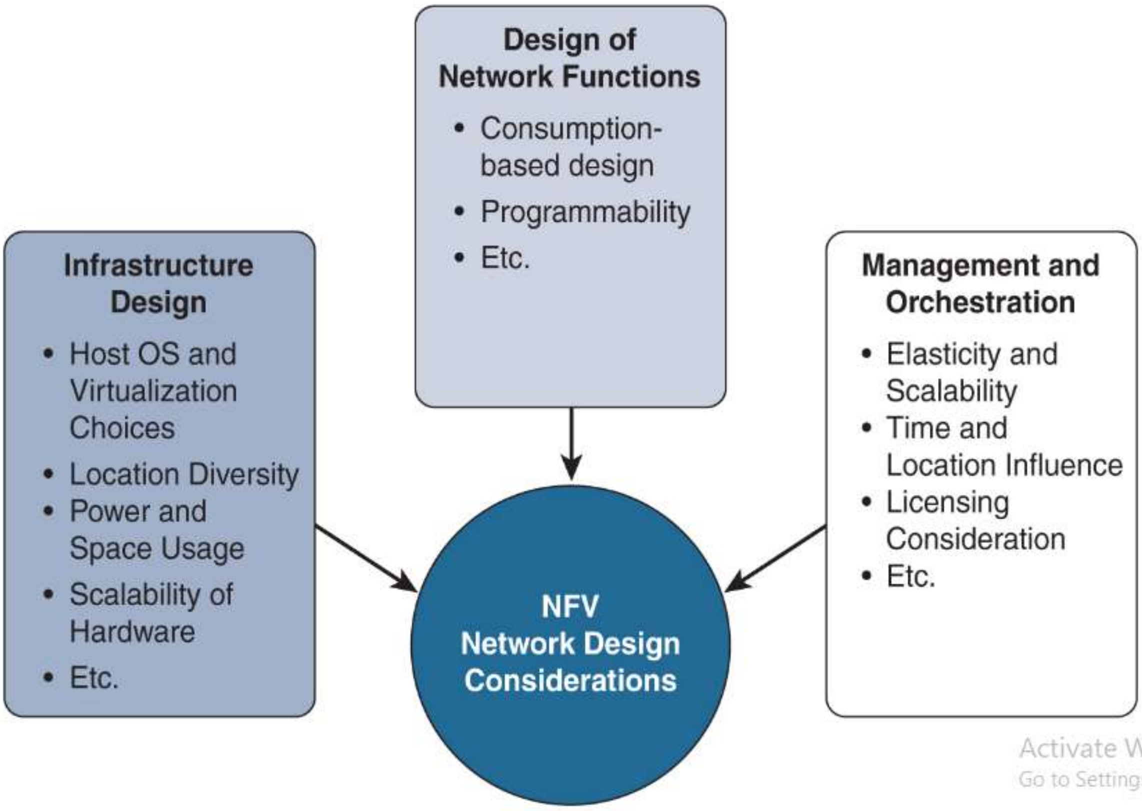

Design of Network Functions
• Consumption-based design
• Programmability
• Etc.
Infrastructure Design
• Host OS and Virtualization Choices
• Location Diversity
• Power and Space Usage
• Scalability of Hardware
• Etc.
Management and Orchestration
• Elasticity and Scalability
• Time and Location Influence
• Licensing Consideration
• Etc.
NFV Network Design Considerations

Figure 23: NFV Network Design Consideration [33]

The criteria for implementing a flexible and open platform for VNFs are discussed in the following subsections.

- **Scalable Hardware Resources**

The infrastructure hardware should be scalable and have the flexibility to scale up and down if needed. It is not always possible to predict correctly the hardware resource requirements that may arise as the infrastructure is designed fairly independently of the overlying networking layer. Even though operators avoid the need for future updates by deploying abundant hardware in the initial deployment but it is still possible that the deployed hardware might prove insufficient for growing needs and require upgrades. To handle these situations, the operator should choose hardware equipment that should not impact the current virtualized applications and VNFs that it may be hosting while it can be easily scaled. This means that the servers chosen by the operator need to be capable of scaling up the hardware resources such as network interface cards (NIC) and memory.

- **Hardware Cost and Capital Expenses**

One of the important selection criterion is the cost of the hardware. Custom off the shelf (COTS) hardware is considered as the best way to achieve the optimal price point for hardware, but big vendors like Cisco, HP, IBM, and Dell have been offering server products and pricing them competitively against COTS. Operators may tend to choose these commercially available servers, since the vendor built hardware would have been tested for any compatibility issues between components, and the servers are backed by the support contract from the vendor. The choice impacts the overall capital expense for the deployment. This choice is also influenced by different factors such as reliability of the network and support available to resolve possible issues.

- **Choice of Host Operating System and Virtualization Layer**

The host operating system (OS) and the virtualization layer (hypervisor) must be compatible and integrate smoothly with the deployed hardware. They should be compatible enough to offer a stable base to build the rest of the structure. When using COTS or a commercially available server, there is a wide range of choices for a host OS, hypervisor, and even orchestration tools. Following factors should be considered while selection: [33]

- Type of technical support available for these software pieces
- licensing costs
- Procurement costs

- The roadmap for future support
- Upgradability support
- Stability
- Ability to interact with open source and commercially available tools

It is a design decision to find the right balance between all of these factors. Some operators may prefer fully bundled software solutions from companies like VMware, RedHat, or Canonical. In which case, the operator will incur licensing costs but will be comforted by the fact that the product has a proven track record, technical support structure, and has a secure future with a clear roadmap and upgrade path. Some operators may find confidence in choosing from the other side of the spectrum for an open source, freely available OS like Ubuntu or CentOS running open source hypervisors such as a Kernel-based Virtual Machine (KVM) where they can eliminate the licensing costs and rely on in-house, third-party, or community-based support structures for future growth and issue resolution.

➢ Efficiency in Power and Space Usage

Considering the in the long-term operating expense of the network, we need to consider power and space requirements for the infrastructure hardware. This becomes much more critical in parts of the world where real estate expensive and power tariffs are high. To understand the criticality of the space and power efficiency issues, let us see the deployment scale of the data centres being built today to host virtualized servers. These data centres are spread over many acres of land (or multiple floors of high-rises in densely populated locations) and consume hundreds of megawatts of power. Any improvements in the amount of space and power consumption for the individual servers can have a big impact in the operational cost of the NFV point of presence (PoP).

➢ Redundancy and High Availability

In traditional networks, it is assumed that the network functions can be lost if the device performing that function goes down, possibly due to even a single component failure which is mitigated in NFV design consideration. In NFV, the chances of loss of a network function due to a single component failure are highly minimized because of implementation of high availability and redundancy. For example, if a router is deployed as a VNF on a server using a Redundant Array of Independent Disks (RAID) technology, then the failure of one of the disks doesn’t have any impact. Building redundancy in the infrastructure is cost effective because NFV infrastructure is shared, and multiple VNF are benefiting from this simultaneously.

Chapter 6

SDN Correlation with NFV

Basically, SDN and NFV are two different and independent innovative technologies. However, both these technologies heavily benefit from each other and support their mutual adoption as many of SDN's goals are shared by NFV.

In traditional network devices built by the vendors, the control, data, and hardware planes are tightly integrated together. Hence, scalability becomes impossible. This also doesn't offer flexibility to implement new services or the agility to absorb changes. SDN and NFV play a role in breaking this bonding between the control, data, and hardware planes in two different dimensions. SDN uses network abstraction for an independent control plane to manage, manipulate, and monitor the forwarding plane and its main focus is separation of the control plane from a forwarding plane. On the other hand, NFV facilitates the use of generic hardware to run the software implementing network functions and its focus is on decoupling the network function from the vendor-built hardware.

Both SDN and NFV offer different approaches to flexible, scalable, elastic, and agile network deployment. The principles of SDN can be applied to NFV by separating the control plane function from the forwarding plane as well as virtualizing the network functions. The below figure reflects their relationship with each other over an application and according to the scenario in figure, NFV uses commodity hardware and implements the network function's forwarding plane, while SDN controller extracts the control plane function

Applications may help to stick this relationship together and maximize the benefits of both technologies to offer a new networking landscape. The combination of these three areas- SDN, NFV and Application offers the perfect features to meet cloud scale requirements of on-demand expansion, optimization, deployment, and speed.

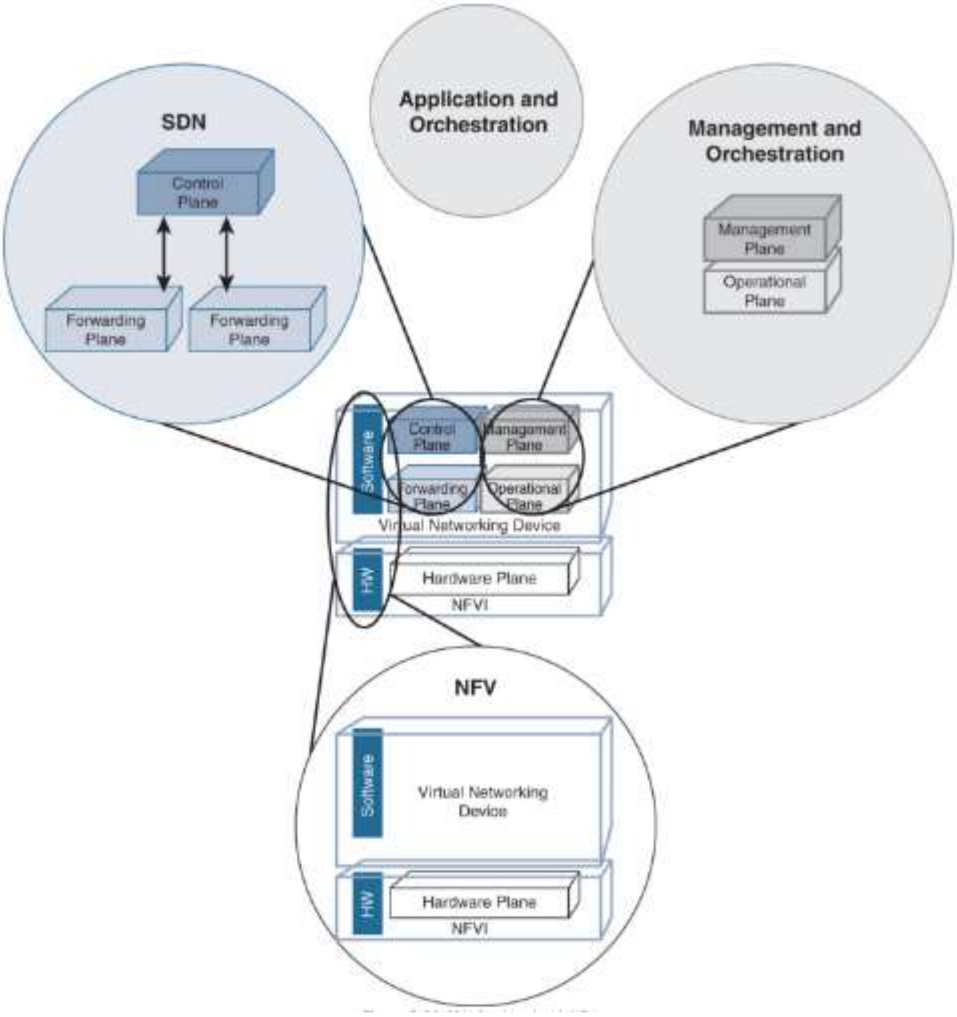


Figure 24: SDN Correlation with NFV [33]

These Feature helps in development of these technologies and it attracts service providers towards this direction to reap the maximum advantage for their business and rapid implementation of new services for end users. There has been an industry shift towards these technologies as major vendors and new entrants are also supporting this trend to become the leading providers for the new market opportunities. There are a number of projects that are being evaluated to collectively use open source tools available in both SDN and NFV. One such example is the Open Networking Lab (ON. Lab) and AT&T joint project called Central Office Re-Architected as Datacentre, or CORD.

Chapter 7

SDN Use-Cases

Initially SDN was introduced as a solution to solve data-centre scalability and traffic control challenges. But with time, this new technology made its way into other segments of the network, finding different use-cases and applicability in other areas as well. The protocols and the technology used by SDN vary based on the solution it brings to the challenges in each segment.

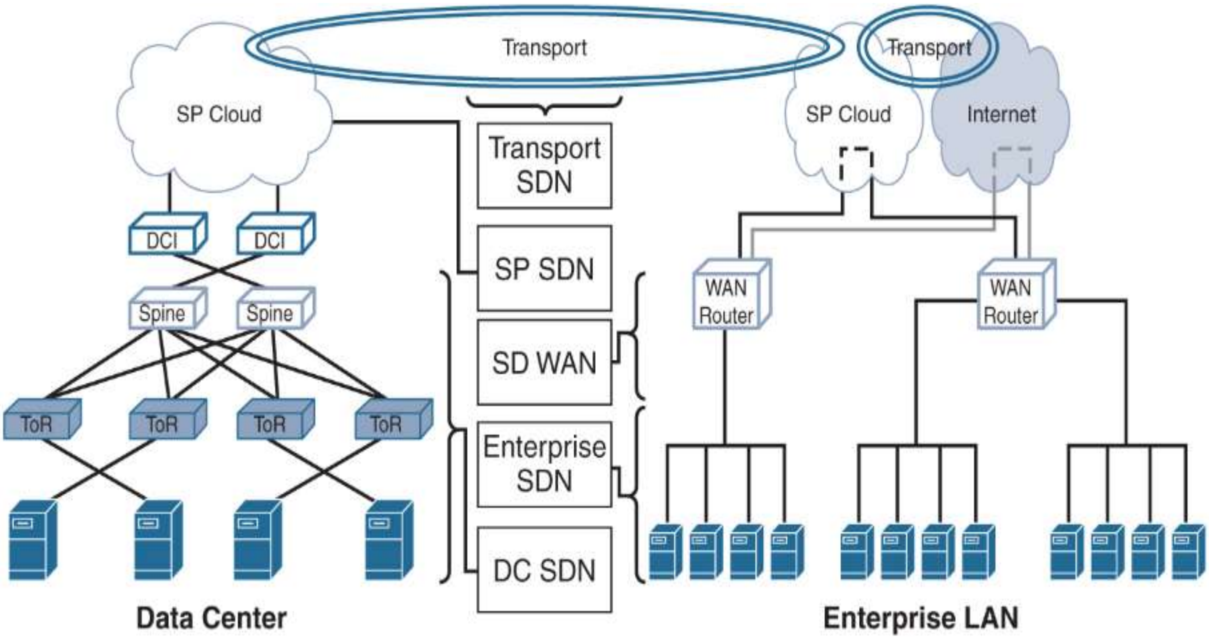


Figure 25: SDN Domains [33]

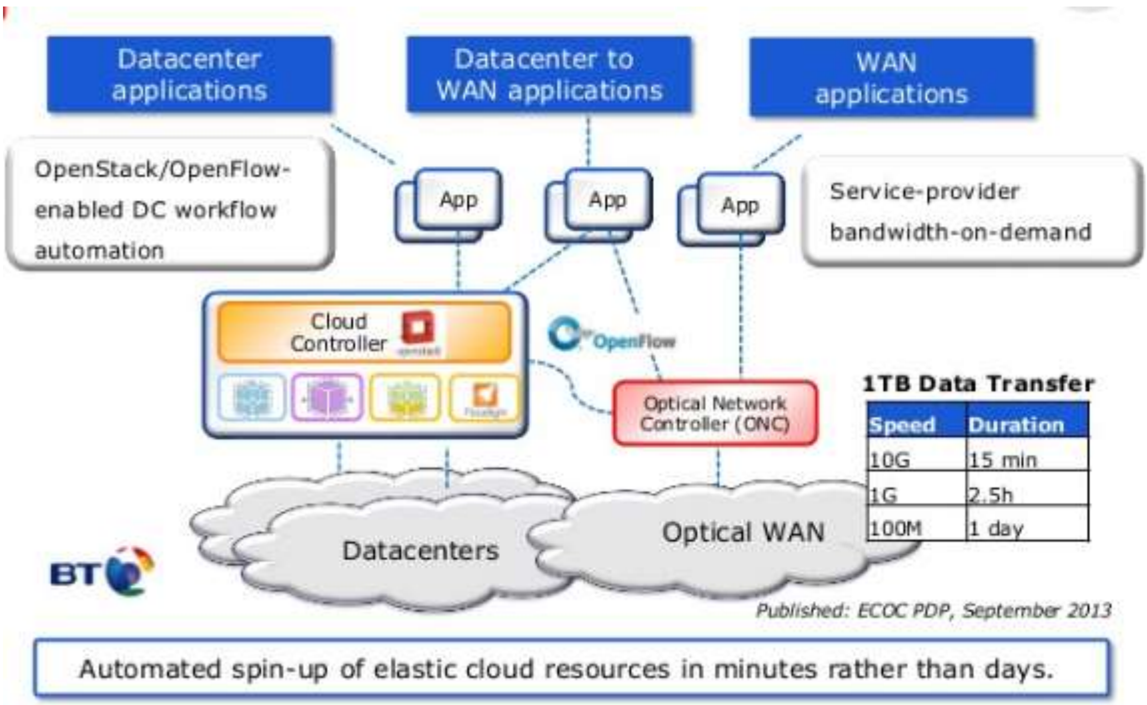
7.1 SDN in the Data Centre (SDN DC)

Data centres have been around in the industry since the mainframe era but they have seen exponential growth in size and capacity in the last decade. This growth has been pushed by the emergence of the Internet, cloud, and the consequent trend of providers to maintain an online presence to match consumer demands. Today we have large-scale data centres housing thousands of servers, deployed on dozens of acres, and consuming multi-megawatt power.

Problems and Challenges

The growth of these data centres has been a driving force behind the virtualization of servers. Virtualization helps in bringing efficiency in space, power, and cost. There were new challenges while implementing the network architecture to interconnect these virtual servers. One of the challenges has been the limitation of virtual LAN (VLAN) scalability limit (i.e. 4096). The virtual servers are typically in the same Layer 2 domain. VLANs are used to segregate between virtual servers to support multi-tenancy. Additionally, there has been a rise in use of cloud-based hosting by businesses which created the need to span that business's VLAN domain across multiple data centres, hence tightening the available VLAN space.

To relieve the above stated constraint, the Virtual Extensible LAN (VXLAN) protocol was introduced. VXLAN offers use a Layer 3 network to provide a Layer 2 adjacency between the virtual servers. It solves the scalability problem by creating an overlay network using the VXLAN IDs that can scale up to 16 million segments. However, it brings a new challenge—to manage, monitor, and program this entire overlay network.


Datacenter applications
Datacenter to WAN applications
WAN applications
OpenStack/OpenFlow-enabled DC workflow automation
App
App
App
Service-provider bandwidth-on-demand
Cloud Controller
OpenFlow
Optical Network Controller (ONC)
1TB Data Transfer
Datacenters
Optical WAN
BT
Published: ECOC PDP, September 2013
Automated spin-up of elastic cloud resources in minutes rather than days.

Speed	Duration
10G	15 min
1G	2.5h
100M	1 day

Figure 26: SDN Data Centre Interconnect [35]

SDN Solution

The VXLAN overlay network’s end points also called as VXLAN Tunnel End Points or VTEPs, are either on the top of the rack (ToR) switch or on the host’s virtual switch. In either case, the VTEPs need to be programmed and associated with the tenant’s virtual machines. The Orchestration tools are used to deploy virtual machines in these massive data centres. Examples of such tools are OpenStack, which helps to deploy VMs in an automated fashion. Therefore, these virtual machines can be deployed on any of the physical servers, but an additional mechanism is required that should have visibility into the entire network for its connectivity to the rest of the network using VXLAN. This is where SDN plays its role, as it has a view of the network and can coordinate with the virtual development tool to program the forwarding plane (which is on the ToR or virtual switch) for the VTEP and VXLAN information. The SDN controller communicates to the switch to create the VTEP interface based on the VMs that are provisioned on servers that are served by switch. It is a possibility that VTEP information may need to be reprogrammed or deleted because the virtual machines may move between a physical servers or get torn down, this is also taken care of by the SDN controller.

7.2 SDN in Service Provider Cloud (SP SDN)

At a high level, we can classify routing devices in service-provider (SP) networks as provider edge (PE) and provider (P) devices. The PE-routers are using large number of interfaces with specific features on these interfaces for classification, QoS, access control, failure detection, routing, etc. because they are directly connected to the customer networks. These routers carry a sufficient amount of customer routing information and ARP caches, forming the perimeter to the service provider network. These devices further uses high bandwidth upstream links to aggregate their traffic to the provider (P) routers.

The P routers require the use of high bandwidth links between each other that are spread across various geographically distributed POPs. In most Service Provider Networks, common core links and core routers are used to offer common services such as voice, video, data, and Internet. These links carry heavy traffic data converged from a large number of customers that these providers may be serving, and any disruption to the high-bandwidth links could impact a high number of consumers. The links and the core routers that these links are interconnecting are deployed with physical redundancy to avoid failures and offer carrier-class availability.

Problem and Challenges

With redundant links, nodes, and paths available to the SP traffic, often the shortest available path between the nodes may not be the best path for cost-per-bit or may not be capable of carrying all of the traffic at once. That is why it is a common practice among SPs to use traffic engineering techniques to steer traffic towards specific paths based on criticality, cost, delays, and network state. This helps service providers in cost optimization and better performance.

As mentioned, the preferred traffic path may not be the optimal routing path (either because of cost, latency, bandwidth availability, etc.), and to meet the need, Specific traffic engineering techniques are used for overriding routing protocol behavior. Over the years, the most common technology used for achieving this goal has been MPLS Traffic Engineering (MPLS-TE). Segment Routing Traffic Engineering (SR-TE) has also been offered for this purpose in recent times. In both these protocols, the end-to-end view of the network link bandwidth, link preference, shared failure groups (such as links that may be sharing the same transport gear), switching information for the engineered traffic-path, etc. is not available to each node. Special protocols or protocol extensions are used to coordinate between the nodes for exchange of these information and decide the full traffic engineering paths. There is a computation of path and the decisions at each node which require data to be kept on each node and there is overhead which takes away device resources, as it is CPU intensive and consumes memory, as well as requiring end-to-end coordination due to the distributed nature of the implementation.

Another challenge in SP is when an event of potential failure occur and its scope of impact to the network and services. Although, there are mechanisms like Fast Re-Route (FRR) that can be implemented but the overall network design can become quite complex and hard to optimize when coupled with capacity planning and QoS guarantee requirements.

SDN Solution

The challenges discussed above, like managing and designing traffic engineering across the network can be solved in an optimal and efficient way through the use of a central controller that has a wide view of the entire network’s link state and can track the bandwidth allocation and allow the controller to handle the decision-making process.

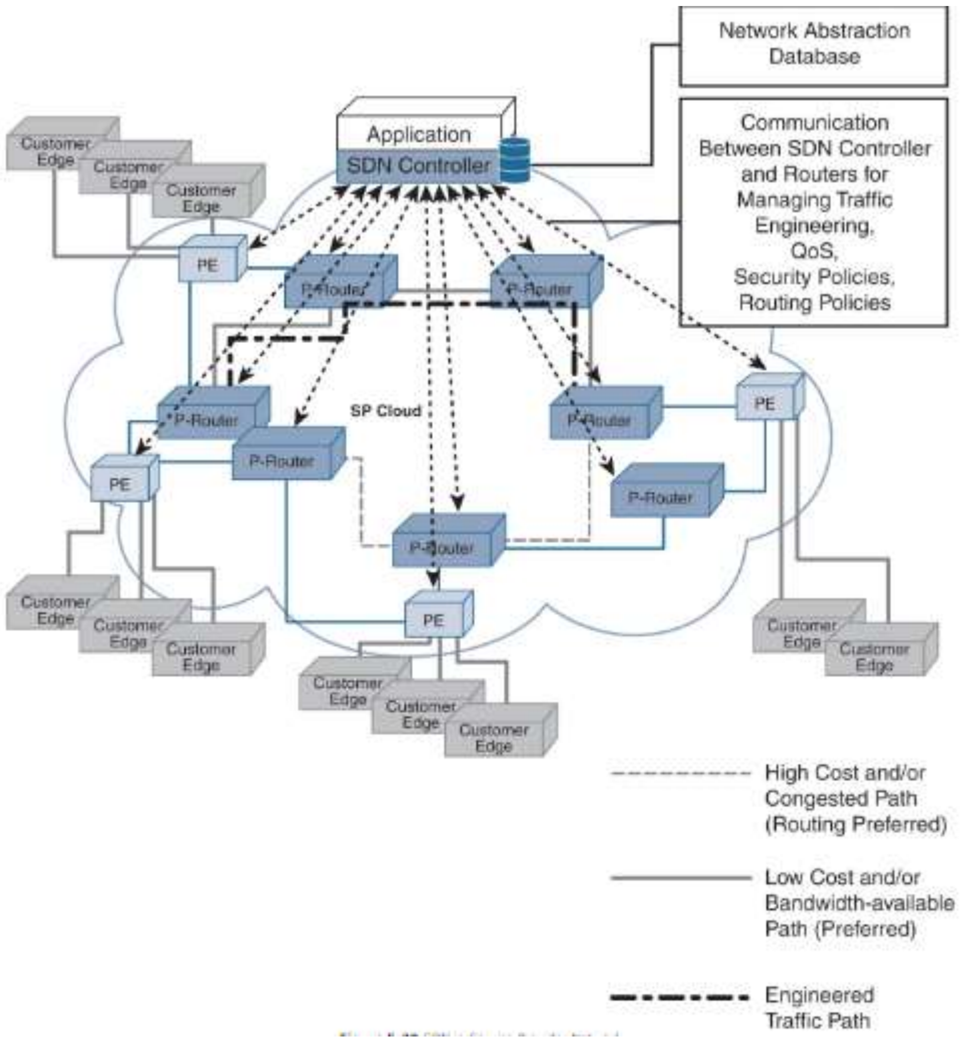

Network Abstraction Database
Communication Between SDN Controller and Routers for Managing Traffic Engineering, QoS, Security Policies, Routing Policies
Application
SDN Controller
Customer Edge
PE
P-Router
SP Cloud
High Cost and/or Congested Path (Routing Preferred)
Low Cost and/or Bandwidth-available Path (Preferred)
Engineered Traffic Path

Figure 27: SDN in Service Provider Cloud [33]

This role is perfectly played by SDN as it provides an appropriate and optimal solution. In the SDN-based solution, there is not much memory and CPU resource overhead as routers do not need to make the decisions or keep the database required for these decisions. The central SDN controller plays a wider role than just the basic decision-making criteria based on traffic and link utilization. The controller can be designed to use policy-based traffic re-routing to interact with a higher-level application. Examples of such policies are pre-emptive traffic re-routing before the maintenance window, changing traffic direction based on time of day or special events, or dynamic change of bandwidth allocation for specific traffic flows to meet temporary requirements.

The central controller can also play a major role in managing the feature-rich PE routers. On the basis of customer Service Level Agreement (SLA) requirements, the new customers that are provisioned on these routers could be configured to have consistent QoS, security, and scale and connectivity experience. The changes can be easily and consistently pushed down to the entire SP network edge routers from the centralized SDN controller in case there is any requirements change at some point (for example, the customer requests a specific data stream to have higher preference).

Other benefits of an SDN-based solution are critical features in a business criticality such as SP network security and high availability. For example, The central controller can be used to deflect the DDOS attacks and other attacks away from the standard routing path and redirect it towards either a central or distributed set of scrubbing devices, protecting the SP infrastructure in case a provider network is experiencing a volumetric distributed denial-of-service (DDoS) attack (either to the customers it is hosting or to the SP network itself).

7.3 SDN in Wide-Area Networks (SD-WAN)

Enterprise and business customer networks are spread over wide area network and different geographical regions, with multiple branch offices connecting to the head office locations. These sites cannot connect these offices within private network so they utilize dedicated leased wide-area network (WAN) circuits, such as T1 or T3, or dedicated lines from a virtual private network (VPN) service provider to connect between different offices. These dedicated links or services come are expensive and increase the operational expense for the networks. Companies have been shifting their connectivity towards secured Internet links, with the help of new technologies such as Dynamic Multipoint VPN (DMVPN) or MPLS VPN in order to reduce the cost overhead. Shared internet connection can be used for the enterprise’s private traffic by dynamically setting up an overlay network with added encryption for data protection. SLA cannot be guaranteed by Internet Connectivity. Also, despite the encryption, it may not be considered suitable for exchanging highly sensitive corporate data. These are the reasons that this approach doesn’t completely eliminate the need for private WAN links, despite the fact that it reduces the bandwidth requirements on the dedicated link yet. Based on the SLA required or the sensitivity of the data, the traffic can be sent via either the Internet link or the dedicated WAN links.

Problems and Challenges

In large-scale WAN deployments, there are hundreds of sites. So, it becomes a complex task to maintain interconnectivity between the branch offices while still utilizing split connectivity of dedicated and interlinked links. There is a need of a policy that can decide which traffic types can be deflected to the Internet link and it needs to be managed on each location as well, and this policy at each location can become a management overhead. It is also not easily possible to optimize these policies based on real-time measurements or making them dynamic with frequent changes. There were benefits like cost and performance benefits, but achieving this result wasn’t a feasible goal without the presence of a management system that could centrally manage the flows and configure the WAN routers.

SDN Solution

WAN network can be deployed using the SDN model of centralized network topology with multiple links of connectivity can be abstracted to a central management system, which is called controller. The controller monitors the SLA on the Internet links and helps in traffic management by instructing the branch or head offices to utilize the right links for the data. Management of traffic flow is also done by controller on the basis of traffic type and nature of sensitivity and to split the traffic to efficiently utilize the dedicated circuit and the Internet link. Additional advantage of link utilization on the remote routers is to make the decision of traffic flow egressing the source router. Therefore, SDN-based solution helps to achieve the operational expense savings. This solution is referred to as SD WAN. There are many vendors’ offerings such solution. Some examples of commercial offerings of SD WAN are Viptela’s vSmart, Cisco’s IWAN, or Riverbed’s Steel-Connect. As shown in the figure below, a central SDN WAN controller manages the WAN routers to monitor performance for uplinks and alter their traffic flows using centrally managed policies based on link performance, nature of traffic, time of day, etc. we will now discuss SDWAN in detail.

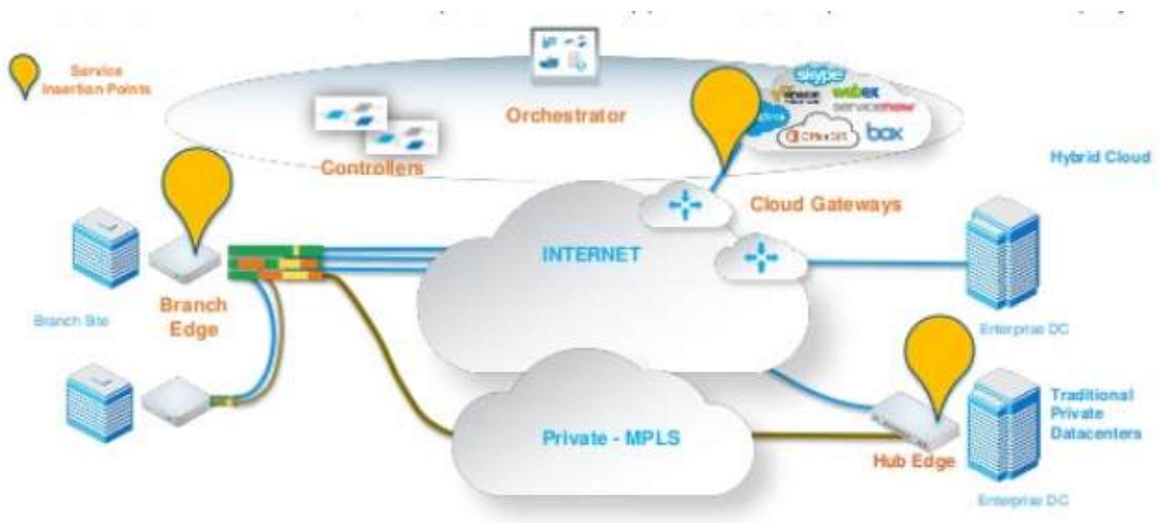


Figure 28: Cloud Delivered SD WAN Architecture. [36]

Chapter 8

Software Defined Wide Area Networks

Today, almost all IT and business leaders today are concentrating on creating a more agile business. The ability to adapt immediately to business climate changes is now the basis to determine which companies will thrive and go ahead of their competitors and which ones will be outdated and out of competition. However, one need an agile IT infrastructure to enable business agility and businesses are ready to spend billions of dollars on technology to make IT more agile. [40]

In the data centre technology, virtualization has become the basis for it and has raised the level of agility at the compute layer. Flash storage are high in use to enable migration of data at unprecedented speeds for businesses. Deployment of network virtualization has been on rise to increase the agility of the data centre in some organizations. It is estimated by ZK Research that businesses have spent $12 billion on infrastructure to make the data centre more agile. Enterprise wide-area network (WAN) is the part of IT that was yet to evolve and lacked flexibility. So, Evolving the WAN was the top priority of every IT and business leader because organizations cannot be agile until their least agile IT component- that is WAN is virtualized.

There are several other factors that are driving the evolution of the WAN, such as:

The cloud is becoming the norm as more applications and major workloads in majorly all enterprises are moving to the cloud. The cloud is one of the fastest-growing technology of enterprise software. According to ZK Research, cloud computing services will see a growth of more than 150% over a period of 5 years and will grow from about $46 billion in 2014 to more than $116 billion in 2019. There will be exponential rise in cloud traffic and hence, the wide-area networks will face a significant variation in the traffic patterns.

In todays's era, companies have to be agile in terms of making quick decisions and also involving the right people, regardless of their location. With this agile motive, it has now become a business criticality to include unified communications (UC). There has been a rise in virtual teams and mobile workers and thus, unified communications has become a mission-critical application for businesses of all sizes. According to the ZK Research 2014 Unified

Communications Purchase Intention Study, there has been 87% organizations to partially deploy UC in their organizations.
With all the advancements in IT, computing has become network centric. Priority is being given to cloud, mobile computing, the Internet of Things (IoT) and big data and these new compute paradigms are all network centric. These initiative are based on wide area network technology and thus their success is largely dependent on the quality of the network, particularly the WAN. This is where SDWAN plays an important role. [38]

Software-Defined Wide Area Network (SDWAN), is at the leading edge of software-based networking (SDN over WAN) deployments. SD-WAN plays an important role in terms of business value for organizations which have distributed branches over a wide area. It helps in terms of business agility, cost savings and the ability to leverage Internet bandwidth economics and simplifying it. SD-WAN simplify delivery of WAN services to branch offices by using software and cloud-based technologies. Software based virtualization helps in simplification of network operations by enabling network abstraction. SD-WAN helps IT and business managers in quality, reliable and secure deployment of Internet-based connectivity easily, quickly. SD-WAN has its own benefits for this deployment like ubiquity, high bandwidth and low cost.

Enterprise networks are the last set of technology that undergone the rapid transformation steered in by computer virtualization and cloud computing. Virtualization and cloud technologies plays a major role to bring improvements in industries like IT flexibility, efficiency and cost benefits. Also, it didn’t changed the basic underlying networks. After the entry of mobile devices and new applications in enterprise workloads, networks was struggling to meet their demands.
Network bottlenecks arise from the traditional architecture and it didn’t support much flexibility and scalability because it was based on hardware-centric and old technologies which are outdated now. Software-Defined Networking (SDN) came into scene after promising the solution to many of the above stated problems. It solved many problems of traditional networks with a solution using software and virtualization and hardware platforms are commercial off-the-shelf (COTS). It helped to tackle issues like scalability and flexibility. SDN led to a transition from the proprietary hardware based traditional networks to software-defined networks to keep up with the innovations in enterprise IT industry with the help of sophisticated software platform. These solutions are programmable, decoupled and agile.

Software-Defined WAN (SD-WAN) is the extension of SDN that is transforming the wide area networks and enterprise network connecting branch offices. SD-WAN enables to bring the advantages of SDN over to WAN and thus, the advantages of SDN are not limited to the data centre. SD-WAN abstracts network hardware and separates control plane and multiple data planes to automate it and use it with cloud-based management to simplify the delivery of services to the branch office. There is a level of manageability, performance and reliability assurances while deploying that enterprises expect.

SD-WAN is gaining popularity over the time in the IT world. With any new disruptive technology, existing incumbents and many adjacent solution providers try to involve that technology into their services and gain a piece of the market in the flow. This activity also create a hype and thus, is part of the IT hype cycle. However, only those vendors emerge as industry leaders and go on to define the technology space who provide solutions with real and measurable benefits.

During the past years, there have been a significant advancements in the data centre through software-defined networking (SDN) which have increased the level of agility and flexibility. However, there are many benefits of SDN that can improve the WAN, and which is also required to support the more agile data centers. Today, WAN must evolve into a software-defined WAN (SD-WAN) for businesses to reach the level of agility required to compete in today’s digital world. [37]

8.1 SDWAN Explained

SD-WAN stands for software-defined networking in a wide area network (WAN). The SD-WAN technology helps to simplify the management and operation of a WAN by decoupling the networking hardware from its control mechanism. This concept is similar to the implementation of virtualization technology by software-defined networking to improve data centre management and operation by separating data plane and control plane. A key application of SD-WAN is to enable companies to build higher-performance WANs using commercially available internet access and it is cost efficient. It helps in partial or whole replacement of an expensive private WAN connection technologies such as MPLS with an efficient and flexible counterpart.

Software-Defined WAN provides the advantages for wide area network solutions for enterprise branch offices that are typically associated with Software-Defined Networking (SDN) in data centres. Both SDN and SD-WAN virtualize resources by automating network deployment and management to provide accelerated services delivery, better performance and improved availability, also reducing the total cost of ownership. SDN in general is finding its role in any networking environment. This technology has been massively getting adopted by Web scale Internet companies, primarily in massive data centres and in the links between them, secondarily by telecom service providers in various scenarios.

The basic principle underlying SDN is that it abstracts the network to a set of roles and functionalities that are independent of how those functionalities are provided. As a result, applications that use the network need not include excessive details of the network equipment, details that change over time. SD-WAN change a hardware WAN circuits to an efficient software controlled network system, it creates a network overlay and decouple network software services by providing a software abstraction.

With the help of SDWAN and the new abstraction, it has become easier for IT managers to control and manage their network that has been possible with managing underlying hardware for WAN networks. This network overlay eases the overall process of network administration, also it provides a common interface across different physical components to help network owners to develop their own infrastructure independent applications.

Like SDN, SD-WAN also decouple the functionality into a control plane layer and a data plane layer. In networking, control plane includes device system configuration and management, it is responsible for traffic signaling and routing decisions for data packets. Whereas, data plane helps in carrying application and user data.

The important concept to SDWAN is that multiple instances of the data plane (typically switches and routers) are served by one logical instance of the control plane. Whereas, traditional networks makes programming of the network impossible as each instance of the data plane contains its own control plane.

There are several benefits of this separation of layers:

- Network service agility is increased as more of the intelligence is moved into the more abstract and programmable control plane from the data plane.

- Control plane perform management of larger and more diverse set of data plane components or physical resources and devices.
- The control plane need to communicate with the various data plane components and to enable this communication there is a need of a communication protocol, such as the standard OpenFlow protocol. This protocol is also called the Southbound Interface (SBI) as it is south of the control plane in an architectural diagram.
- An Application-programming interface (API) enables applications to program the network as an abstraction. It is called the Northbound Interface (NBI) because it is north of the control plane in an architectural diagram.
- Various options for both the NBI and SBI ends in various choices for operators, but eventually a few open standards will be finalized by the industry for these interfaces to facilitate multivendor interoperability.

8.2 The Challenge with Legacy WANs that need to overcome by SDWANs

Legacy WANs have several disadvantages that create problems in their use in retail environments. Firstly, legacy WANs lack the agility and flexibility that are required by digital organizations, especially in retail. For a retailer, these are the key requirements and the network should not fail as the failure of a network can cause immediate loss of business and revenue, even the brand can be abandoned. The list of issues a network failure can cause can be too long. To get an understanding, consider few scenarios- If the point-of-sale system or sales application is down, revenue comes to a halt which is a major problem for a business. Potential sale is lost in case the data a customer needs in order to make a choice is unavailable because of the network failure. An employee will have to search the warehouse for a product in case the inventory system is down. The problem is, most shoppers don’t wait for long as they have short patience. They’ll just go out and shop from the mall nearby and get what they want. In fact, according to a recent study in Britain, 41% of shoppers, after seeing a long checkout line have changed their mind about making a purchase and 86% shoppers avoid a store that they think has long lines and 74% of them saying they would rather shop at a competitor. So, considering this study one should improve the legacy WAN.

Which is possible, but it may takes time. Usually, it’s a box-by-box update with a lot of manual work that results in long lead times. Traditional networks were designed with an “active–passive” architecture, which means that only one link can be active at a time. This increases their functional costs as companies end up paying for bandwidth they don’t use. And in traditional approach, the average time to make a network-wide change is four months which is a long time and customers would be certainly looking for better options and will abandon a brand rather than wait in long lines. There are other factors that are making legacy WANs inadequate for use. New applications are consuming much of the available bandwidth on network, leaving little for other business-critical applications. For example- In-store WiFi is a great value for customers and good for the foot-fall, but if it consumes bandwidth that the store needs to process transactions, it can be self-defeating. So, there is a need for priority in network policy. Many retailers use a hybrid in-store/internet structure where major operations happen online like In-store purchases and in-store ordering of out-of-stock items. This requires high bandwidth at each location to download images without referring to the primary data centre. Using a legacy WAN often reduces the efficiency of retailer with backhauling internet over the WAN, which means traffic traverses the WAN twice. This affects the efficiency of network. So, the retailer need some network engineers to manage the network and to avoid slowdowns and route traffic more efficiently. This is not an easy task as those network engineers lack the visibility into network conditions to effectively plan capacity. Legacy WANs provide no ability to prioritize unified communications or video over other applications so, prioritizing applications is also not an option. An attempt at capacity planning could do more damage because of little visibility or control over individual applications. Legacy WANs has major concerns like capacity planning and network traffic. In addition to these, security is also a major concern with legacy WANs. So many organizations use an inefficient “hub-and-spoke” model which makes securing internet links at each location difficult. [42]

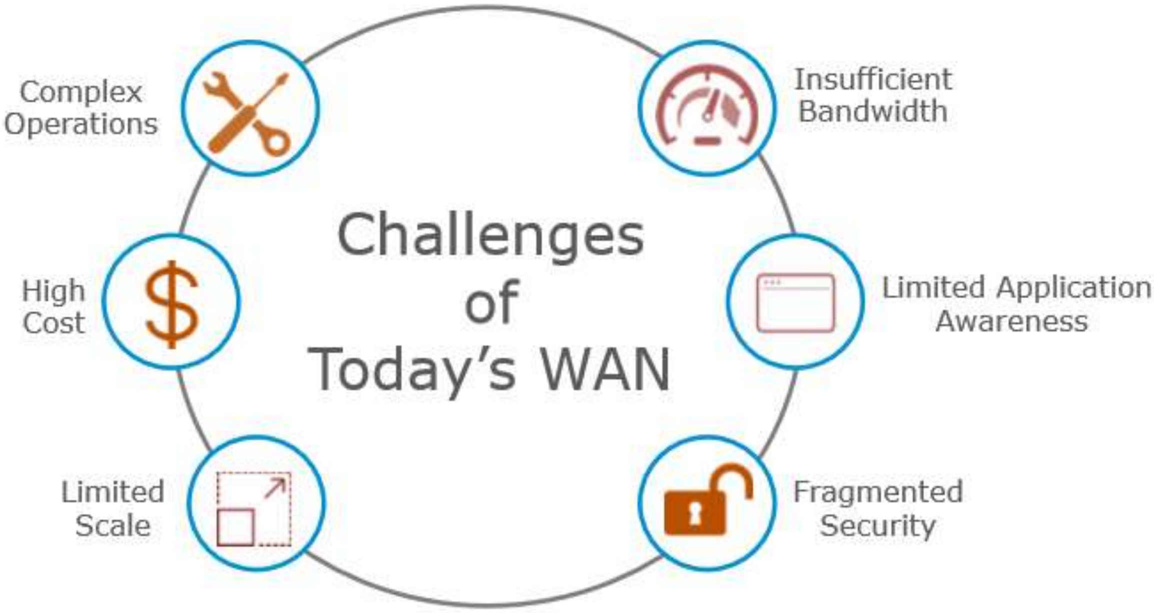


Figure 29: Major Challenges with WAN [39]

Today's WAN challenges are varied and complex. According to a new survey by IDG Research and Nuage, Networks WANs must maintain a growing number of network appliances, and they continue to increase in complexity, say more than half (51 percent) of the respondents. 51 percent of the survey respondents said that contributing to the challenges were time-consuming manual processes as it involves moves, add, changes and the commissioning of new locations. Many other challenges are faced by WANs such as decreased network responsiveness to the dynamic business environment, the difficulty and expense of keeping up with regulatory and compliance auditing and the need for a better security framework around the WAN. In addition to these challenges, organizations have their own goals for their WANs. The most important goal which is noted as critical or very important by 71 percent of respondents is boosting network responsiveness. Other key goals include optimizing the network and enabling it to support newer technologies such as cloud computing and big data/analytics and expanding the network geographically. Increasing the number and/or size of the locations that depend on the network, further concentrated on the network's flexibility and ability to adapt quickly. According to the survey, many companies lack confidence that their current WAN architecture, rollout processes and management tools can support their future goals. Only 11 percent of the respondents were satisfied with their current WAN architecture and were extremely confident that their current WAN architecture would support future goals. [41]

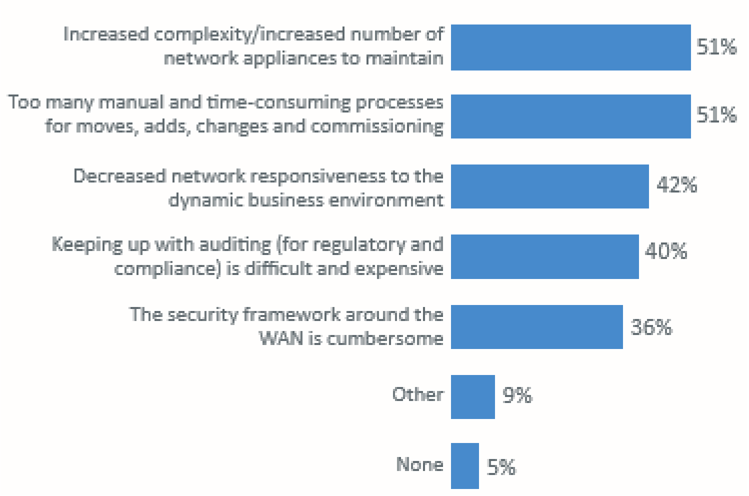


Figure 30: Challenges of WAN According to respondents of IDG Research and Nuage Networks. [41]

The existing architecture used to build WANs has been in place for a long time. It used the traditional "hub and spoke" design for implementation of the efficient delivery of client/server computing and best effort Internet traffic. In the era of traditional networking, the majority of network traffic consist of the data that moved from the data center to the branches. However, today, the fastest-growing application types are cloud, mobile computing and multimedia traffic and they drive significantly different traffic patterns compared to the legacy compute models. The evolving business climate is putting new demands on the WAN that are difficult to meet because of the following challenges:

Inefficient network design: There has been an efforts for moving away from a hub-and-spoke design to a partially or fully meshed network since long time. However, the level of complexity is very high to run a network that is even partially meshed. It is highly complex to practically install partially meshed network for most organizations and migrate away from the hub-and spoke design. Also, the redundancy of WAN links is based on an active–passive model. If the primary link fails, then the backup connection becomes active. This means businesses are paying for up to twice the amount of bandwidth that they are actually using. Which makes this an expensive model.

Resiliency/Business Continuity: WAN connectivity failover options for MPLS (4G/3G networks) are not reliable and there is lack of resiliency in organizations that their business requires. This can cause downtime and incur lost productivity with disruption in regular business operations. With all the above issues associated with traditional WANs, there is a need for enterprises to find a replacement infrastructure with significant simplification, an improved cost advantage, and better support for cloud adoption.

Poor use of network bandwidth: With the hub and spoke architecture, Internet traffic is passed down a WAN link and the hub I to pass the traffic before accessing cloud data centres and software-as-a-service (SaaS) applications. This effect usually saturate WAN links and degrade the network bandwidth. The performance of both the backhauled applications and other applications are degraded on the WAN link. The rise of mobile and cloud computing has led to more and more of an organization's network traffic to travel through Internet, which means the performance problems created by the trombone of traffic will be even worse in the future.

High cost of bandwidth: With legacy networks, using expensive private network services such as MPLS or leased lines is the only way to ensure available bandwidth for applications. There is no cost effective alternative as lower-cost Internet connections do not offer any kind of service-level agreements (SLAs) or the assurances necessary for bandwidth to be used as business-class circuits. Traditional WAN solutions have expensive bandwidth. They rely on costly MPLS circuits to maintain inter-site connectivity and quality of service. MPLS also requires long deployment times, which affects company's growth and overall productivity. These installations are also more complex. Including security and network functions means that there are more pieces to keep track of, which makes it even complex. Management through a command-line interface (CLI) compounds this problem. CLI requires a lot of time dedicated to log management tracking, increases the chance of human error, and decreases overall staff productivity.

Difficulty in optimizing the user experience: The cost of private networking services is very high but it is still a major difficulty for most network managers to optimize the quality of the user experience for applications. Consequently, network managers are constantly changing QoS settings, creating alternative paths or changing other network parameters. The constant jerking in the network is often done and users are complaining about problems and, consequently, IT is working under pressure. One of the largest cause of network outages today is human error.

Cloud Visibility and Adoption: Traditional WANs have inferior cloud visibility because they cannot offer application visibility but, only offer a packet level and routing level view. Traditional WANs can also create performance bottlenecks as the features and cloud based demands are increasing, which can impact user performance and productivity across the business. In Cloud based networks, traffic gets routed through the data centre, this can further increase latency because traffic has to travel further. Also, traditional WAN traffic cannot be assigned by policy to the right broadband channel as it is not intelligent.

Security and performance are major challenges: Securing a legacy network is typically accomplished by adding the new functionality required by layering on additional physical or virtual appliances. The overlay approach used in traditional WANs, which is built on multiple appliances, can further increase the complexity of the WAN.

Security at the Data Centre: There is a centralized security in private MPLS connections that most traditional WANs utilize. Every Traffic is entered into the data centre (a process known as backhauling) in a typical “hub-and-spoke” network architecture where everything passing through the network can be checked and filtered. Traffic is secure but it affects the performance as everything is funneled into data centre. According to the design, traditional WANs have no direct internet access via public link, which affects the performance and limit it for the ever-expanding use of cloud services such as Software-as-a-Service (SaaS) applications. Some organizations choose to purchase their network and security solutions separately, this limit the transparent visibility because of devices with separate management consoles. This makes operations more complex and time-consuming.

Long lead times for new network services: According to ZK Research 2014 Network Purchase Intention Study, the average time taken to start implementing new network services is four months. The long lead time is due to the fact that changing network settings requires a highly skilled engineer with full awareness of the network, so very few engineers in organizations can implement these changes. Also, modifications to large networks can often take months to complete because most changes to configuration are done on a box-by-box basis in traditional approach, even port-by-port basis which makes it a slow process for a large network.

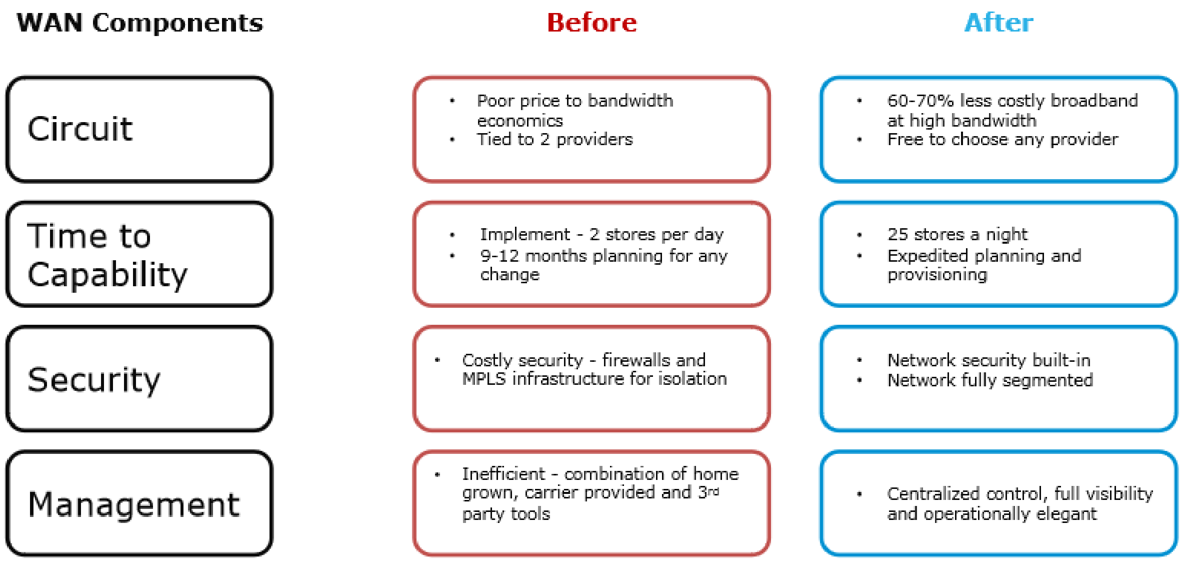
Network agility is business critical requirement today, it is no longer something companies can simply aspire to have in the future. Organizations must come out of their legacy thinking regarding network design and try to gain the same level of agility in network deployment that is present at the computation level and application level. To accomplish this, there is a need for a new agile WAN architecture. It’s time for the SD-WAN. [38, 42]

8.3 SD-WANs: A Better Way

An SD-WAN is a WAN that uses software to define business and IT policies. Traditional network operations have the disadvantage of no visibility into the business, but an SD-WAN is introduced to overcome that because it is tightly coupled to the business through policies. The policies are used to ensure the network is continually meeting the needs of the organization by automating major processes of changing configurations, moving traffic flows or performing other changes. In a traditional network, any kind of configuration or change must be done on a box-by-box basis because the transport functions and controller layer are resident in each device. In a software-defined network, the control layer is abstracted above the infrastructure and transport layer. In SDWAN, the control has been decoupled from the physical layer and runs in software, services can be virtualized and delivered from the cloud to any point on the network real fast. Service orchestration layer resides above this control or service delivery layer. In service orchestration layer, network changes are determined by business applications and policies. Application Programming Interface (APIs) are used to communicate these changes to the service layer, enabling administrative tasks to be fully automated.

The SD-WAN helps the network to become a strategic and agile business resource that can be flexible to adapt the changes as determined by business policy. The term “software defined” can be confusing to business and technology executives as it is used to mean different things in the various parts of IT. [41]

So let us understand, what is SD-WAN? To understand it simply, SDWAN simplifies branch office networking and ensures there is no downtime and applications perform at optimal levels. A software-defined WAN helps the network to be more agile and reduces cost. SD-WAN share the same principle as software-defined networking (SDN) to abstract the major characteristics from the applications that use the network like network hardware and transport characteristics. SD-WAN uses SDN principles to bring a level of agility to the WAN never seen before. SD-WAN enables multiple links, devices and services to coexist with existing solutions and interoperate with it. This setup is simpler and can make a retailer more agile and capable of adjusting the network based on in-store needs and requirements. Software-defined WANs have APIs that integrate seamlessly with popular retail management and reporting systems. This saves a lot of time that many retailers spend as it eliminates the cobbling together of disparate systems. SD-WAN enables the centralization of configuration tasks by separating the control and data-forwarding planes. As a practical matter, this means retailers can control devices that are placed in many locations with software hosted at their place, which leads to faster installation and provisioning. Mostly it just means retailers have to just perform unboxing, powering up, adding the SD-WAN appliance to the WiFi network or plugging it into Ethernet. This simplifies the process at great extent and reduces the need for expensive truck rolls and time-consuming configuration, because once connected to the network, configuration can happen centrally via SD-WAN. In addition, it is simpler to include any changes in a device that need to be applied because the software-defined WAN has a centralized portal and includes the ability to deploy changes to the whole network simultaneously with the centralized controller. High availability is another advantage of SD-WAN because, unlike legacy WANs, software-defined WANs use an “active–active” architecture. This approach can induce failover in less than a second and it provides protection against blackouts and brownouts. Making the network run more efficiently is simpler with SD-WAN because it is easy for retailers to set quality of service (QoS) policies for applications and traffic types, thereby making it easy to optimize voice, video traffic or other priorities. In addition, because SD-WAN can easily connect directly to cloud apps, which greatly improves convenience while also increasing branch and cloud security. Also, it is easy to conform to PCI requirements on the processing, storage and transmission of credit card information because SD-WAN enables easy network segmentation.



WAN Components	Before	After
Circuit	• Poor price to bandwidth economics • Tied to 2 providers	• 60-70% less costly broadband at high bandwidth • Free to choose any provider
Time to Capability	• Implement - 2 stores per day • 9-12 months planning for any change	• 25 stores a night • Expedited planning and provisioning
Security	• Costly security - firewalls and MPLS infrastructure for isolation	• Network security built-in • Network fully segmented
Management	• Inefficient - combination of home grown, carrier provided and 3rd party tools	• Centralized control, full visibility and operationally elegant

Figure 31: Legacy WANs vs SDWAN implementations [39]

The network visibility provided by SD-WAN offers a significant advantage for engineers looking to proactively manage the network. Engineers can monitor the network conditions 24/7 easily, troubleshooting any issues across any number of stores from a central console without the need of any on-site IT staff. So, these are great capabilities for an enterprise network installation and management. Still it can be challenging to find a solution provider that delivers the precise solution retailers need. Software Defined-WANs offer a potential solution for addressing the challenges faced by traditional WANs and meeting organizations' network-related goals. All (100 percent) of the respondents whose organization had already implemented SD-WANs said in a survey that it had simplified the management and operation of the overall network. A large majority (80 percent) said that the SD-WAN enabled them to seamlessly link enterprise locations, and size or geography was not an issue. 60 percent of the respondent said the SD-WAN helped deliver centralized, policy-driven management across locations. SD-WANs are based on an overlay model that uses any IP network to provide connectivity between sites. Also, SDWAN supports multiple access technologies in order to deliver a responsive network to branch locations. SD-WAN is significantly different from traditional, static WANs, as they are slow to adapt and change and they cannot support big cloud applications whose traffic demands changes heavily over time. With SD-WANs, the intelligence behind network functions is placed in software rather than hardware platforms, unlike in traditional networks. The network functions such as routing and switching, as well as the movement of traffic are performed by software rather than hardware. Software defined Networking concepts are possible due to technologies such as virtualization and the cloud. SD-WANs help companies to save a lot of time it takes to set up new offices and their networks and connecting them to head office and other branches. The time it takes to set up a new branch with a traditional WAN can be around three months, whereas with SD-WAN, the same setup can be can complete in one to two weeks. It is flexible to give companies and independence to choose any provider for network bandwidth, or a process that removes steps and reduces complexity. Also, SD-WANs can use consistent processes and standardized policies to improve security and compliance. SD-WAN is a relatively new technology, only a small percentage of the respondents (5 percent) are extremely familiar with it. More than half (56 percent) are at least somewhat familiar with the SDN technology and virtualization. Those organizations that are most familiar with SD-WANs are deploying or evaluating the technology. Nearly two-thirds (62 percent) of the respondents said their organization was likely to evaluate the technology and may promote it as mainstream technology. The benefits of SD-WANs address many of the challenges that are faced by today's companies with

WANs, this makes SD-WANs an extremely viable solution. The next-generation SD-WAN technology can help reduce complexity and speed up the process of deploying new cloud-based applications and bringing new locations onto the network.

SD-WAN can be defined by the following characteristics and these characteristics can better explain why this technology shift is of the utmost importance to businesses of all sizes:

- **Hybrid network architecture**: As stated previously, traditional WANs were built primarily on expensive, private IP network services. They were deployed exclusively as backup connections, if Internet connections were used. An SD-WAN is composed of a hybrid of network services including MPLS, private line, broadband Internet or even 3G/4G wireless and it is not limited to them only. A hybrid configuration becomes the norm with an SD-WAN.

- **Internet connections for critical business applications**: In a traditional WAN configuration, the Internet would never be used for business critical applications such as VoIP, video, big data or others. However, in a software defined configuration, the virtual services layer can be used for different business critical applications as it can quickly switch among multiple Internet connections, ensuring the fastest and highest quality path is always being used. Matching the performance characteristics of an MPLS network with a single Internet connection is impossible. However, with an SD-WAN, the network can have equivalent or better performance than MPLS at a fraction of the price as the best path is always chosen dynamically from multiple Internet connections. ZK Research estimates that this can equate to anywhere from a 10x to 100x cost efficient depending on link length and bandwidth capacity.

- **Multiple active paths become the norm**: The previous sections explained how legacy WANs were costly as they use active–passive connections where the backup (passive) connection only becomes active when the primary one fails. This is highly inefficient, as every connection is not sized to handle all corporate traffic, whereas, it must be sized to handle all corporate traffic. This is like building a highway system where all cars must take the same route, and alternative routes can be used only when the main road is unavailable. Each road would have to be built such that it can accommodate all traffic. Although this may not seem right to a civil engineer but it has been the norm with network engineers. With SD-WAN, automated and dynamic path selection is used to optimize application performance and security and multipath networks (active–active) are the norm.

- **Dynamic meshing**: Meshing has always been challenging for network managers as it complicate the network management. A mesh network is much more efficient than a hub-and spoke design, as every location is directly connected to every other location and it allows traffic to go from location to location over a single hop. However, the higher the degree of meshing, managing the network becomes more complicated due to the exponential growth in network links. Using SD-WAN, a connection would be created between two locations only when needed and when business policy dictates as SD-WAN can dynamically mesh connections. Even turning down a connection dynamically, can be easy. So, SD-WAN can give all the benefits of a fully meshed network without the associated management overhead.

- **Optimized for cloud computing**: In Legacy WANs, all Internet traffic goes through a single choke point and then is distributed over the WAN to the remote location, This makes it ineffective for cloud. Whereas in SD-WAN, cloud and mobile applications can be optimized and the entire network can be used more efficiently as it offers secure and high performance direct Internet access.

- **Automation of configuration processes**: In SDWAN, the business policy layer can be used to fully automate configuration changes. For example, if a video session is being initiated between two points, the application can use dynamic QoS and direct the network to create a path between the two locations. After the call has ended, the dedicated path can be automatically removed by application. The automation of processes can eliminate unplanned downtime caused by human error and it ensures the best possible performance.

- **Virtual service delivery**: In a legacy network, deployment of physical appliances in each location is required to deliver services to those locations such as branch offices. This makes the process of deploying new services very slow and costly, and even the simplest changes can take months to complete. With an SD-WAN, services can be delivered virtually to any location on an on-demand basis. Thus, SD-WAN makes this process faster.

- **Cloud presence**: In this era of cloud computing, there is a need for the business WAN to extend past the traditional boundaries and out to the cloud. An SD-WAN is inclusive of the cloud.

The rise of software-defined WANs is being enabled by Moore’s Law. Expensive dedicated hardware solutions compared by software solutions running as virtual resources in the cloud deliver performance. It is a key to greater network agility, and using software based solution helps to achieve this agility with no loss of performance. The shift to software also allows application flows to define data paths instead of packet flows.

Businesses that adopt an SD-WAN will realize greater network agility as compared to traditional WAN, which will enable greater business agility, which helps to compete in the digital business era. Also, based on research by ZK Research, it estimates that network operation costs can be cut by as much as 50% by running an SD-WAN which will be beneficial for business. Also, highly paid network engineers will have more time to dedicate to strategic initiatives instead of spending the majority of their time just maintaining the network status. We will discuss about features of SD-WAN in the next section [38]

Chapter-9

Fundamental Characteristics of SD-WANs

There is no standardized definition in the industry for an SD-WAN. SD-WAN implementations have not only incorporated newer SDN, NFV, and Service Orchestration technologies but also, legacy WAN technologies and functions that are developed over the past times and still evolving such as WAN Optimization , analytics, policy management , IPsec tunneling, deep packet inspection, hybrid WAN, VPN and service assurance. The latest technologies mentioned like SDN, NFV, and Service Orchestration technologies provide the integration and service deployment automation. These technologies has made SD-WAN Managed Services so efficient, flexible and cost efficient at the same time. SD-WAN Managed Services deliver agile, assured and orchestrated application-driven connectivity services by using overlay networking technologies. Also, SD-WANs does not need to establish interconnect agreements with off-net access providers to operate over Internet access connections, they uses SD-WAN to add off-net sites and quickly enable them.

Major SD-WAN vendors are working to standardize SD-WAN terminology and its entities like service components, reference architectures, LSO APIs, and its service definition for the betterment of the technology and their businesses. After all the standardization also, the service providers will have varied SD-WAN service offerings, we will discuss the major and critical features. [43]

9.1 Virtualizing the Network

SD-WAN as a network overlay offers a transport-independent overlay that helps to carry application traffic independently of the underlying transport layer. SDWANs major feature is virtual WAN that constitute a unified pool of resources from multiple links, even from different service providers. This feature helps SD-WAN to provide high availability and performance for applications. It also simplifies the network and increases the utilization of resources.

A key benefit of the abstraction principle is that there is no static tie exists between the application and the link it must use so, network operators can add new links and applications easily. When links experience degraded performance, virtualization provides self-healing for these links. [37]

9.2 Secure, IP-based Virtual Overlay Network

SD-WAN uses IPsec tunnels over Internet or MPLS underlay networks in order to provide secure, IP-based virtual overlay networks. SD-WANs can support any topology like full/partial mesh and hub & spoke. There is no need to make modifications to any of the underlay networks Because IP-based SD-WANs are virtual overlay networks. Also, SD-WAN need to include some basic network gateway security technologies like Network Address Translation (NAT) capabilities and firewalls because IP-based SD-WAN implementations often use the public Internet as one of their WANs. SD-WAN provides a secure overlay that is independent of the underlying transport components. Before participating in the overlay, SD-WAN devices are authenticated.

SD-WAN supports secure and encrypted transmission, no matter what combination of circuits and service providers it is using. It also enable automated configuration and key management across multiple branches located at various altitude as SD-WAN consist of separated control plane. Additionally, there can be inclusion of segmentation as an overlay by network designer as that is both independent and consistent across the various underlying components. [37, 43]

9.3 Transport-independence of Underlay Network

SD-WANs can operate over any type of networks- wireline or wireless access. Each WAN may use a different underlay service/technology. For example, Dedicated Internet Access, Broadband Internet (Cable, DSL or PON), Internet over LTE, MPLS over T1s, or MPLS over fibre. There is tremendous agility and simplicity in creating and deploying virtual network connectivity because of this independence and flexibility from the underlay network. [43]

9.4 Simplifying Services Delivery

SD-WAN programmability does not restrict to just connectivity policy, it also extends to the insertion of network services anywhere, in the cloud, on the customer premise equipment

(CPE) of the branch, or in regional and enterprise data centres. The business-level abstraction of SD-WAN helps to simplify configurations to route the traffic to the service delivery node and also, to configure the policy. It also simplifies complex configurations of traffic routing and policy definitions.

9.5 Providing interoperability

SD-WAN has capability of separation and abstraction of the control plane from the data plane which provides the ability to incrementally add resources and interoperate with existing devices and circuits.

SD-WAN helps to enable multiple devices, circuits and services to exist together and operate effectively with each other. APIs enable integration into existing and different management and reporting systems deployed by enterprises.

9.6 Leveraging Cost-Effective Hardware

SD-WAN improves cost effectiveness and flexibility by empowering commercially available hardware and network appliances or servers. There is a use of standard hardware for the data plane because of the separation of the control plane from the data plane.

Virtual appliances can take advantage of existing or standard commercial off-the shelf (COTS) servers using software defined technologies and it can be remotely delivered. However, we on-site IT installations for the initial installation and configuration of these servers. It is also possible to deploy virtual appliances in hosted cloud environments.

Custom-designed network appliances based on standard CPUs, memory and other components can be cost efficient, and still provide the advantages of purpose-built hardware. Custom-designed appliances can be a significant advantage for smaller and remote branches without on-site IT resources because it will come with just the right configuration out of the box, thus enabling deployment in sites without IT support.

9.7 Supporting Automation with Business Policy Framework

SD-WAN enables the abstraction of configuration into business-level policy definitions that span multiple data plane components. It remain stable over time, even as the network changes. The control plane of SD-WAN provides the programming flexibility and centralization over a diverse and distributed data plane. Enterprises can expect application awareness from SD-WAN. It also provide smart defaults that provide further abstraction from the detailed transport level details. Policy definitions are also automated and it refers to the users and groups, the applications they should use and what level of service they should receive.

There is a self-provisioning delivery model due to the abstraction from the physical layer. There is no need for pre-configuration on a per-device basis for devices. Rather, they inherit the configurations and policies automatically based on their assigned role in the network.

9.8 Monitoring Usage and Performance

There can be consolidated monitoring and visibility for whole network with the help of SD-WAN. SD-WAN provides a whole network view and thus helps to monitor across the variety of physical transports and service providers, as well as across all remote sites. This feature enables business-level visibility, such as application usage and network resource utilization. SD-WAN also provides detailed performance monitoring across all components of the data plane. As, Performance monitoring works along with business policies. It enables intelligent path development of application traffic across different paths and resources within the virtual WAN network.

9.9 Supporting Interoperable and Open Networking

SD-WAN uses its approach of open networking, interoperability and evolving standards to further improves agility, cost-effectiveness and incremental migration. Open Networking Foundation (ONF) and Open Networking User Group (ONUG) are two major organizations at the forefront of SDN and open networking**:**

- **Open Networking Foundation (ONF**): The Open Networking Foundation helps in achieving the goal of accelerating SDN’s commercial adoption by open, vendor-neutral SDN architecture, interfaces, protocols and open-source software with.
- **Open Networking User Group (ONUG):** The Open Networking User Group (ONUG) is a community of IT business leaders who exchange ideas and best

practices for implementing open networking and SDN designs. There is an ONUG Working Group for SD-WAN. [43]

9.10 Enabling Managed Services

Most of the enterprises, even the largest, outsource the management of their branch networks and WAN to either managed IT providers or to their network service providers. Additionally, some cloud application providers, such as Unified Communications as a Service (UCaaS) providers, need to access their applications by provisioning and managing the circuits. This Business requirement should be addressed by SD-WAN. It should enable managed service providers (MSPs) to manage the WAN networks of their clients with a multi-tenant infrastructure. Along with management and orchestration functions, the data centre networking components should also be designed for multi-tenancy and scalable virtual deployment in providers' cloud data centres. [37]

9.11 Service Assurance of each SD-WAN Tunnel

One of the critical part of any managed network service is its service assurance, including SD-WAN managed services. QoS performance is part of service assurance, for example, packet loss and packet latency are measured over each SD-WAN tunnel in real-time. These measurements determine the performance and quality of WAN. It determine whether a particular WAN meets the performance requirements of an application resulting in application-based performance assurance. For example, in a conferencing application, there is requirement of packet loss not more than 2% and a packet latency less than 50ms for an acceptable quality of experience. If any WAN meets this criteria, the application can be forwarded, provided transmission over a particular WAN is not disallowed by any pre-existing policy. There is higher QoS in the SD-WAN overlay tunnel as SD-WAN service can correct for any packet loss in the underlay network only.

9.12 Application-Driven Packet Forwarding

SD-WANs perform application-level classification (up to OSI Layer 7) at the customer premises. This enables subscribers to specify the applications which are forwarded over SD-

WAN tunnels over different WANs. The WAN or SD-WAN tunnel selection is determined by an application’s QoS, security or business policy requirements.

9.13 High Availability through Multiple WANs

SD-WANs support packet forwarding over one or more WANs at each site. When a site has two or more WAN connections and each WAN uses a different WAN technology, e.g., Internet and MPLS VPN, This is referred to as hybrid WAN. SD-WAN tunnels are created over each WAN while using multiple WANs. In multiple WANs connected via SD-WAN tunnel, Each WAN underlay network can use a different wireline or wireless access provider, providing SD-WAN tunnel diversity. SD-WAN tunnels can operate over different underlay network technologies. For example, SD-WAN tunnels can be created over Internet connections from different ISPs, Internet and MPLS VPNs, or MPLS VPN and LTE (Internet) enabling service provider, network path, or physical path diversity.

9.14 Policy-based Packet Forwarding

SD-WANs uses policies to make application forwarding (or blocking) decisions for SD-WAN tunnels over each WAN. Policies can be based on each application or application grouping, Examples of applications are real-time media or conferencing application. Application’s QoS performance requirements or an organization’s security or business priority policy requirements are considered for policy enforcement. For example, a QoS policy may be set as long as Skype for Business is QoS performance requirements to forward Skype for Business packets over any WAN. Also, for users to get acceptable quality of experience (QoE), user’s packet latency and loss are met as requirement. A security policy may be set so Skype for Business packets are not sent over Internet, but are sent over the MPLS VPN. A business priority policy may be set to prioritize the payment card transactions be sent over any Skype for Business packets. This may result in an occasional degradation of a user’s QoE for a Skype for Business call occurring during the payment card transaction. However, in this case, the organization would want to prioritize their payment card transactions to have higher importance than Skype for Business calls.

9.15 Service Automation via Centralized Management, Control and Orchestration

Service automation is achieved with automatic configuration of SD-WAN customer premises equipment with the help of centralized management, control and orchestration of SD-WAN tunnels. It is also referred to as “zero-touch provisioning” (ZTP) where pre-population of all configuration information is done into the centralized management system. The SD-WAN customer premises equipment (CPE) retrieve its configuration and policies without needing to send a service provider installer to the customer premises after it is powered up and connected to the Internet. ZTP just need subscribers to simply plug in LAN, WAN and power cables in order to self-install the CPE. Web portals or APIs are used to access management, control and orchestration functions. Depending upon the role assigned to a user, the function can be performed like web portal can be used for subscriber, service provider or network administrator, on-demand service modifications and service monitoring.

Chapter-10

Analysing SD-WAN Architecture

The SD-WAN architecture has these three layers (from bottom to top) as shown in figure below:

- Secure cloud network
- Virtual services delivery
- Orchestration and analytics

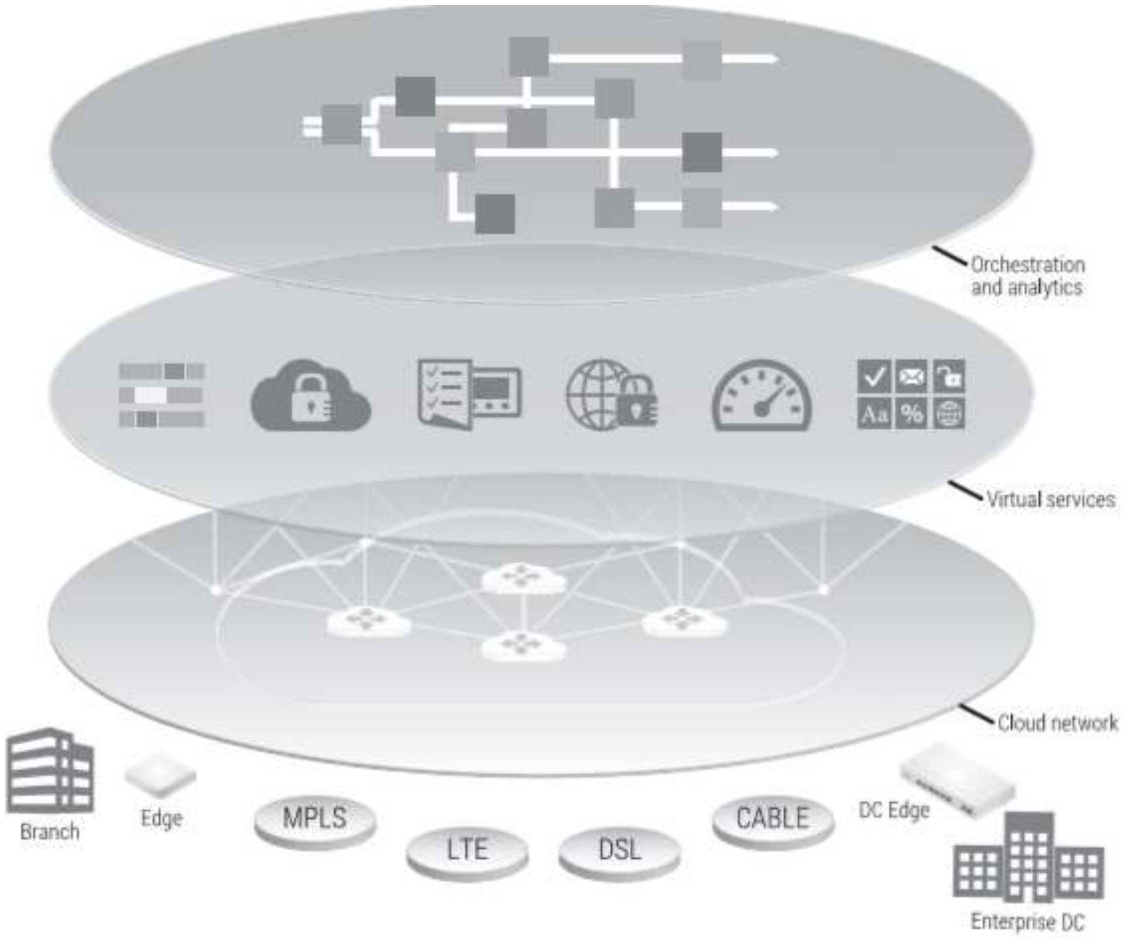


Figure 32: SD-WAN Architecture [37]

We will discuss each of the layers in more detail.

10.1 Secure Cloud Network

Secure overlay works across any combination of public or private circuits, also it is a transport-independent overlay. Connectivity to both enterprise data centers and SaaS applications should be enabled by this layer.

SD-WAN to address the issues with traditional WAN Traditional WAN delivers security and performance across private links to applications, these private links reside on a customer data centre. This arrangement has two issues:

- For enterprise-grade security and performance, traditional WAN ties a customer to a private circuit which affects its performance as the customer loses the flexibility of transport independence.
- Traditional WAN causes performance penalties for backhauled Software as a Service (SaaS) applications.

SD-WAN addresses these issues very effectively and make its solutions both secure and reliable across a combination of private-only, hybrid, and dual Internet and Internet-only sites by delivering transport independence. SD-WAN should optimize access for all types of applications that is, both on-premise and SaaS applications. At the core, SaaS applications should reduce impact of backhaul with its ability to go direct to Internet with security. It is optimal to have dual-ended service with the node hosted in the cloud for enterprise SaaS applications like mission-critical collaboration applications that require a highly resilient WAN with dynamic path forwarding, It is often close to the SaaS application that could offer per-packet application steering. Connectivity of branches to multiple cloud and on-premise gateways should be automatically multi-home and secure. Multi-homing to multiple gateways enables direct access to cloud data centres and applications, and still eliminates backhaul penalties while enabling assured performance, monitoring and additional dual-ended services.

It also helps MSPs to bring up new sites quickly by providing ordinary broadband links and therefore it also helps customers to get their branch sites up and running without waiting for private circuits. MSPs need to achieve this goal, so they need to ideally look for solutions for utilization of inexpensive public Internet links to offer reliability for voice and video transmission. There is an automatic inclusion of hybrid transport into virtual resource pool, as the private circuit comes in, thus WAN availability gets even better.

Create a scalable, secure cloud network: SD-WAN need to provide secure connectivity over any type of transport so it uses standard based encryption, such as AES to form a secure cloud network.

A new SD-WAN device can participate in the secure cloud network, but before that SDWAN management plane need to authenticate it. After it is authenticated and authorized, the SD-WAN device is given access to the secure cloud network after downloading its assigned policy. There can be sensitive traffic which needs to be isolated from the rest of the traffic. So, they can have separate encryption keys based on the policy.

Based on the traffic type, the delivery of security and optimization services can be done at a cloud node or on-premise node. Additionally, the best combination of links and gateways are picked by network layer on the basis of security criteria and performance requirements of the applications and users.

10.2 Virtual Service Delivery

The deployment of the services or a set of services from ecosystem partners, should be possible from the list of applications. SD-WAN services can be delivered at the branch, in the data centre or in the cloud, according to the service requirements. These options help to reduce the stretching of devices in the branch as only the required can be included.

Services in the branch: some services could run only in the branch like firewalls, while other services need to be positioned at the end of the destination like WAN optimization, example of their positioning can be in the data centre. SD-WAN simplifies the delivery of these services in the branch. Self-service applications can be benefit for the customers also, as it provide them a catalog to serve themselves according to their requirements. Also it is beneficial for service providers to provide the platform to deliver managed CPE offerings with dynamic services to the customers and this platform is referred to as network functions or services virtualization.

Services delivered from a regional data centre: There can be partial or complete centralization of enterprise services on the premises, such as firewalls and Web security, rather than deployed at every branch. Centralizing reduces the IT effort required to establish every branch, it also reduces number of devices used for the same with the help of these functions. However, these functions requires the forwarding of the appropriate traffic from each branch to the centralized data centre or to one of regional data centres.

SD-WAN enables the use of a simple single click to easily implement a business policy-based backhaul to a regional branch rather than setting up complex and static policy-based routing rules. However, Enterprise should also consider a reliable and secure SD-WAN enabled overlay for backhaul, instead of establishing private link only for backhaul, this save a lot of cost on expensive private links.

Services in the cloud: Traditionally, There has been a use of backhaul data majorly due to the security purpose or because of non-reliable Internet links. This resulted in inefficient use of expensive private link bandwidth and degradation in performance of SaaS application.

Providing a direct-to-Internet path to access SaaS also doesn't provide solutions for the concerns over Internet reliability and security. In order to deliver direct, secure, optimized access to cloud applications, a combination of per-packet application can be leveraged including it with cloud Web security. This approach helps to free private links for other corporate traffic by eliminating backhaul penalties to a SaaS application. It is important to easily arrange these services in the branch, regional data centre or in the cloud. And to perform these functions, one needs a powerful orchestrator with a business policy framework.

10.3 Orchestration and Analytics

SDN improve agility, leverage commodity hardware and avoid vendor lock-in by using the control plane which is separated from the data plane.

The SD-WAN architecture uses a similar principle of separation of control and data plane to provide better control and services. The orchestration layer in the SD-WAN architecture provides this feature. It provides the control plane for forwarding traffic to and from the cloud nodes and on the branch premise, also traffic forwarding across the multiple underlying transports to provide flexibility and with the policy-driven insertion of distributed network services.

The orchestration layer is highly resilient and is safe from failure, and additionally because of the separation, the data plane functions separately from the control plane. A cloud-delivered orchestration layer also provide features that make it cost efficient by simplifying end user deployments without requiring IT administrator installation.

This layer has three functions:

- **Management plane**: The management plane provides various features like zero-touch deployments, configuration monitoring, troubleshooting and reporting. It has a proper consolidated dashboard for these features. Zero-touch deployment automates the performance of WAN capacity and link characteristic measurements, including latency, jitter and loss. Zero-touch provisioning of a branch CPE is extended to zero-touch WAN configuration using Zero-touch deployment, this reduces the requirement of manual configurations of link characteristics for configurations. It enables the automatic adjustment of QoS policies with the changing link conditions. Authentication and authorization of new SD-WAN devices into the network is also the responsibility of the management. The identification of SD-WAN devices and the distribution of identity information is performed by the PKI in orchestration layer. Secure authentication of SD-WAN devices by each other and exchanging of encryption keys is also enabled by orchestration layer. It can revoke the device identity and further stop a SD-WAN device from being able to participate in the secure cloud overlay.
- **Highly available and resilient control plane**: There are many requirements of SD-WAN for its efficient functioning, one of them is a highly scalable, resilient control plane to be offered on a commodity hardware. The control plane can be present on-premise or optionally hosted by cloud. There should be a smooth migration of customers from legacy WAN to SD-WAN and this can be offered by SD-WAN control plane by interoperating with existing L2/L3 infrastructure, configuration should be maintained and there should be minimum configuration changes.
- **Business policy framework**: It represents the policies at the business level. SD-WAN should meet business policy requirements like service assurance, security and corporate governance requirements.

MSPs can manage multiple customers with a single unified dashboard with the help of SD-WAN control and the management plane using it’s features like scalability and multi-tenancy. [37]

Chapter-11

Service Components of SD-WAN

In this section, we will discuss the service components that are used in an SD-WAN managed service and their fundamental functionality. This is correspondence to MEF's LSO architecture. In this section, we will discuss each of these service components as they are listed below:

1. SD-WAN Edge
2. SD-WAN Controller
3. Service Orchestrator
4. SD-WAN Gateway
5. Subscriber Web Portal

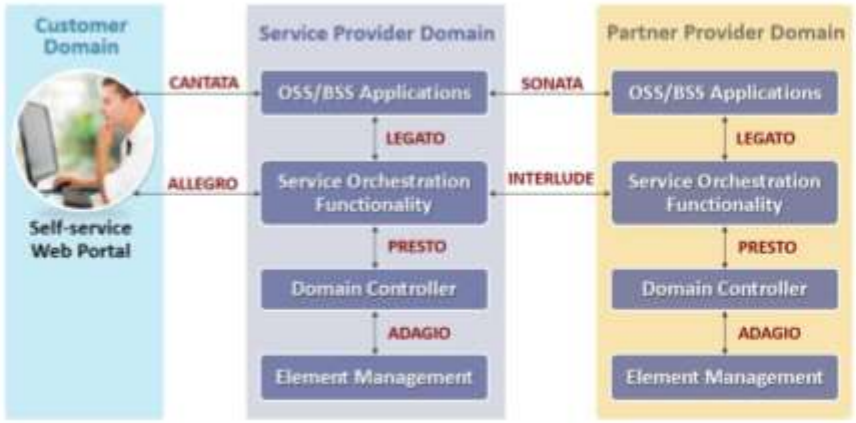


Figure 33: LSO Architecture of MEF [43]

We will overlay each of the SD-WAN Service Components and understand their placement in the MEF LSO Reference Architecture as shown in the figure, we will also understand the associated LSO RA interfaces with which these components must interact. The below figure signifies the placement of the different SD-WAN service components. We will simplify the discussion about these service component by focussing on a single provider domain and not about multiple service providers and thus, the communication of inter-provider service functions.

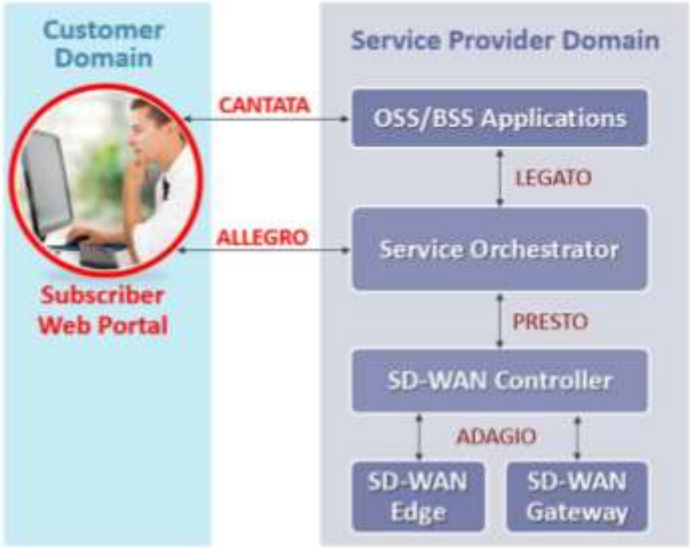


Figure 34: Placement of SD-WAN service component in the LSO reference architecture

11.1 SD-WAN Edge

The SD-WAN Edge is located at the initiation or termination point of SD-WAN tunnel. It provides the SD-WAN service demarcation which is same as service demarcation for a Carrier Ethernet service by an Ethernet NID. The main function of SD-WAN Edge is to use different types of wired or wireless underlay networks, such as T1s/E1s, broadband Internet (DSL, Cable, and PON), Wi-Fi and LTE wireless access networks, and IP (Internet) and MPLS core networks to create and terminate secured tunnels which are encrypted.

Other functions of SD-WAN Edge include application-based QoS and enforcement of security policy, using multiple WAN connections for application forwarding, and to find WAN path selection using QoS performance measurements over each WAN. It also performs functions that are used for WAN optimization such as packet buffering or reordering, de-duplication of data, data compression, and forward error correction. SD-WAN Edges must include NAT and firewall capabilities at minimum as they usually connect to public Internet WANs.

A physical CPE device placed on the customer premises can also provide the functionality of the SD-WAN Edge. CSP or MSP are used to manage these devices. Software-based virtual network function (VNF) can also be used to implement SD-WAN Edge and that VNF can run on a virtual CPE (vCPE) at the customer premises or any other type of computation platform like server in a data centre. This service can be managed by any of the CSP, MSP or by a cloud service provider. There are multiple names to a same term like vCPE, uCPE and ‘white box’ servers, these terms are used interchangeably in the industry.

The SD-WAN Edge is operated and maintained by MSP or CSP as part of an SD-WAN managed service. In most of the architectures, An Adagio interface as shown the figure (it has different names in different architectures), is used for communication between SD-WAN Edge and SD-WAN Controller.

11.2 SD-WAN Gateway

The SD-WAN Gateway helps in building connection between sites interconnected via the SD-WAN and other sites interconnected via alternative VPN technologies such as, CE or MPLS VPNs. It can be called as a special case of an SDWAN Edge. It is used in one of the two ways for delivering an SD-WAN service to sites connected via another VPN service. One way that does not require SD- WAN Gateway is to create an SD-WAN tunnels over the VPN by placing an SDWAN Edge at each subscriber site connected to that VPN service.

Another way that requires the use of an SD-WAN Gateway is cost efficient way. This approach does not require placement of an SD-WAN Edges at each VPN site to achieve interconnectivity. Rather it requires an SD-WAN Gateway which initiates and terminates the SD-WAN tunnels and perform the function like an SD-WAN Edge. It then perform the initiation and termination of VPN connections to and from sites interconnected by the VPN. This approach provides the intercommunication between the sites that interconnected via SD-WAN and other VPN technology domains. However, MPLS VPN sites will not have SD-WAN service capabilities such as forwarding traffic of applications over multiple WANs or QoS and managing security policy because SD-WAN Edges perform these functions and they do not have it.

The SD-WAN Gateways are managed and operated by the MSP or CSP as part of an SD-WAN managed service.

11.3 SD-WAN Controller

The main purpose of SD-WAN Controller is to provide management of physical or virtual devices for all SD-WAN Edges and SDWAN Gateways. It provides its services to all the SD-WAN Edges and SDWAN Gateways associated with that controller. This major functions of

SD-WAN controller are configuration and activation, management of IP address, and pushing down policies onto SD-WAN Edges and SD-WAN Gateways and implementing them. There are many more functions as it is not limited to just these. The SD-WAN controller also helps in identification of the operational state of SDWAN tunnels across different WANs by maintaining connections to all SD-WAN Edges and SD-WAN Gateways. It also retrieve information like QoS performance metrics which are critical for each SD-WAN tunnel. These performance metrics are further used by the Service Orchestrator.

The SD-WAN Controller are operated and maintained by the MSP or CSP as part of SD-WAN managed service. In most of the architectures, Presto interface as shown the figure (it has different names in different architectures), is used for communication between SD-WAN Controller via northbound to SD-WAN Edges and SD-WAN Gateways it controls and it uses Adagio interface as shown the figure (it has different names in different architectures) for communication between SD-WAN Controller via southbound Service Orchestrator to SD-WAN Edges and SD-WAN Gateways it controls. It is important to notice that some SD-WAN implementations may merge the SD-WAN Controller and Service Orchestrator.

11.4 Service Orchestrator

The service management of the SD-WAN service lifecycle is provided by the Service Orchestrator. Its functions are fulfilling services, performance, control, assurance, usage, analytics, security and policy. For example, The Configuration of the end-to-end SDWAN managed service between SD-WAN Edges and SD-WAN Gateways over various underlay WANs is responsibility of the Service Orchestrator. Examples of underlay WANs are Internet and MPLS. Other functions of SD-WAN orchestrator are using security policies, QoS policies or business or intent-based policies to set-up application-based forwarding over WANs.

Service Orchestrator is also operated and maintained by the MSP or CSP as a part of an SD-WAN managed service. In most of the architectures, the Legato interface as shown the figure, is used for communication between Service Orchestrator via northbound and Service Provider’s OSS/BSS applications for functions such as service activation and it uses the Presto interface as shown the figure, for communication between southbound and SD-WAN Controller. Other functions of service orchestrator is to use Allegro interface to obtain service modification requests from a Subscriber Portal. It is important to notice that some SD-WAN implementations may merge the SD-WAN Controller and Service Orchestrator.

11.5 Subscriber Web Portal

The MSP or CSP enable integration of the Subscriber Web Portal for the SD-WAN managed service and their present portal for customers that other managed services uses. In most of the architectures, the Cantata interface as shown the figure, is used for communication between the Subscriber Web Portal communicates and Service Provider's OSS/BSS applications for functions such as account setup of initial subscribers, ensuring availability of service payment method and ensuring that it is active, user-authorization to activate a new service, and the initial activation of SDWAN service.

After the activation of SD-WAN service, the SD-WAN service modifications is done by communication between Subscriber Web Portal and Service Orchestrator via the Allegro interface. The examples of SD-WAN service modifications are setting up separate QoS, setting up security or business policies based on the role of user, like 'view-only' capability or access for modification of SD-WAN service. The Allegro interface also helps in sending service modification requests from the Subscriber Portal to the Service Orchestrator.

Chapter-12

Deployment of SD-WAN

We need to understand the various options available for deployment of SD-WAN for connecting branches. Traditional WAN networks was totally dependent on private links based on MPLS protocol, whereas SD-WAN accesses cloud applications and data centre hosted applications using different flexible link options listed below:

1. Deploying SD-WAN on cloud
2. Deploying SD-WAN with Internet links
3. Deploying SD-WAN with hybrid WAN

12.1 Deploying SD-WAN to connect Enterprise to Cloud Services

Adoption and use of cloud services from Infrastructure-as-a-Service (IaaS) providers, such as Amazon Web Services (AWS) as well as from Software-as-a-Service (SaaS) providers, such as WebEx, Salesforce.com and Office 365 and is one of the main motivation for utilizing Internet/broadband links to connect branch sites in a WAN.

The traditional WAN architecture does not perform its functions appropriately to use cloud services for connecting enterprise branches to it. This is because of backhauling of all Internet-bound traffic to a central site using expensive private WAN links. This happens because of several reasons discussed below:

- Using software-as-a-service (SaaS) still require its traffic to go through centralized services, such as security scanning, filtering and monitoring.
- The branch needs to rely on the Internet connectivity at the central site. As there is an absence of robust connectivity to the Internet at branch.
- It affects the application performance and the end-user experience because of the traffic backhauling, also known as the hair-pinning or the trombone in other terms. This effect introduces unnecessary latency by inefficient use of private WAN bandwidth.
- SD-WAN promises to perform with the flexibility to utilize Internet/broadband links to maximum, It even try to replace expensive private WAN links. Using SD-WAN, traffic can be sent directly to cloud services over Internet/ broadband. SD-WAN

works on business policy that specifies which cloud applications should be sent directly to the Internet or whether it is required to be sent to internet or not. For performing additional network services, it redirect to other cloud services or backhaul it to a central site – for example, using broadband/Internet link to send trusted SaaS applications, such as Salesforce.com, directly instead of backhauling through a central site.

- The need for a broadband/Internet link to send Web traffic to a cloud Web security service.
- The need to scan an email traffic by a data-loss prevention (DLP) appliance by backhauling it to a central site.

Without SD-WAN, the process of using Internet/ broadband for private WAN links is complex and it requires complex setup. Even after the setup is done, it is inefficient as it will allow only rigid traffic patterns. It is really a complex task to keep record of all the IP addresses of each application and tuning the routing across each link manually on the basis of the application and the condition of the links.

SD-WAN uses business policies and automation of network implementation and thus, it simplifies the WAN. Let us discuss how it is done. For SaaS applications, an enterprise select applications and make a decision if it want to directly send the applications to the cloud, additional cloud service insertion or backhaul to a central site, business priority is to be specified as high, medium or low. Whereas, an enterprise just have to specify the business priority if they are using enterprise data centre for hosting enterprise applications. Selection of the most appropriate link for delivery of the applications is done by the SD-WAN solution on the basis of their business priority and the real-time link conditions.

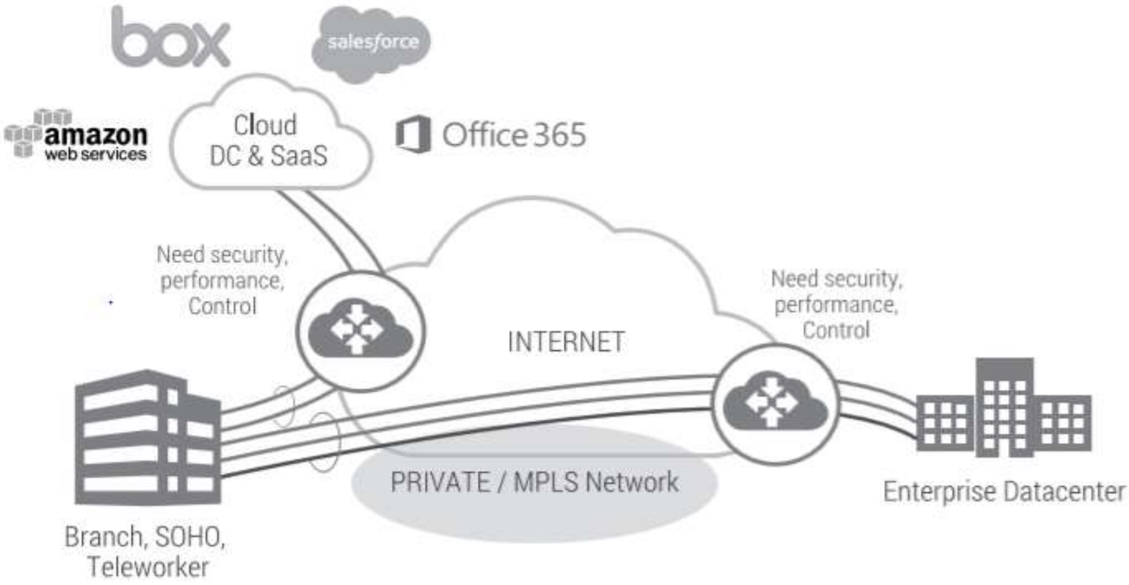


Figure 35: SD-WAN Solutions

Let us discuss the two major issues with not using SD-WAN to send traffic directly over the Internet/broadband links:

1. When the applications are traversing private corporate networks, there is no guarantee by enterprises about the availability or performance achieved of these applications. There is a major failure in delivering the performance required by applications by using the Internet/broadband links, according to VeloCloud Internet Quality Report 2H/2014, there is a 25 percent failure in delivery of good real-time application performance.
2. There is no place to deploy additional security and visibility services for enterprises, while sending the direct traffic

SD-WAN helps in using Internet/broadband to send SaaS applications and Internet Web traffic directly by the enterprises. It also helps in maintaining visibility, performing control and performance. There is a need for additional footprint in the cloud to accomplish this, and this is possible due to the software nature of SD-WAN. Also, SD-WAN helps in insertion of network services without even considering where the traffic is sent. [37]

Checking its Deployment Options

The use of Internet/ broadband by enterprises have already increased but still they use it for not-so critical purposes, such as a backup link. However, there has been a significant increase in cloud adoption and video applications, which have resulted in a drastic increase in the demand of WAN bandwidth. Now, additional branch architectures can fully utilize Internet/broadband links for their critical tasks as part of enterprise WAN because of the ability of SD-WAN to fully enforce and promote Internet/broadband links, while still competing with the private links in terms of maintaining the reliability and delivering performance. [37]

Branch Type	Traditional WAN	SD-WAN
SOHO, Small office Small office	Single Internet branch Single private WAN	Dual-Internet WAN branch
Medium office Large office	Private WAN with backup link Multiple private WAN	Hybrid WAN branch using one or more private WAN and Internet

Table: Deployment options for SDWAN in enterprises by branch type [37]

12.2 Deploying Internet WAN Branch using SD-WAN

One or more Internet links can be terminated by this type of branch, the terminated links can be any combination of broadband, wireless (3G, 4G LTE) and fiber. Using SD-WAN, the enterprise data centre performs a reliable, secure connection and the access to public cloud services is better and differentiated. Same Internet links is used by critical business applications and low priority traffic for traversing, however they perform at different service levels.

If there are two Internet/broadband links then SD-WAN has the ability to dynamically select path and steer application on the basis of per-packet that too in the middle of the active sessions. This improve the reliability and performance of the applications to a great extent.

In addition to the dynamic steering of application packets, to overcome short-lived performance issues seen in Internet/broadband links, there are many other task that SD-WAN can perform to mitigate the underlying performance issue such as on-demand remediation like Forward Error Correction (FEC). According to VeloCloud Internet Quality Report 2H/2014, SD-WAN results in having an Internet WAN branch that have more than 99 per cent of the capability to support enterprise real-time applications.

12.3 Deployment of Hybrid WAN Branch using SD-WAN

A combination of private WAN and Internet links to utilize the benefits of both is referred to as Hybrid WAN. While dual private WAN links are utilized by enterprises, it can be cost prohibitive to increase private WAN bandwidth or slow due to the circuit availability. To overcome these challenges to application performance management, SD-WAN is used across heterogeneous networks.

SD-WAN does not require an operator to manually tune the routing protocol for every application over each link rather it provides full utilization of all the available links automatically using business policy abstraction. For example, high priority real-time applications can traverse the more reliable private WAN links also while doing the same, it can use the Internet/broadband links for bursts. The utilization of the aggregate bandwidth across all links can be done by file transfer applications. SD-WAN provides a very easy option to control the link selection on a per-application basis. For example, the enterprises require an application to utilize a specific link for compliance or security reason, then this feature of SD-WAN is handy.

Chapter-13

Migrating from Traditional WAN to SD-WAN

Migrating to SD-WAN should be smooth and easy for enterprises without the complete ripping out or replacement of their traditional WAN devices. It is important to note that an SD-WAN solution is deployed on an incremental basis and it is flexible to interoperate with existing enterprise devices. Here are some scenarios to understand the easy migration from traditional WAN to SD-WAN:

1. **Scenario 1: Connecting the new SD-WAN branches to the data centre:** There should be no way that adding an SD-WAN sites implies the requirement of a new datacentre device or replacement of the traditional WAN headend. The SD-WAN solution should support standard IPSec rather than to replace a SD-WAN headend, as enterprises already use IPSec widely. This also does not implies the establishment of IPSec connectivity in every SD-WAN device to enterprise VPN headend, as this will affect the simplicity of SD-WAN. This is what SD-WAN do for its solution as it terminates the connectivity from SD-WAN devices by providing a footprint in the cloud. Which result in the requirement of only one IPSec VPN from the SD-WAN cloud into enterprise VPN headend.
2. **Scenario 2: SD-WAN device and branch firewall:** An SD-WAN device will sit in front of the firewall's public interface if an enterprise already has an existing firewall at that branch location. It will terminate the WAN connectivity and the branch firewall will be provided with the complete bandwidth. Enterprises will now have can more bandwidth, ease of management and the reliable connectivity with the help of SD-WAN along with the same security policies implementations that they already have on their firewall. SD-WAN should have basic firewall capability to accept only applicable traffic because these devices faces the Internet directly.
3. **Scenario 3: Existing MPLS WAN router with SD-WAN offload:** Enterprises do not want to make significant configuration changes to the existing infrastructure while offloading certain traffic to the SD-WAN device, examples of existing infrastructure are the WAN router and layer 3 LAN switch. The part of the traffic that SD-WAN wants to be sent through SD-WAN overlay should traverse through SD-WAN Devices. An SD-WAN device advertises the subnets to or from which it wants to process the traffic while

it is using routing protocol such as OSPF. An SD-WAN policy decides whether the traffic should be forwarded through the SD-WAN overlay or handled by the traditional WAN device after the specified traffic arrives on an SD-WAN device.

4. **Example 4: communication between SD-WAN branches to be Secure and optimized:** Enterprises should ensure that the communication between SD-WAN branches is secured and optimized. The application performance in traditional hub-and-spoke architectures can be non-optimal as they are static. For example, the branch in Florida needs to hairpin via California if they want to perform a video call between employees in two branch offices in Florida with their data centre in California. To resolve this problem in terms of traditional WAN, it need to sets up another static tunnel between branch offices in Florida which will be problematic as the number of branch offices increases because this will result in scalability and management challenges not only for their customers, but also there would be performance and reliability issues with statically defined WAN if Internet VPN is used to connect these remote branches. SD-WAN can solve these issue easily as it can provide scalability, manageability, reliability and security for branch-to-branch communications.

Chapter-14

Future of SD-WAN

There has been a significant technological shift from the traditional hardware-centric static WAN to a software-defined WAN. The shift is still in its early days so there is still a lot to be achieved by this technology. As, the trends in technology has been towards Cloud-Based Computing, Internet of Things, SaaS, IaaS, we will notice a far-reaching impact of SD-WAN on the connected world. We will conclude this report with this by exploring the extent of that impact of SD-WAN and its future prospects. We will first understand the scope of SD-WAN at present and then we will understand the relationship of SD-WAN to mobility, the Internet of Things and NFV

14.1 The Current Scope of SD-WAN

SD-WAN is used by enterprises currently for deployment to their remote sites and branch offices. The physical structures of these sites are quite different from each other's, ranging from office buildings to construction trailers, pop-up retail locations inside malls and home-office locations for teleworkers.

The users of networks in these sites are of three types:

1. The employees of the enterprises who work at these sites.
2. Customers who visit these sites. especially in retail locations, like shops, pharmacies or branches of financial organizations
3. These sites are also inhabited by partner users– for example, a supplier trying to get access to their applications that are back in the data centre or in the cloud at a construction site.

The number of SD-WAN end points under the scope of users in physical branch sites would range to even the millions.

14.2 Extending SD-WAN for Mobility

There has been a significant growth in the number of smartphones, tablets and laptops that are being used by each individual user, who consider it to be their own branch office or remote site. These devices are also called micro branches as their physical location is not stationary. The locations of these devices are incredibly diverse spread all over the places, from coffee shops to train stations.

In case of the mobile, the device of end users is also the location of their SD-WAN edge. So, it need to deploy the SD-WAN edge software on that mobile device, tablet or laptop. The addressing of the WAN links changes because of the dynamic change in physical location of that device. For example, if a user is sitting at the train station or mall or at a coffee shop then the device will use the Wi-Fi of that place or on his LTE connection. The network address of each of these links can be dynamic but still the session need to be persistent to the data centre and the cloud applications that the user wants to access.

One of the benefits that SD-WAN provides for this end user is that it allow user on the basis of the business policy, the ability to use both the Wi-Fi and LTE connections. The business policy can vary from enterprise to enterprise as it may want the user to only use the LTE connection for sending highly secure traffic, but may allow the user’s voice calls to use the Wi-Fi if the link quality is good.

The number of SD-WAN end points under this scope of mobile users will be in billions.

14.3 Growth of SD-WAN with the Internet of Things

The thing around us are getting interconnecting to the Internet and there has been a tremendous excitement about this technology in the enterprises around the world. The endpoint are now more than just human users and are these devices in the things around us from the exercise sensor to a refrigerator.

This emerging technology also has a three-tier architecture which comprised of sensors, gateways and the cloud, this architecture responds and fits very well with the SD-WAN framework. If we consider the architecture of IoT with the SD-WAN, the end point in correspondence to SD-WAN context will be each sensor in IoT and the IoT Gateway will be co-located with the SD-WAN edge.

There are already many IoT cloud services that exists in both the consumer IoT as well as industrial IoT. The SD-WAN framework can be used in the implementation of these IoT devices to make the task of setting up an IoT network very simple and easy by inserting and chaining the cloud services with the SD-WAN.

There is an exponential growth in IoT devices which may result in a huge number of SD-WAN end points, they can be in the billions to tens of billions. There has been a significant broad area covered by IoT with broad number of IoT developers, it is not just a mass market for IoT services but a broad one also. There is an open hardware movement that move in parallel to the growth of cloud software to make this formerly complex technology and processes available to a wide range of people. For example, the simplifying computing and control by the Arduino microcontroller and the Raspberry Pi Linux computer have become popular and is used by the masses. This is the goal of SD-WAN to bring the simplicity into the network.

It has major scalability challenges because of the extension of the scale of the WAN to billions of endpoints. This is already in the analytics collection. There are various apps that provides an incredibly rich information like killer app for IoT, it uses big data that will tell us how each system in this world operates. There can be thousands of useful variables that can be contributed by just a human body– from the obvious ones like heartbeats, hand pressure points or foot pressure points of a runner. Each of these sensors will produce a time-series and that series of sensors will need to be aggregated to the orchestrator using SD-WAN and then an automated functions can act upon them with alerts for the human operator. There may be requirement of a small throughput, but to manage latency and jitter across a variety of wireless and wired networks and the real-time nature of the information will require the SD-WAN overlay.

14.4 Growing SD-WAN with NFV

Network Function Virtualization is relatively an old and matured technology as compared to the SD-WAN, NFV is pushed from the Communications Service Providers (CSPs). SD-WAN and NFV are closely intersecting in many use cases. Let us understand few broad use cases of NFV:

- The Packet Core Virtualization (vEPC)

- Radio Access Network Virtualization (vRAN)
- Mobile Core Network and IMS Virtualization (vMCN)
- CDNs Virtualization (vDCN)
- Virtual Network Functions as a Service (VNFaaS)

Out of all these use cases, VNFaaS is closely intersecting with SD-WAN. The ETSI (www.etsi.org) framework in their group specification consider the vCPE or vE-CPE as one of the Virtual Network Functions.

According to the ETSI GS NFV 001 document, ‘multiple services are being deployed at the edge-of-branch offices by today’s enterprises. The cost of a dedicated standalone appliance per-feature is considered by many enterprises to be prohibitive, inflexible, slow to install and difficult to maintain’.

Essentially, this use case is about the replacement of the hardware with a set of virtual functions and virtualization of the Customer Premises Equipment (CPE). To implement the vCPE use case for NFV with an SD-WAN architecture, there are few key attributes to be considered:

- Multi-tenancy: There should be multi-tenancy between the control and data planes and they should allow sharing.
- Virtual functions should have flexible location: VNFs should be flexible for choosing any location, it should be possibly located at the CPE or in the network (cloud), or at both locations.
- Network or cloud-delivered services: Services should not just be delivered from premises based functions but also from the cloud. [37]

REFERENCES

[1] McKeown, Nick, et al. "OpenFlow: enabling innovation in campus networks." *ACM SIGCOMM Computer Communication Review* 38.2 (2008): 69-74.

[2] Campbell, Andrew T., et al. "Open signaling for ATM, internet and mobile networks (OPENSIG'98)." *ACM SIGCOMM Computer Communication Review* 29.1 (1999): 97-108.

[3] Tennenhouse, David L., et al. "A survey of active network research." *IEEE Communications Magazine,*35.1(1997): 80-86.

[4] Van der Merwe, Jacobus E., et al. "The tempest-a practical framework for network programmability." *IEEE Network* 12.3 (1998): 20-28.

[5] "Devolved Control of ATM Networks," Available from http://www.cl.cam.ac.uk/research/srg/netos/old-projects/dcan/.

[6] https://www.cs.princeton.edu/courses/archive/fall13/cos597E/papers/sdnhistory.pdf 20-40.

[7] Shalaby, Nadia, et al. "Snow on Silk: A NodeOS in the Linux kernel." *Active Networks*. Springer BerlinHeidelberg,2002.

[8] https://en.wikipedia.org/wiki/Active_networking#/media/File:Active-Network-Information-Theory.svg

[9] Software Defined Networking Concepts Xenofon Foukas Mahesh K. Marina Kimon Kontovasilis The University of Edinburgh The University of Edinburgh NC...

[10] Yang, Lily, et al. *Forwarding and control element separation (ForCES) framework*. RFC 3746, April, 2004.

[11] Greenberg, Albert, et al. "A clean slate 4D approach to network control and management." *ACM SIGCOMM Computer Communication Review* 35.5 (2005): 41-54.

[12]https://www.google.ca/url?sa=i&rct=j&q=&esrc=s&source=images&cd=&cad=rja&uact=8&ved=0ahUKEwjRs4KJ7_jXAhUT5WMKHSP4CdYQjRwIBw&url=http%3A%2F%2Fslideplayer.com%2Fslide%2F6847519%2F&psig=AOvVaw1J2Zc8LQEqXrU9-IfAbJjR&ust=1512769196185989

[13] https://www.slideshare.net/HuiCheng2/open-stack-withopenflowsdntorii

[14]https://www.opennetworking.org/wp-content/uploads/2013/02/TR_SDN_ARCH_1.0_06062014.pdf

[15]https://www.slideserve.com/josiah-burks/a-survey-of-sdn-past-present-and-future-of-programmable-networks

[16] Sivaraman, Anirudh, et al. "No silver bullet: extending SDN to the data plane." *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*. 19 (2013): 1-7.

[17] Curtis, Andrew R., et al. "Devoflow: scaling flow management for high-performance networks." *ACM SIGCOMM Computer Communication Review*. 41.4 (2011): 254-265.

[18] Yu, Minlan, et al. "Scalable flow-based networking with DIFANE." *ACM SIGCOMM Computer Communication Review* 40.4 (2010): 351-362.

[19] SDN and NFV Simplified- A visual guide to Understanding Software Defined Networks and Network Function Virtualization by JIM Doherty

[20] Koponen, Teemu, et al., "Onix: A Distributed Control Platform for Large-scale Production Networks," *9th USENIX Symposium on Operating Systems Design and Implementation, OSDI.* 10 (2010): 1-6.

[21] A. Tootoonchian and Y. Ganjali, "Hyperflow: a distributed control plane for openflow," in *Proceedings of the 2010 internet network management conference on Research on enterprise networking. USENIX Association,* (2010): 3-8.

[22] S. H. Yeganeh and Y. Ganjali, "Kandoo: a framework for efficient and scalable offloading of control applications," in *Proceedings of the first workshop on Hot topics in software defined networks. ACM*, (2012): 19-24.

[23] Shalimov, Alexander, et al. "Advanced study of SDN/OpenFlow controllers." *Proceedings of the 9th Central & Eastern European Software Engineering Conference in Russia*. ACM, (2013).

[24] S. Schmid and J. Suomela, "Exploiting locality in distributed sdn control," in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking. ACM,* (2013): 121-126.

[25] Canini, Marco, et al., "Software transactional networking: Concurrent and consistent policy composition," in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking. ACM,* (2013): 1-6.

[26] http://homepages.inf.ed.ac.uk/mmarina/papers/sdn-chapter.pdf

[27] SDN: Software Defined Networking by Ken Gray, Thomas D. Nadeau.

[28] Cisco Connect- Belgium and Luxemberg (18 April 2013) – Bjorn R. Martinussen (DC solution Architect).

[29] Foundations of modern Networking: SDN, NFV, QoE, IoT and Cloud by Stallings, Williams Chapter-3.

[30] **ODCA14:** Open Data Center Alliance. Open Data Center Alliance Master Usage Model: Software-Defined Networking Rev. 2.0. White Paper. 2014.

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

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

[33] Network Function Virtualization (NFV) with a touch of SDN by Chayapathi, Rajendra

[34] http://agemasystems.com/blog_con.php?id=86

[35] https://www.slideshare.net/ADVAOpticalNetworking/transport-sdn-use-cases-and

[36] https://www.slideshare.net/velocloud-official/maximizing-sdwan-architecture-with-service-chaining-velocloud

[37] Software-Defined WAN for Dummies by Sanjay Uppal, VeloCloud, Steve Woo and Dan Pitt

[38] A Software-Defined WAN Is a Business Imperative by Zeus Kerravala

[39] SDWAN 101 by Rob McBride in Viptela

[40] https://en.wikipedia.org/wiki/SD-WAN#cite_note-networkworld1-1

[41] CIO Quick Pulse * Software-Defined WANs: Answering the Demands of the Cloud-Based IT Ecosystem

[42] SD-WAN IS A CRTICAL COMPONENT in the future of retail by Zeus Karravala-ZK Research.

[43] Understanding SD-WAN Managed Services by MEF July 2017.