SOFTWARE DEFINED NETWORKING

Report

Ksenia Belikova (43091201)
Rojesh Upretty (43320219)
Gayatri Hiregoudar (43350467)
Romaan Khadeer Ahmed (42794550)

10/11/2013

CONTENTS

Executive Summary

As information technologies constantly progress and get more sophisticated old standards of organizing computer networks cannot remain efficient in the intensively changing environment. Today's networks have to process a noticeably more complex content and, moreover, provide capabilities to support different devices, platforms and technologies. Complexity of the network has been there for a long time, some of the complexity has been broken into simpler form through the layer architecture but still control plane remains highly complex. SDN as a new technology comes with the proper abstraction of detailed configuration and forwarding model of network along with simpler networking principal that can be flexible, cost-effective and support high power technology.

According to [1], some of the key computing trends driving the need for a new network paradigm include:

- ***Traffic patterns' changes:*** At the present time the scale of traffic traversing data networks within the enterprise data center has changed so significantly that back some 20 years ago no one would believe it. Applications no more communicate with each other following the standard client-server scheme where the core of the interaction lies between the two, the client and the server. Today's trend is network software accesses different servers and retrieves data from multiple databases, suchwise generating an intermediate traffic with a snowball effect before the actual data delivery to the end user device. User behavior encourages changing in network traffic patterns as well. As statistics reveal, corporate content and applications are more and more often accessed by non-corporate devices, e.g. smartphones, tablets and etc. from anywhere at any time. Finally, many enterprise data centers managers are embracing a utility computing model with a private cloud or public cloud included, or even a mix of both which creates additional traffic across the network [1].

- ***IT consumerisation:*** As mentioned above in today's environment corporate networks host a steadily increasing number of personal devices. Thus, the field becomes more user oriented since it has to provide accurate data to different type of end devices, yet the data must remain protected as intellectual property [1].

- ***The rise of cloud services:*** The reason cloud services, both private and public, have become so popular among enterprises if large scale business units now have a strong need for accessing multiple servers with various operating systems and configurations, with tens and hundreds of applications installed and of course the business expects these IT resources to process data in a efficient, secure and as high-speed way as possible. Growing security and auditing requirements along with business reorganizations, consolidations, and mergers that can change assumptions overnight even add complexity to it. Hence, enterprise networks have to meet needs for flexible scaling and quick yet relatively simple adjusting functionality [1].

- ***Bulky bandwidth:*** As the scale of business increases from a small company to a corporation, so does the data the business wants to be processed. Data manipulation processes become sophisticated, leading to huge inputs and outputs when completion of a task in a single thread on a sole machine would take long hours, days or even months. To avoid those delays, enterprises

use parallel processing on hundreds of servers directly connected to each other and sharing their task subset's results. Obviously, network capacity in data centers must respond to datasets growing that sometimes can cause solving a problem of scaling the network to previously unimaginable size while maintaining any-to-any connectivity without incurring financial losses [1].

Thus, the rise of mobile devices, expansion of cloud services, and server virtualization are the factors that have made the standard networking tree-like paradigm to yield.

This work is dedicated to observation of the SDN technology, an alternative to the conventional networking architecture, its purposes and challenges, implementations of the SDN concept within different vendors as well as real deployment practices.

Technical overview of SDN

Principle, goals and benefits

As the need for a new networking paradigm became obvious, the Open Networking Foundation, a non-profit industry consortium that is leading the advancement of SDN and standardizing critical elements of the SDN architecture, presented Software-Defined Networking (SDN) which was to totally transform networking architecture.

SDN is a new approach to networking that, in opposes to the conventional networking paradigm, decouples the control of the network from the hardware level and gives it to an application, called a controller. The goal is to avoid the physical limitations of a network [1].

Logical view of the SDN architecture can be seen on Figure 1. The network intelligence is (logically) centralized in software-based SDN controllers, which has a global view of the network [1]. Due to the concept, the applications and policy engines recognize the network as the only logical switch. Thus, SDN equips enterprises with a powerful, vendor-independent technique of control over the entire network from a single logical point [1].

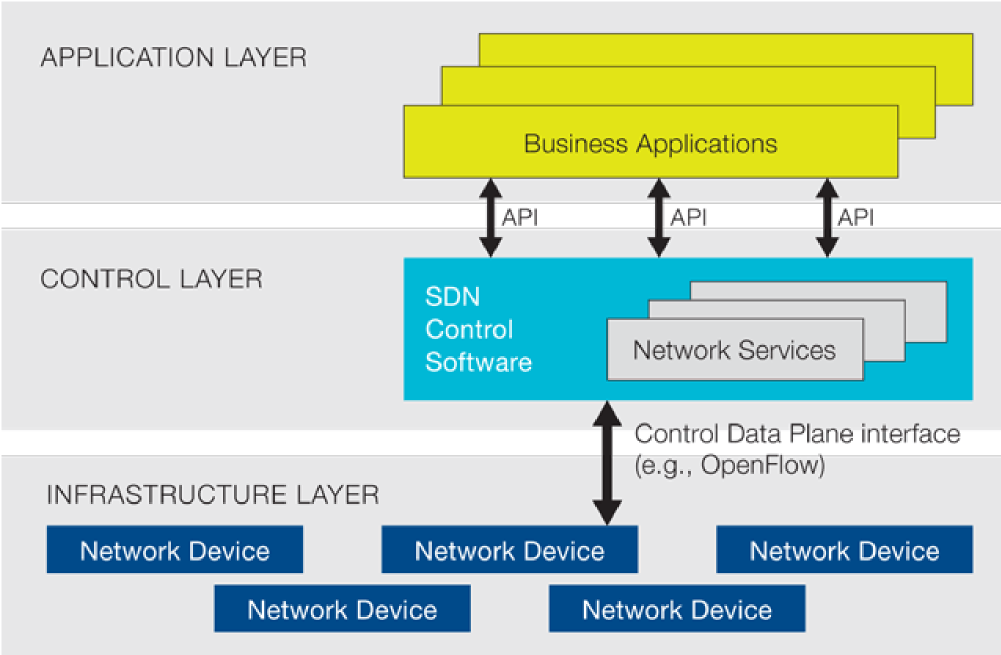

Figure 1. Software-Defined Network Architecture [Reference- Online [1]]

SDN noticeably simplifies the network design and networking operations [1]. Network administrators, “armed” with this new era networking technique, are capable to filter traffic from a centralized control console without adjusting individual switches [1]. The functionality, obviously, becomes more valuable with an increasing number of computer nodes, end-systems and, consequently, flows of data that need to be managed by a centralized system.

The value of SDN in the enterprise lies specifically in the ability to provide network virtualization and automation of configuration across the entire network/fabric so new services and end systems can be deployed rapidly and operational cost can be minimized [2]. As a result, enterprises get centralized network control along with high scalability and flexibility of networks based on SDN, while configuring such a network becomes less time, money and human recourses consuming.

With SDN network devices themselves can stop “learning new languages”, since they no longer need to understand and process thousands of protocol [1]. With a single software manageable control point on the controller side, devices’ only duty is to follow the instructions coming from the controller.

To summarize, SDN is a young yet very perspective technology which provides business with great benefits that appear especially noticeable with an increasing scale of the company, where SDN practices are deployed. Among benefits are single point of managing the whole network, hardware independence, meaning vendor-independence, rapid and seamless deployment of new devices and services and reduced operational costs.

OpenFlow

Saying about SDN in general, OpenFlow cannot be skipped as it is the first standard interface designed specifically for SDN. High-performing, with granular traffic control across multiple vendors' network devices the protocol controls and structures communication between the control and data planes of supported network devices [1]. At the moment OpenFlow is being implemented by such big names vendors as Big Switch Networks, NEC, HP and others. It is officially implemented in many of Ethernet switches and routers, currently available on the market, which demonstrates an increasing interest to SDN technology.

Based on a report [1], substantial benefits of OpenFlow to both enterprises and carriers, include:

- Centralized management and control of networking devices from multiple vendors [1].
- Improved automation and management by using common APIs to abstract the underlying networking details from the orchestration and provisioning systems and applications [1].
- Rapid innovation through the ability to deliver new network capabilities and services without the need to configure individual devices or wait for vendor releases [1].
- Programmability by operators, enterprises, independent software vendors, and users (not just equipment manufacturers) using common programming environments, which gives all parties new opportunities to drive revenue and differentiation [1].
- Increased network reliability and security as a result of centralized and automated management of network devices, uniform policy enforcement, and fewer configuration errors [1].
- More granular network control with the ability to apply comprehensive and wide-ranging policies at the session, user, device, and application levels [1].
- Better end-user experience as applications exploit centralized network state information to seamlessly adapt network behavior to user needs [1].

Summarizing the above, SDN is a dynamic and flexible network architecture with a big future. It reduces limitations of the conventional tree-like network architecture, provides centralized control, which significantly simplifies managing of huge networks and makes deployment of new physical or virtual servers into the existing network rapid, smooth and cost effective. It protects existing investments while future-proofing the network [1]. With SDN, we can watch a new era in networking has come.

Key research challenges in SDN

Network management is complex and challenging. Due to the rigidity in underlying network, there is minimal chance of improvements. Network infrastructure is rigid because network devices are integrated and difficult to modify [3]. SDN is a new networking paradigm which simplifies the network configuration and management by separating the data plane and the control plane. This decoupling allows both the planes to evolve independently with standard set of API's between the two, giving rise to various advantages in flexibility, programmability, and centralization. It hence accelerates the innovations in control plane [4].

Scalability of Software-Defined Networking

Centralization of the network by SDN brings about numerous advantages, but it also leads to concerns about scalability and resiliency [4]. Even with enormous controller capabilities, the ability to withstand the growth in network goes bleak. The concerns were further intensified when earlier benchmarks on the first SDN controller NOX showed it could handle only 30,000 flow initiations per second with sub-10 ms flow install time [4]. Also, as earlier SDNs were flow based, additional flow initiation delay became a problem [4].

According to a report decoupling in case of SDN has its own snag [4].

- Firstly, defining a standard API between two planes is not easy. It should be able to heed to the demands of various architectures, and should be able to aid independent evolution of both planes. The switches have to be accustomed with the same APIs, resulting in tie ups with specific vendors; this leads to proprietary layers hampering innovations and changes in network [4].
- Secondly, Moving traditional local controller functionalities to a new remote SDN controller results in new bottlenecks. It can also result in substantial signalling overheads depending on the network and applications [4].

The issues with SDN are described below [4].

Controller Scalability

SDN design relocates all the control functionality to a centralized controller with complete network-wide view. However as the network grows, more and more events occur, requests starts pouring in and there will be a point when the centralized controller will not be able to handle the requests, leading to bottle necks [4]. This issue can be addressed by the following methods:

1. Level parallelism in multicore systems and improve IO performance.
   Providing adequate channel bandwidth to the controller increases its ability to support large network. Or we can also reduce the number of requests to the controller. For example In case of Devoflow [4], short lived flows are handled in the data paths itself and only larger flows are forwarded to the controller, effectively reducing the load on the SDN controller, thus improving scalability [4].

2. Distributed control plane functionality over multiple controllers.
   SDN is not restricted to just having 1 controller, the computation of the control functionalities can be distributed over multiple controllers. However, while considering availability and partition tolerance achieving consistency is not feasible [4]. Furthermore Distribution of the controller can be made elastic to grow and shrink as per the traffic conditions and the load dynamically distributed among the controllers to improve scalability and performance [5]

Flow initiated overhead

Having controller install forwarding state on switches per flow bases increases flexibility but may cause flow set up delays and limit scalability. A good design technique can efficiently reduce the bottlenecks and delays. Proactive design technique in which the forwarding entities like switches are organized and configured even before the actual flow is initiated is one such design technique [4].

Resiliency to failures

Resiliency to failure and time taken to recover from failure is an issue with SDN and network performance in general. In case of multiple controller networks, with appropriate controller detection mechanism, a switch can detect if there are any controllers in their partition and notify it, in case of a failure. This is one way of tacking the issue of resilience to failures [4].

Inspite of SDN having few scalability, resiliency and overload issue, the advantages provided by SDN is undeniably valuable. By using the right technology choice, the issues can be mitigated. We now take a look at few current researches in the field of SDN.

Current research work in SDN

Controller-centric hybrid networking model

SDN in switching/routing networking realm is all about providing a clear separation functions in data plane and control plane. This abstraction is implemented using numerous architectures such as SS7, IMS/NGN, GMPLS, PCE and so on [6]. OpenFlow is the main protocol or API used between controller or forwarding elements. However, there has been very little interest in implications of SDN crossing paths with traditional routing and forwarding IP. OpenFlow provides the opportunity for the users and operators to modify the prevailing routing services without expensive vendor dependence.

A lot of research is going on to understand if traditional environment of closed vendor’s routers and switches can co-exist with new OpenFlow–enabled devices. This is called a Hybrid model.

Christian E. Rothenberg et al [6], for example discuss Routing control platforms (RCP) in context of SDN and OpenFlow, describing case of how it can be potentially used. They propose a Controller-centric hybrid networking model and exhibit the design of the Route flow controller platform (RFCP) along the prototype implementation of AS-wide abstract BGP routing services [6].

SDN in wireless networking

The increase in popularity of SDN has interested the wireless networking community. They are now trying to understand if there are any advantages of using SDN in wireless networking scheme and if so, how the SDN concept should be enhanced to suit the essence of wireless and mobile communication [6]. To understand if SDN can be applied in case of IEEE 802.15.4-based *low rate wireless personal area networks* (LR-WPAN) [7]. 3 things need to be considered:

- Analyse the advantages of SDN in LR-WPANs.
- Identify the system requirements for LR-WPANs as compared to wired networks.
- Apply Software defined wireless network, a solution proposed by Salvatore Costanzo et al [7], to support SDN approach in LR_WPANs.

Advantages of using SDN in wireless networks:

SDN is about simplification and resolvability rather than improving performance of the network. SDN allows network control and management solutions to be deployed on existing underlying equipment easily. As per the requirements and context of operation, these solutions can be replaced or modified without changing the equipment. This concept can also be applied to WPANs. While there is a general agreement on technical solutions to be used in the first two layers of protocol stack, the upper layers and in the management plan there are various options available. Depending on the network scenario different alternatives can be used. Ex: in wireless sensor network, there is a universal agreement on the access technologies to be used (Should be based on IEEE 802.15.4) but there are options available for characterisation in higher layers. ZigBee and 6LOWPAN are popular alternatives but are incompatible with each other [7].

Due to the differences in upper layers of protocol stack, the nodes applying the same access technology in the lower layers may not enter a new network thus limiting migration to new networks. This is where SDN can play a role. In SDN network management operations are decoupled from the forwarding plane. Hence the functionality applied on higher layers is defined through software and can be easily modified [7].

Network Control framework

The main issues with network management is: enabling frequent changes to network states, network configuration should be allowed in high level language, Better visibility and control over tasks should be provided for network diagnosis and troubleshooting. There is an evident gap between high level event driven- network policies and low level network configuration.

Hyojoon Kim and Nick Feamster [3], have suggested a network control framework called “Procera”, which allows operators to use high level functionality to express event-driven network policies that react to various types of events [3]. Procera [3] bridges the gap between high level policies and low level networking effectively.

Procera provides a set of control domains to the operators to express event driven network policies. Depending on the condition, the operators can use these control domains to set certain conditions and specify appropriate packet forwarding action. Increase in control domains may aid in implementing flexible and reactive network policies. Operators can also combine control domains to bring out rich network policies, hence reducing the dependency on timer or event driver scripts which are error prone. Procera is appropriate to support a wide range of network policies in different types of network environment that is hard to implement in traditional configuration languages [3].

SDN Controller currently available on the market

SDN controller is the core part of the SDN architecture. All the abstractions of overall network for the simplicity of networking system relays highly on the application program that deals with the flow control, so that network is creative or intelligent. More specifically we can say that SDN controller is responsible for determine the forwarding path for each flow over the network. SDN controller resides between application and device so that any communication between them goes through it, Such that whenever packet is received from forwarding element it is controller that computes forwarding path sending suitable entries back to forwarding element [8].

SDN Controller allows user to control network flow irrespective to the underlying complexity of physical devices. SDN Controller is based on open flow protocol. OpenFlow is the approach used to communicate between controller and switches and network devices and controller.

SDN being a new concept in networking it has to come up with potentially strong feature to replace current networking system. SDN Controller being the centralized application in SDN it has a great challenge to satisfy different kind of evaluate criteria, some of which can be network virtualization, scalability, network functionality, performance, security and reliability. Since OpenFlow enable SDN it should support OpenFlow as key aspects. With all these features and many more to support along with competitive market there are many vendor that has come up or going to bring SDN controller.

Big Vendors commercially competing for the SDN controller

Cisco

Open Network Environment (ONE) Controller: ONE is the Controller that support OpenFlow protocol as well as own onePK (one platform kit) APIs. The ONE Controller is a software platform to support SDN architecture which is comprises of foundations like platform APIs (onePK) as well as providing easy connection between physical and virtual infrastructure [4]. The Controller has three applications:

- Network Slicing
- Network Tapping
- Custom Forwarding

Big Switch

Big Network Controller is SDN Controller released by Big Switch Network that support OpenFlow protocol and provide programmatic interface for application. Big Network Controller is very much compatible with Floodlight Controller and is also known as commercial OpenFlow Controller based on project floodlight with additional interface for application. **Floodlight** is open source whereas Big Network Controller is commercial Controller [10] [12].

It supports up 1,000 network devices, a maximum of 250,000 new host connections per second and can deliver 600,000 OpenFlow updates per second [10]. Some of the key features of Big Network Controller are:

- It creates common abstraction and universal data model for underlying network data plane elements.
- It provides a flexible and scalable platform for deploying network applications.
- Capable of re-establishing connection by providing new flows as backup controller shows that it support resiliency [12].

There are two application Big Virtual Switch which simplify physical network into logical construct and Big Tap application which creates flow maps that emulate network taps for traffic interception. Both of these application are SDN application that came out with Big Network Controller and take advantage of it [10].

Juniper Network

Juniper Networks Contrail is SDN Controller from Juniper Network is Linux-Based software that run in X86 server. The software is compatible with CloudStack and OpenStack cloud orchestration platforms. Juniper Networks Contrail is commercial whereas juniper has open source SDN platform known as **OpenContrai**l. Both products are founded on the same code base and include the same functions. All commercial customers will receive Juniper service and support with their purchase [13].

NEC

ProgrammableFlow PF6800 Controller is first available SDN Controller developed by NEC which support OpenFlow 1.3 and provide more choice in designing network by avoiding vendor lock-in making it advantage of the Controller. For providing more control of network resource and for the support of multitenant network virtualization technology NEC come up with another Controller called ProgrammableFlow 2.0 which is built on OpenFlow 1.0 , which can provide real time view of network monitoring, policy managing and service resiliency [14] ,[15].

IBM

IBM not too far from the new technology in networking called SDN has come up with its own Controller called **Programmable Network Controller** is said to have similar features like NEC's ProgrammableFlow Controller which will again based on OpenFlow 1.0 [16].

IBM controller is very much compatible with network devices from other vendor and it also support multi-tenancy. IBM controller being a very expensive commercial controller focuses in larger installation.

Plexxi

Another SDN Controller based on optical interconnect rather than OpenFlow. **Plexxi controlle**r is a tried architecture where its switch performs distributed processing for real-time forwarding by avoiding the real-time computation in the data path [17].

HP

HP SDN Controller based on x86 which is said to offer complete SDN portfolio and runs on an x86-based server. HP controller provides the simple abstraction of network view by eliminating the CLI entries. It features with exposed API providing third-party software developers to write their own applications on it. HP controller is based on OpenFlow protocol.

SDN Non-Commercial products

There are lot more commercial product available now to boost SDN as our next future in networking but along with commercial aspects there are also many non-commercial open source project that act as a platform for SDN technology. Some of the SDN Controller/ OpenFlow software project which are Open source and are Controller platform are [18]:

Floodlight

It is a SDN Controller which is based on Java and supports OpenFlow protocol. It is from Big Switch Network which has built commercial Big Networking Controller which is based on project Floodlight.

MUL

It is another SDN Controller that supports OpenFlow which based on C. It is from Kulcloud and uses multi-threaded infrastructure as its features.

NOX:

It is first OpenFlow Controller which support both Python and C++. NOX provides fast, Asynchronous IO and NOX acts as the basis of other SDN Controller. **POX** is another NOX based controller based on Python only.

FlowER

It is underdevelopment OpenFlow Controller that creates a platform for writing network control software in Erlang in simplified way. FlowER is built under the SDN and NFV project by German based software company which already use FlowER.

Beacon

It is dynamic, fast java based controller that supports both event based and threaded operation. There are different version of Beacon available and is developed by Stanford University.

Jaxon

Jaxon is a Java-based OpenFlow Controller borrowing some features from NOX.

NodeFlow

It is an OpenFlow Controller written in pure JavaScript for Node.JS. NodeFlow is actually a very simple program and relies heavily on a protocol interpreter called OFLIB-NODE.

There is also Controller to be release in near future like **SKY** Controller from ADARA Networks, a global software company which is showing its interest in SDN technology. Not only ADARA there are few more vendor's product are to be released that support SDN networking. With all these different SDN Controller for SDN platform it is very interesting to see which vendor get lead in the race of SDN controller production which is a core application program in this new networking evolution.

Network function virtualization

How it started

The network is crowded with large number of variety of hardware appliances. Service providers, who were looking to deploy network services, felt the constraints of hardware appliances in terms of space and power to accommodate them, Cost of deploying those, skills required to design them, and manage the increasing complexity. Also Hardware based- appliances come with a shelf life, which disintegrates further with rise in technology and service innovations and there by suppressing the growth of network services. This is where the concept of Network function virtualization comes into picture. NFV proposes a new way of designing, deploying and managing networking services.

Definition

Network function virtualization intends to change the way the network architecture is designed by the network operator by evolving standard IT virtualization technology to consolidate various types of network equipment onto industry standard high volume server, switches and storage, which could be located in data centres, network nodes, and in end-user premises [19]. NFV implements the network functionality such as NAT, intrusion detection, caching, DNS, firewalling etc. in software that can be run on different industry standard server hardware, can be relocated or instantiated in any network location as required, without needing to install a new equipment.

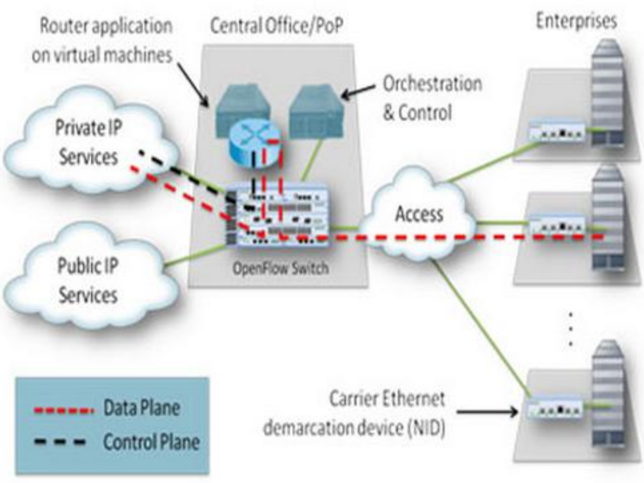

Figure 2: Network function virtualization [Reference- Online [20]]

The main objective of NFV is to simplify software-defined, virtualized service providers' network [21]. These networks have rapid introduction to new services and flexibility to adapt to changing network scenarios enabled, due to application-aware routing, along with decoupling of packet forwarding from the control. All the network functions and applications run in virtualized distributed data centres instead of a dedicated function based hardware equipment [21].

Potential Benefits of using NFV [19] [20] are:

- Reduction in capital expenditure- Using NFS greatly reduces the cost of purchasing custom made hardware equipment and promotes pay-as-you-grow models to eliminate wasteful provisioning [20].
- Increased velocity of time to market- It effectively reduces the time taken to deploy new networking services to keep up with changing business requirements. It seizes new market opportunities and improves the returns on new service investments [20].
- Efficient testing and integration – It minimises the risk accompanied with rolling out new services, by allowing service providers to easily do the testing and trial methods to suit the customers' specifications and needs all on the same infrastructure [19] [20].
- Reduce Operational expenditure – exponentially reduces the space occupied, power consumption and cooling requirements of hardware equipment and cut down the effort of rolling out or managing network services [20].
- Multi-tenancy- It provides tailored services and connectivity to multiple users, applications, or other network operators on the same hardware with clear segregation in different administration domains [19].
- Improved efficiency- Improves the efficiency in operation due to having uniform physical network platform and its tolerance and correlation with other platforms. It can also quickly grow or shrink depending on the changing demands [19] [20].

Challenges for Network Function virtualization [1]

- Portability/interoperability – Is important as it creates different ecosystem for virtual appliance vendors and datacentre vendors, while both are clearly coupled and are dependent on each other. It also provides the operator with the freedom to optimise the location and virtual appliances resources without any constraints [1].
- Performance trade-off – As NFV is based on industry standard hardware, a probable reduction in performance have to be taken into consideration. By using appropriate technologies and hypervisors the performance degradation can be mitigated.
- Migration and co-existence of legacy and compatibility with existing platforms – NFV architecture must favour migration of path from today's proprietary physical network appliance based solutions to more open standard based virtual network appliance solutions [1].
- Security and Resilience –The virtual appliance should be as secure as the physical appliance in the system, especially the hypervisor and its configuration. Network operators should have tools to control and verify the configurations [1].
- Network Stability – Ensuring stability is a challenge when managing large number of virtual appliances between various hardware vendors and hypervisors. It could lead to potential instability in network.
- Integration – Network operators must be able to integrate servers, hypervisors, and network appliance from different vendors without significant costs and avoiding lock-in [1].
- Simplicity – It should ensure that the network platforms evolved over the years will be simpler than ones that exist currently.

NFV's relationship with SDN

Network function virtualization and software defined networking share several common benefits. Both of them enjoy strong vendor support due to strong end-user following. Additionally, both are based on virtualization. NFV and SDN are not fully standardized architecture by themselves [22].

NFV and SDN are not necessarily dependent on each other; however the concepts and solutions from both can be combined to achieve greater value. NVF goals can be achieved independently without the help of SDN just by relying on currently used techniques in data centres. But SDN's approach of separation of control plane and data forwarding plane can be used to enhance its performance, facilitate better maintenance procedures [19]. On the other hand, NFV supports SDN by providing infrastructure upon which SDN software can be run [19].

NFV and SDN address very different problems. SDN aims at exploring programmability and virtualization of network devices, Whereas NFV deals with virtualization and shrinking of network appliances. SDN's objective is to automate configuration of network and NFV is concerned about automation of deployment and configuration of network functions running on virtualized server platform.

SDN for data center networking

Current data centers' potential and issues

Nowadays data centers can integrate dozens of hundreds of physical servers and up to ten times more of virtual machines. From the economical point of view a bigger number of servers reduce costs of maintenance and upgrade per server. However, large scale turns out to be a source of problems when it comes to building a network data center. The bottleneck of traditional approaches is in Ethernet network segment size limitation. As the size is usually limited to a few thousands of devices, this leads to further scaling issues as the number of the network "consumers" grows.

IP-based networks can be of any size, although it imposes certain restrictions on the movement of virtual machines and real servers within IP subnets. The conventional 3-tier data center architecture *Core*, *Aggregation*, and *Access is* a way too far from the maximum attainable values along with its fading ability to correspond to increasing complexity of tasks. Subsequently, Distributed Fabrics as well as reducing the number of levels of aggregation and cellular network structures (Mesh) have become a popular solution. Today most of networking equipment manufactures for data centers offers the appropriate solutions. However, as they are usually not compatible with each other, it puts a limit on data centers' choice of the equipment. Eventually, more and more often data centers "vote" for SDN by deploying SDN-based solutions.

As already mentioned in the work, SDN provides "more" benefits for data centers as its seamless integration functionality and centralized networking management appear more visible on large scale examples. Another obvious benefit of SDN for data centers comprises "painful", vendor-, device- and operating system-independent data access within the network. In addition, the data is retrieved and delivered to the recipient regardless number of end users.

SDN Practices

"It's an area that hasn't had any innovation in decades," says Jamie Goldstein of North Bridge Venture Partners about the networking industry. "Vendors are still selling the same Ethernet switches they were in the 1990s. Maybe the interfaces are faster, but it's all basically the same technology. Until recently "SDN is a fundamental change in networking", he says, which is causing excitement in the venture capital community [23].

As the research on the topic reveals, SDN practices have been deployed in many data centres and as there are a few big names in the list of those, this might mean SDN has made their business more efficient. Further are listed some companies that collectively invested dozens of millions in funding SDN and, more important, their products are deployed and have been solving problems of real customers quite effectively.

HotLink

With the excessive amount of software available, each one claiming to pointedly simplify modern data centers life, there is no surprise a data center infrastructure is infested with heterogeneous software. Thus, to keep up the consistence data centers have to run multiple servers with varying operating systems to be able to launch different application. The bad news is "heterogeneity is typically synonymous with complexity," as HotLink CEO LeBlanc says.

Hotlink have developed their SuperVISOR product [23], which goal is to “fight” against a heterogenous environment. The software is claimed to simplify management of various hypervisors via using a plugin to VMware vCenter [23]. Hybrid Express platform gives vCenter users control over Amazon Web Service's public cloud resources.

In effect, Hotlink's software tricks VMware vCenter into thinking that Hyper-V, KVM or Xen-virtualized machines are running on VMware hypervisors. A variety of management platforms allow users to deploy virtual machines across hypervisors, but HotLink goes beyond just that and allows users to migrate workloads across them, too [23]. "No one else is really doing integration to this degree," says Bernd Harzog, an analyst at The Virtualization Practice. And, the functionality it provides obviously highly appreciated by big companies, as the company has an impressive list of backers. Its advisory board includes CIOs from some of Silicon Valley's biggest names, including from Facebook, Citrix and EA. Partners include VMware, Red Hat, Microsoft and Citrix, makers of the four major hypervisor platforms [23].

The attractiveness of the HotLink’s application is it gives IT administrators the power to set access controls, track benchmarking and use, make changes to, configure and manage the instances across clouds [23].

Flush with venture funding and actual customer-generated revenue, and fresh off winning a best in show award at VMworld 2012, HotLink is now looking to broaden its products' integration with additional management platforms and to increase their functionality [23].

Lyatiss

Lyatiss is going in another direction by elaborating an approach to SDN. Their software is mainly focused on application-defined networking.

"Cloud networks are unpredictable and complex; traditional networks are blind to the applications running on them and rigid," says CEO Vicat-Blanc.

Lyatiss’s CloudWeaver was developed as an operating system to make networks application-aware. Running on Amazon Web Services' cloud, the product has a series of collectors that monitor application demands in real-time through an API, collects the correlating network requirements and provisions the network to those specifications [23]. If one application needs a low-latency connection for speed, the network will automatically be configured for that. If another app needs high bandwidth for a certain type of traffic, the network can be optimized for that, too. Lyatiss was born out of the publicly funded French technology research institute INRIA where Vicat-Blanc studied high-demanding applications. Researchers monitored the virtual networking developments in U.S. academia, as well as the market adoption of cloud computing, and saw an opportunity to link the two with network controllers [23].

"There are a lot of business opportunities in this market," Vicat-Blanc says. "Networking and the cloud used to be separated, but now there's a convergence and there's a lot to be done to make them work together." Users will not embrace SDN technology if it isn't practical for them to use, and making it practical is about ensuring that the network can respond to the needs of the applications running on it, she says.

Midokura

Midokura was started by the founders with a goal to create a public cloud service in Japan that would rival Amazon Web Services. Members of the team worked with Amazon CTO Werner Vogels to help create Amazon Web Services' distributed Dynamo database system. But when the Midokura team began planning for the massive data center infrastructure that would be needed, they ran into a big problem. Networking, they found, was not ready for the cloud. To get to their goal, they needed to solve the networking problem first, which is why they created MidoNet software. When testing the technology they believed it could be as big of a business opportunity as creating an AWS in Japan, and have been focused solely on their networking operating platform since [23].

MidoNet software centrally controls all aspects of an IaaS cloud network, from the switches to the routers to x86 servers to firewalls to any other piece of commodity hardware. It centrally analyzes traffic at the very edge, as soon as it enters the network. Each aspect of a Midokura network is linked, which the company says allows the software to route traffic directly from the entry point to the destination in the most efficient way possible [23]. In doing so, it's "blowing up the router," says Midokura's Cherian.

"Since you're abstracting away the devices from the network controls and pushing the intelligence of the network to the farthest edges, the physical infrastructure is still there, but you're not requiring as much from it," Cherian explains. Switches and routers, he says, simply aren't designed to handle the massive amount of changes that occur within a cloud network. MidoNet is designed for the dynamically scaling network requirements of a cloud. Initial deployments of the system work with OpenStack-powered clouds [23].

Nutanix

Efforts by big name vendors such as Cisco, VMware and EMC (via their VCE venture) to combine data center elements such as compute, storage and networking into one plug-and-play system have been mildly successful to date, according to industry analyst Arun Taneja. What Nutanix is doing is different though, what Taneja calls "hyper-convergence" [23].

Nutanix sells a hardware-software combination that combines compute and storage, meant to allow for easy scalability to a massive size with simple management. In 2011 the company won a best in show at VMworld in the virtual desktop category and investors have already poured more than $70 million into the company. Nutanix says in 30 minutes its system can be installed and IT admins have a data center in a box, ready to scale just by adding more nodes [23].

The key, says CEO Pandey, is the notion that data stored in Nutanix is written locally and stored redundantly across however many nodes are included in the deployment. That allows for easy horizontal scaling - just add more boxes to increase capacity, while not degrading performance. Nutanix is built on Intel Sandy Bridge processors and the VMware ESX hypervisor, but its Version 3.0 software released in December expands support to Red Hat's Kernel Virtual

Machine (KVM). Automatic redundancy across the system means virtual machine-level disaster recovery capabilities, while the VMware platform allows for vMotion - the transferring of virtual machine instances - across nodes within the system [23].

Nutanix is playing in a crowded field with some heavy hitters. Along with VCE, IBM has Pure Systems and NetApp has FlexPod. Taneja, the analyst, says those are "pseudo" convergence efforts - disparate pieces assembled together and optimized to work collectively. "Hyper-convergence can only happen if you start afresh," he says, which is exactly what Nutanix has done [23].

Each of the companies above has great achievements in SDN field. HotLink’s application is a powerful tool for managing instances across clouds, which implements a very tricky way of integration when virtual machines can be deployed across hypervisors. The technique is quite unique at the moment, as no other software yet allows users to migrate workloads across the network devices.

CloudWeaver by Lyatiss is a”smart” operating system, run on Amazon Web Services', that monitors network workload and demands in real-time through an API and optimizes the network by allocating resources according to the collected requirements.

Midokura’s MidoNet software implements the concept of centralized control over IaaS cloud network. The benefit functionality of the product is it centrally analyzes traffic starting from the entrance point. Thus, the problem of routing the traffic doesn’t occur when using MidoNet.

Nutanix hardware-software solution combines compute and storage as a single piece to implement the idea of scalability and simple management over massive data. The benefits are rapid and simple deployment, effortless scalability, reduced complexity of storage and reduced cost.

Conclusion

At the moment software-defined networking (SDN) stands to establish itself as a major theme throughout session talks and informal conversations. Companies, small or large want to associate with this emerging technology so that they are not left out. Breaking the complexity of overall network and making its foundation strong compare to the traditional networking ideology, SDN is sure to be a new paradigm where current networking is shifting. Also, NFV being a huge complementary to SDN, this technology is a hot evolution in networking. As the industry watches the results of early adopter case studies and hears pundits advocate for this new means of controlling the network fabric, consensus is building for the potential that SDN could be an IT game-changer.

References

[1]. *Software-Defined Networking: The New Norm For Networking* – **Open Networking Foundation** [Online]. Viewed 30th September 2013. Available: https://www.opennetworking.org/sdn-resources/sdn-library/whitepapers

[2]. Enterasys Secure Networks (2012). *Software Defined Networking (SDN) in the Enterprise* [Online]. Viewed 2nd October 2013. Available: http://www.enterasys.com/company/literature/SDN_tsbrief.pdf

[3]. Kim Hyojoon and Nick Feamster, "Improving Network Management with Software Defined Networking," in *IEEE Communications Magazine*, Feb 2013.

[4]. Soheil Hassas Yeganeh, Amin Tootoonchian, and Yashar Ganjali, "On Scalability of Software-Defined Networking," in *IEEE Communications Magazine*, Feb 2013.

[5]. Advait Dixit and Fang et al Hao, "Towards an Elastic Distributed SDN Controller," in *HotSDN'13*, Hong Kong, China, August 16, 2013.

[6]. Christian Esteve Rothenberg and Marcelo Ribeiro et al Nascimento, "Revisiting routing control platforms with the eyes and muscles of software-defined networking," in *HotSDN '12*, NY, USA, 2012, pp. 13-18.

[7]. Salvatore Costanzo and Laura st al Galluccio, "Software Defined Wireless Networks:Unbridling SDNs," in *Software Defined Networking (EWSDN), Workshop on Software Defined Networking*, Darmstadt, 2012, pp. 1-6.

[8]. S. Agarwal and et al., 2013 , "Traffic Engineering in Software Defined Networks", in INFOCOM, 2013 Proceedings IEEE, pp. 2211-2219. [Online] .Viewed on 30th September 2013.
Available:
http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=6567024&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D6567024

[9]. "HP makes SDN splash", 2012, *Network World*, vol. 29, no. 17, pp. 8. [Online]. Viewed on September 29th 2013.

[10]. G. Ferro, 2012, "Big Switch's OpenFlow Controller Guns for SDN Leadership", in **Network Computing online** [Online] Viewed on 1st Oct, 2013.
Available :
http://search.proquest.com.ezproxy.library.uq.edu.au/docview/1151686952?accountid=14723

[11]. F. Greg (2013). *Cisco's ONE Controller Debuts; Targets SDN* - **Network Computing - Online** [Online]. Viewed on October 1st 2013.
Available:
http://search.proquest.com.ezproxy.library.uq.edu.au/docview/1283980570

[12]. Big Network Controller (2013). *Big switch Networks*. [Online]. Viewed on September 29th 2013.
Available: http://www.bigswitch.com/products/SDN-Controller

[13]. J. Duffy (2013). *Juniper ships SDN controller, dismisses Open Daylight technology* **- Networks Asia** [Online]. Viewed on October 1st 2013.
Available : http://search.proquest.com.ezproxy.library.uq.edu.au/docview/1434869245

[14]. *NEC Announces First Generally Available SDN Controller with OpenFlow 1.3 Standard* (2013) **- Business Wire** [Online]. Viewed On October 3rd 2013.
Available: http://search.proquest.com.ezproxy.library.uq.edu.au/docvie

[15]. *NEC Continues Leadership in Software Defined Networking with Latest OpenFlow Compliant Product Suite Release* (2012) **- Business Wire** .[Online]. Viewed On 2013 September 30.
Available :http://search.proquest.com.ezproxy.library.uq.edu.au/docview/1000437403

[16]. M. Fratto, 2012, 'IBM's New OpenFlow Controller Priced for Deep Pockets' in Network Computing - Online .[Online]. Viewed On 2013 September 30
Available: http://search.proquest.com.ezproxy.library.uq.edu.au/docview/1082400676

[17]. A. C. Murray,2012, 'Plexxi Cuts a New Path to SDN' in **Network Computing - Online** [Online]. Viewed On 2013 October 1.
Available: http://search.proquest.com.ezproxy.library.uq.edu.au/docview/1222219475

[18]. 'Software Defined Networking Open Source Projects', 2013, SDN Central. [Online]. Viewed On 2013 October 1
Available: http://www.sdncentral.com/comprehensive-list-of-open-source-sdn-projects/

[19]. Margaret Chiosi et al, "Network Functions Virtualisation," in *SDN and OpenFlow World Congress*, Darmstadt-Germany, october 22-24, 2012.

[20]. http://www.sdncentral.com/. [Online].
Available: http://www.sdncentral.com/whats-network-functions-virtualization-nfv/

[21]. (2013) 6WIND. [Online].
Available: http://www.6wind.com/solutions/network-functions-virtualization/

[22]. Marc Cohn. (2013, Sep. 05) SDN central. [Online]. Available: http://www.sdncentral.com/education/nfv-insiders-perspective-part-1-goals-history-promise/2013/09/

[23]. Brendon Butler (27, February, 2013). 10 network virtualization, SDN and data center companies to watch – **Network World** [Online]. Viewed 3rd October, 2013. Available: http://www.networkworld.com/news/2013/022713-virtualization-sdn-data-center-267119.html

[24]. Rakesh Saha, Amit Agarwal. *SDN Approach to Large Scale Global Data Centers* – **Open Networking Foundation** [Online]. Viewed 3rd October, 2013. Available: http://www.opennetsummit.org/archives/apr12/agarwal-saha-mon-datacenters.pdf

[25]. Colin Barker (1 October 2013). *Software defined networking is about to hit the big time: Could SDN apps be the key?* **- ZDNet** [Online]. Viewed 4th October, 2013. Available: http://www.zdnet.com/software-defined-networking-is-about-to-hit-the-big-time-could-sdn-apps-be-the-key-7000021201/