INVITED PAPER *Special Section on Network and System Technologies for Sustainable Society*

# Deployment of OpenFlow/SDN Technologies to Carrier Services

**Yoichi SATO**[†a], **Ichiro FUKUDA**[††], ***Members*, *and*** **Tomonori FUJITA**[†††], ***Nonmember***

**SUMMARY** The use of computing resources on network is becoming active in the Internet and private networks. OpenFlow/Software-Defined Networking (SDN) is drawing attention as a method to control network virtualization for the cloud computing services and other carrier services. This paper introduces examples of OpenFlow/SDN technologies applied to commercial cloud services. Various activities to expand coverage over commercial carrier networks are also mentioned.

***key words:*** *OpenFlow, Software-Defined Networking (SDN), Network Functions Virtualization (NFV), Operation Administration and Maintenance (OAM), automation, datacenter, cloud*

## 1. Introduction

Traffic flow patterns in the Internet are drastically changing in the Cloud era. Before the Cloud era, the most disturbing issue to the Internet backbone was the inflation of peer-to-peer (P2P) traffic typically generated from file exchange software between personal computers. Recently, many information resources are stored in cloud computing storages behind the Internet. We can easily access the information resources by not only personal computers but mobile devices such as smart phones and tablet computers. Thus, the traffic pattern which we should care about is shifting from P2P to Cloud-to-End style. For example, heavy traffic is generated by video sharing services such as YouTube, Niconico, and so on. In addition, enterprises' business systems are run on virtual machines provided by cloud computing services as well as virtual machines on the premises. Thus, many information resources are transported from cloud computing resources to end terminals via networks. Telecommunication service providers are now facing and dealing with both an increasing of amount of traffic and fast changing of traffic pattern in the cloud era.

An existing IP network consists of many network equipment that have functions for both data-plane and control-plane, and the network equipment work based on autonomous distributed control. In order to add a new function in the network, a modification on specific software to the network equipment is necessary. But only a related development vendor can handle such a software modification. A network service provider has no option but to wait for the vendor to decide to develop the function. After the new function is developed by the vendor, it is necessary for the network service provider to setup all equipment by manual labor using Command Line Interface (CLI). During the setup process, it is also needed to consider the state transition of working routing and signaling protocols of the whole network and equipment. Thus, it is difficult for network operators to effectively manage the whole network based on autonomous distributed control. This forces them with labor burden and sometimes also causing human errors.

OpenFlow [1] is an architecture developed by carriers and vendors and it is one of the themes of Clean Slate Program of Stanford University. OpenFlow separates the data plane and the control plane inside network equipment, and OpenFlow Protocol (OFP) provides a communication method between an OpenFlow Switch (OFS) that plays for the data plane and an OpenFlow Controller (OFC) that plays for the control plane. It is expected to enable network service providers to develop by themselves the software which is on the OFC, and to effectively manage the whole network through the OFC in centralized fashion. In March 2011 the Open Networking Foundation (ONF) [2] was founded to standardize and spread SDN (Software-Defined Networking) by using OpenFlow. The word, SDN, was originally started using to express an overall network technology, typically OpenFlow. However currently it has a more wide meaning through broad understandings. Considering the future carrier network, SDN technology is thought to be an effective method for flexibility and rapid adaptation to realize service differentiation and cost reduction.

There is an active movement to realize network functions such as firewalls and WAN accelerators by software on commodity servers. These functions are provided by vendor specific hardware of network appliances up to now. The main reason of this movement is that the European Telecommunications Standards Institute (ETSI) established a sub working group of Network Functions Virtualization (NFV) [3] and is summarizing the specification requirements against NFV. However, it is also true that emerging technologies that enable servers to become a replacement of network equipment contribute to this movement. For example, the progress of semi-conductor technology greatly enhances the server processing performance. Also acceleration software of packet forwarding process on commodity server such as Intel's Data Plane Development Kit (DPDK)

Manuscript received April 10, 2013.
Manuscript revised August 30, 2013.

[†] The authors is with NTT Communications Corporation, Tokyo, 116-8019 Japan.

[††] The author is with NTT Innovation Institute, Inc., CA 94491, US.

[†††] The authors is with NTT Corporation, Musashino-shi, 180-8585 Japan.

a) E-mail: y.sato@ntt.com

DOI: 10.1587/transcom.E96.B.2946

[4] is generously provided as open source software. ONF and NFV maintain the collaborative relationship and expand open and innovative network technology.

NTT Communications (NTT Com) aggressively promotes OpenFlow/SDN technology R&D as one of the board members of ONF and also watches the trend of NFV closely. Specifically NTT Com as a carrier has already provided commercial service's for the first case in the world [5]. For example, OpenFlow technology has been introduced in the network infrastructure for our cloud computing services since June 2012. NTT Com is also developing its own controller for purpose of applying SDN to backbone network as well as datacenter network and held some demonstrations at Interop Tokyo 2012 and 2013. In addition, NTT Com studies network operation and management in SDN era to keep providing the same level of stable and current services to many customers.

In this paper, NTT Com's SDN activity and the direction of where to apply SDN technology are described. The second chapter outlines the problems carrier networks have, the third explains NTT Com's activity in detail, the fourth mentions future issues for SDN, and the fifth gives summary of this paper.

## 2. Issues for Carriers' Networks

Figure 1 shows the network model of carrier network that should be considered for SDN technology adoption.

In the data center (DC) there are various building blocks to provide cloud services such as Infrastructure-as-a-Service (IaaS), email, web and Voice-over-IP (VoIP). The DC network needs flexible and scalable intra-DC networks to interconnect numerous DC equipment in response to the expansion of cloud services. There is also a need to connect to the backbone network and to communicate externally out of the DC.

The backbone network accommodates various network services and multiplexes packet-based traffic from a huge number of customers. It is also flexibly allocates the network bandwidth to the services and customers depending on the requirements. Transport network layer underneath the backbone network consists of Packet Transport Network (PTN) and Optical Transport Network (OTN) devices.

Access network includes wireless access lines, the Internet lines, and also private wired lines.

On-premise network is not a part of carrier network, but it is desirable for carriers to perform unified service operation and management to provide enterprise customers with high-quality end-to-end services.

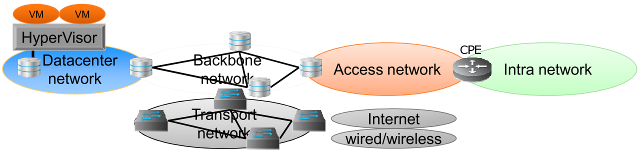

**Fig. 1** Reference network model.

The followings are issues of each area:

(1) Intra Datacenter Network

Intra cloud datacenter network accommodates many physical servers on each of which a hypervisor runs virtual machines (VMs) for cloud services. Cloud services need to distribute the load as equally as possible between servers for load balancing and also sometimes eliminate some server load for maintenance. VM migration should be carried out between different servers. IP addresses of VMs must not change after VM migration in order to keep the connectivity even when VMs are migrated to far located servers. This is the reason why Ethernet technology has been used to configure intra DC network. However there are various issues to address when the scale of the intra DC network grows.

(a) VLAN ID limitation to 4094 IDs: network isolation is performed by VLAN technology in most of the data center networks, but the number of VLAN IDs is limited to 4094 with 12-bit IDs, and this imposes restrictions on network size.
(b) Intermediate switches' MAC address table explosion: MAC addresses of VMs are kept in intermediate switches such as Top-of-Rack (TOR) switches and spine switches, but current typical switches' MAC address table size is limited to around 128K entry.
(c) Large network overhead (e.g. Address Resolution Protocol (ARP)): ARP relies on broadcast within a VLAN, and when the number of VMs grows in datacenter, overhead ratio cannot be ignored.
(d) Inefficient network usage: Layer-2 network is statically configured, and for link redundancy there are link redundancy protocols such as Multiple Spanning Tree Protocol (MSTP). However, standby links are not utilized in normal case.
(e) Possible network down due to misconfiguration
(f) Possible loop occurrence due to human error or configuration error
(g) Insufficient Operation Administration and Maintenance (OAM) functions in the case of using inexpensive Ethernet switches

Above problems are categorized mainly into scalability, hardware flow table limitation, network stability and operability. SDN is expected to resolve these problems by integrating each independent service network into one physical network within a datacenter and by reducing cost and delivery lead-time by configuring each virtualized network for each service. In Sect. 3.1 some examples are introduced.

(2) Backbone Networks

It is important to start investigating the possibility of early SDN introduction into existing IP backbone networks where we have problems to resolve. In the future, introduction of SDN into the new networks include the optical layer using optical switching devices such as Reconfigurable Add/Drop Multiplexers (ROADMs) and packet transport layer using Multi-Protocol Label Switching - Transport Profile (TMPLS-TP) will be required. For migration from existing network to SDN-based network, it is important to plan it in consideration of equipment investment cycle. The following items are issues to be considered when we design backbone network.

(a) Layer-2 and layer-3 network convergence.
(b) Network provisioning and configuration automation.
(c) Efficient network resource usage from layer-1 through layer-3 and up.
(d) Fault/Disaster tolerant network design.
(e) Efficient and easy network operation.
(f) SDN interoperability with other carriers' network.

Section 3.3 shows examples of SDN applications that can resolve the above issues.

(3) Access network

The access network is the segment between the carrier Point-of-Presence (POP) and customer premises. It is difficult to cover all of the service area by one carrier's own facility especially in a global network, so it is important to use other carriers' access services. When we design access networks, interconnection between various services such as wired/wireless/Internet services should be considered. This is based on the interconnection with other carriers' network. Currently, the connection is made by service order (SO) basis. It is expected to reduce operational cost and to enable early service delivery if automation by SDN among carriers becomes possible. In order to deal with diverse terminals and access methods (Smart phones, 3G/LTE, WiFi, etc.), it is necessary to consider OpenFlow/SDN control of these terminals. The following items show issues for designing access network.

(a) Interconnection with various access services such as wired, wireless and Internet.
(b) Handling of Quality of Service (QoS)/Quality of Experience (QoE) under limited network resources
(c) Handling of various terminals
(d) Cost reduction of Customer Premises Equipment (CPE) by its simplification

(4) Enterprise network

The enterprise network is not what carrier's handle directly. However, it should be investigated from the point of configuring/operating the combined network of enterprise network and carrier network. It is especially important for the enterprise network to manage both network and Information Technology (IT) resource access control such as the case of changing the attributes of employees in response to personnel changes.

## 3. NTT Communication's Activities on SDN

### 3.1 Application to Cloud Services

NTT Com operates an OpenFlow-based network for cloud services which started in June 2012 [5]. This service provides each customer/operator with user-friendly operation through the customer/operator portal and enables automation of necessary network configurations in response to VM creation requests. Figure 2 shows the mechanism of making self-provisioning possible. OpenFlow is used to control OpenFlow switch route information, but other methods such as CLI are also used to control non-OpenFlow-based network appliances, such as firewalls and load balancers. Orchestration functions are built on top in the cloud management system to control both cloud resources and network resources. The orchestration functions also provide management capabilities for several items such as customer attributes, service orders, billing, failures as well as cloud and network resources. Having the customer/operator portal, it makes self-provisioning possible. It is thought that OpenFlow would make network operation automaton easier, however the truth is that OpenFlow contributes only partially, i.e., the importance of design and its making of the whole system to realize automation and self-provisioning takes less time to build and operate networks that also reduces human error.

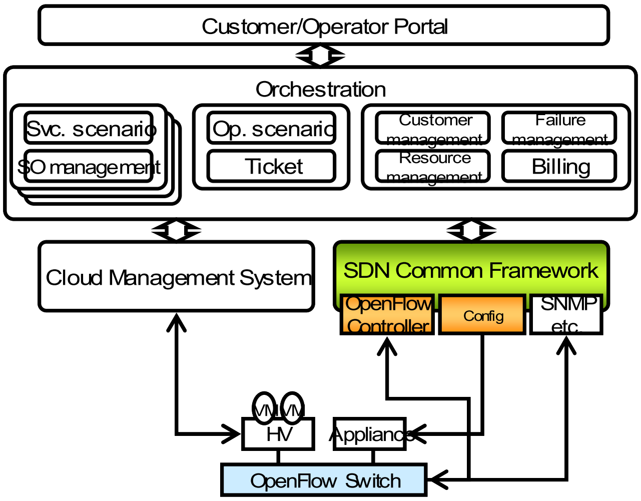

**Fig. 2** Self-provisioning mechanism for cloud and network resources in datacenter.

OpenFlow is also used for inter DC network and enables automatic change of network bandwidth between DCs, which is useful for data backup among DCs.

### 3.2 Developing the SDN Controller

As expected, SDN provides the ability to develop on our own the SDN controller to create network functions by software in a timely manner. With our own controller, it will become possible to create original network applications and to build the environment for choosing price-competitive switches. NTT Com has built a controller prototype in cooperation with NTT Innovation Institute, Inc. (NTT I$^3$) and NTT Lab. Figure 3 shows its configuration.

As an example of OpenFlow controller platform based on RYU [6] and built above it is a SDN common framework called BigBoss. It enables us to develop original network

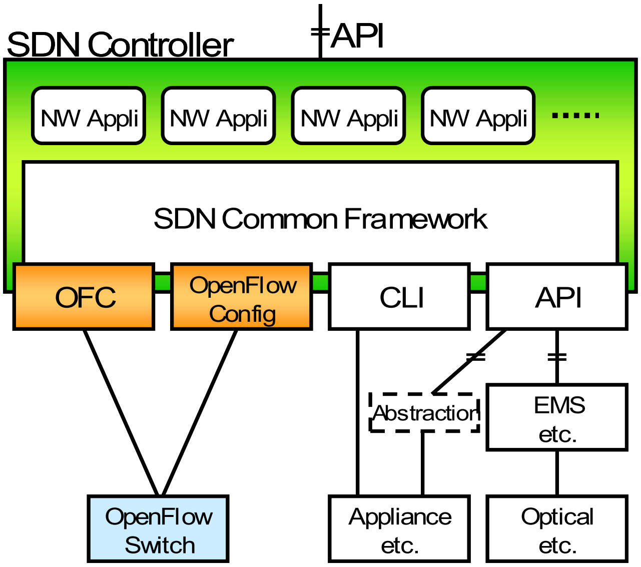

**Fig. 3** Example of SDN controller.

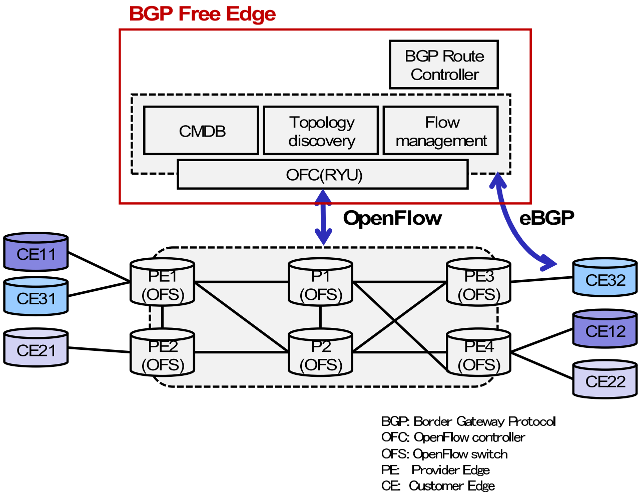

**Fig. 4** BGP free edge.

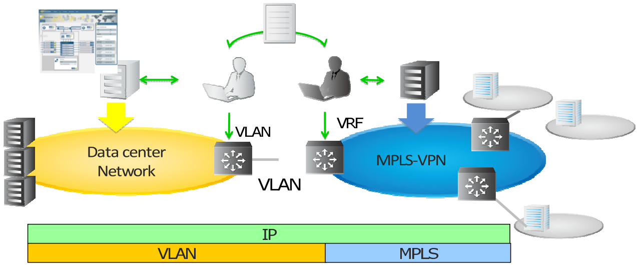

**Fig. 5** Boundary of cloud network and MPLS-VPN.

applications. It should be mentioned that an important point is each network application can use many common function modules provided from the OpenFlow controller and the SDN common framework. Anything in the SDN common framework should be open source so that many companies and laboratories can develop network applications in the same development environment. RYU is already publicly open as open source.

### 3.3 Examples of Developed Network Applications

Network application developed on the top of RYU/BigBoss is introduced. One is an application, called BGP Speaker, which enables BGP protocol processing on a server. The following prototypes are being developed by using this application.

(1) BGP Free Edge

BGP Free Edge technology enables to offload BGP control on edge routers to a centralized control plane on a carrier's server. It allows building carrier IP-VPN service [7] on an OpenFlow network. Current IP-VPN service networks are built on expensive MPLS routers which require advanced skills to build and operate such networks. It is one of the causes of high Capital Expenditure (CAPEX) and Operating Expense (OPEX) on IP-VPN service. BGP Free Edge technology allows as Fig. 4 shows, offloading BGP functionality on Provider Edge (PE) routers in IP-VPN service network to a centralized BGP Free Edge system. The centralized system manages and operates the information for Customer Edge (CE), such as routing information, forwarding information, network view, etc. OpenFlow Controller (RYU), a part of BGP Free Edge system executes path configuration for VPN in OpenFlow switches by using OpenFlow protocol. We believe that replacing MPLS routers with inexpensive commodity OpenFlow switches can drastically reduce the cost of IP-VPN service infrastructure and operation. The configuration in Fig. 4 shows the demonstration at Interop Tokyo 2012 [8].

(2) Automating the connection between cloud network and VPN

Connecting a tenant network in a cloud data center with an appropriate MPLS-VPN that the tenant network belongs to (e.g. [9]), is a time-consuming operation. Figure 5 shows how tenant networks in cloud data center and MPLS-VPNs are connected. In the cloud data center, VLANs are used to isolate tenant networks. On the other hand, Label Switched Paths (LSPs) are used to isolate customer networks in the MPLS-VPN network. Upon receipt of a customer order, the mapping information between VLAN and LSP is manually created. Then the VLAN configuration in a gateway device between the data center and MPLS-VPN networks and Virtual Routing and Forwarding (VRF) configuration in edge routers are updated. The above configuration change by network operators takes long time to process. Thus, to address this issue we created the automation of the connection between cloud and MPLS-VPN networks by using SDN technology (see Fig. 6). Cloud's tenant network configured as layer-2 VLAN connects with MPLS-VPN by which tenant belongs to in the Inter-AS option B function of MPLS-VPN. Intra DC network gets LSP and tenant mapping information from MPLS-VPN by using BGP, and then it exchanges the

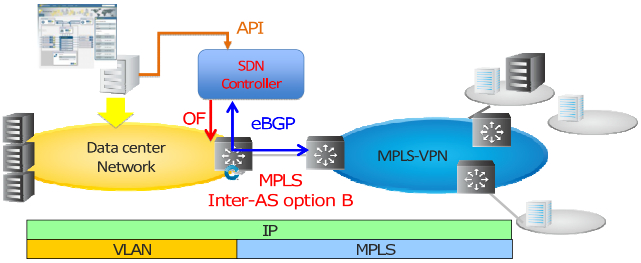

**Fig. 6** SDN use case for cloud network and MPLS-VPN boundary.

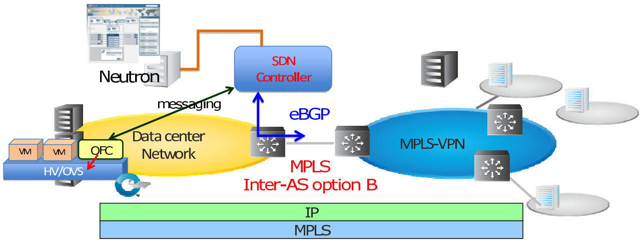

**Fig. 7** SDN use case for integrated cloud network and MPLS-VPN.

headers of VLAN and MPLS by using OpenFlow. The SDN control application can create the mapping between both the VLAN and LSP by using the above both information. The SDN control application updates the flows in an OpenFlow switch that works as a gateway device by using the mapping information. A great amount of operational cost reduction is expected by automating the connection setting of tenant and MPLS-VPN networks.

(3) Integration of cloud network and VPN

Instead of using layer-2 VLAN networks to isolate traffic of tenants, MPLS could be used (see Fig. 7). For example above (2), LSP is used from MPLS-VPN to the border gateway device, but instead LSP could be used to a software switch (e.g. [10]) in a hypervisor. Mechanism is the same as in the example (2), however, the virtual switch in hypervisor binds VM and LSP. Because it does not use VLAN, it is possible to accommodate more than 4094.

### 3.4 Another Network Application Example

Another application is OAM functionality with OpenFlow, which is a must in a large carrier-grade wide area networks. Adopting OpenFlow/SDN technology to a carrier network allows flexible configuration management and operation, however, it also complicates the diagnostic operation. The expected features of OAM are the following items.

- Health checking mechanism on service and connectivity
- Detection mechanism on failed devices and the affected logical networks due to failure
- Anomaly detection mechanism, redundant failover mechanism, etc.

Implementing only necessary OAM features in software running a SDN controller is rather easy. Figure 8 shows the example of an implementation of simple OAM features in OpenFlow networks. Upon receipt of a request from an operator, the OpenFlow Controller builds and sends a packet for examining, then verifies test packets from Customer Premises Equipment (CPE) or OFS(s) in the network. The mechanism uses Ping function for layer-3 and EtherOAM function such as Loop Back (LB) and Continuity Check (CC) for layer-2. But Ping or EtherOAM is not good enough to verify the flow-based packet forwarding in OpenFlow. A new way to verify finer flow tables than before is necessary.

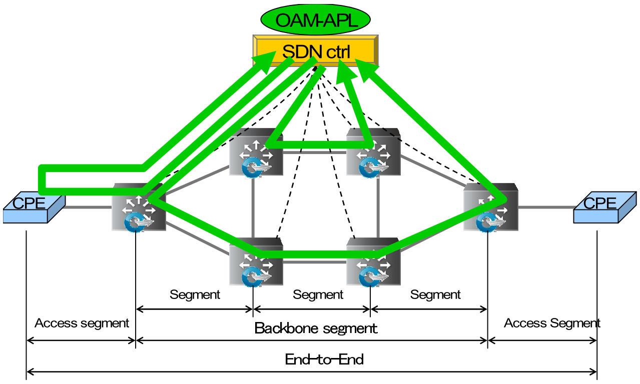

**Fig. 8** Example of implementation of OAM functions.

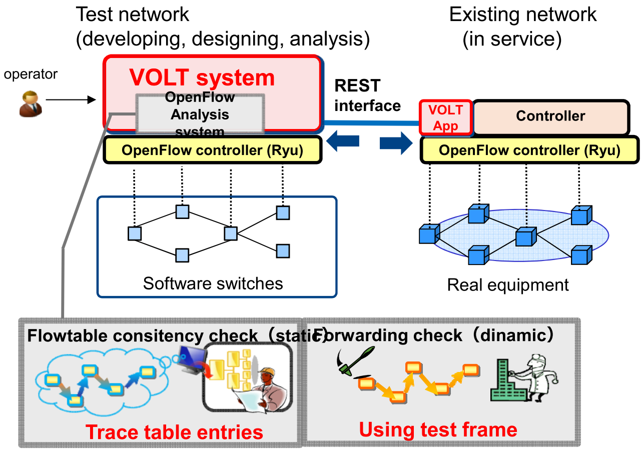

**Fig. 9** VOLT.

So we developed a Versatile OpenFlow validator (VOLT), providing OAM functionality for designing and testing OpenFlow networks (Fig. 9).

This system can build a testbed to design and test a network under the same conditions of a real network. For example, the testbed is built based on a topology and routing information in the real network. The system can verify the correctness of the flow tables to correspond with topology and routing information with the traffic data in the real network. This feature is useful for designing a network and also for failure analysis of networks in operation.

## 4. Issues and Expectations for SDN

SDN consists of multiple component technologies: switch, controller, application, etc. To realize a particular solution, it's necessary to choose some of these component technologies and evaluate the combination of the chosen technologies. It is also vital to make clear problems about deploying the solution from the perspective of operation, quality, cost, etc. not only on by paper but by experimenting. We believe that the guidelines for designing, deploying, and operating SDN-based networks are important to make SDN technology widely used.

First, we describe the issues and expectations about OpenFlow/SDN-based systems. The OpenFlow specification includes both OpenFlow protocol and OpenFlow switch specifications. As for hardware switch problems, current switches implement only OpenFlow matching and action functions on the basis of available commoditized switch LSIs and don't have sufficient number of flow entries due to hardware limitation. We expect future LSIs to resolve these problems.

On the other hand, representing the software switch is Open vSwitch (OVS) [10]. It is configured by user space daemon with OpenFlow protocol and kernel space packet match cache for high speed transfer. OVS is expected to provide flexible networking but has the following problems.

- OVS sets the packet match cache immediately upon receipt of the first packet, but it is possible to cause packet disorder if the next packet arrives before setting the cache
- Will need to rely on the for expanding the function in the flow rule where redundancy by bonding is needed.
- Very difficult to add new functions because of complicated design. In addition, we can have no Operating System vendor support when the change is required to the Linux kernel
- OpenFlow protocol version implemented in OVS is not the latest.
- Less performance due to overhead of cache management and kernel/user space communication. This happens in the case of traffic pattern with low cache hit ratio for various destination packets.

From the point of view of problems, a newly designed open source switch would be necessary. The combination of DPDK technology [4] with high-speed transfer, using commodity server, a simple designed software switch, utilizing only user space seems to be effective. Various functions become possible by having various applications on servers, for example, TCP high speed function, Session Border Controller (SBC) used in VoIP network, and DDOS detection.

As an intermediate solution between software and hardware a switch using Network Processor Unit (NPU) is a candidate. Programming NPU's micro code enables flexible function implementation for packet processing. It is far easier to add functions to a NPU-based switch than redesigning LSI. As programming NPU micro code needs the deep understanding of the abstracts of NPU hardware, to make NPU use easier we need to examine it's environment. NPU could be implemented in a server's PCIx expansion card, and by doing so high speed and complexity is possible by using complex process together with software on the server. However, the number of ports and interface speed might be limited because of packet processing capability of NPU.

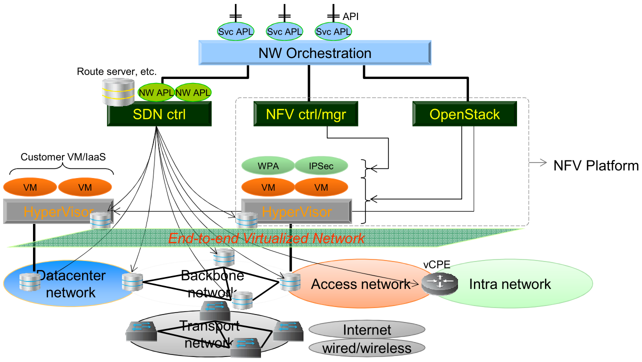

**Fig. 10** SDN, NFV and cloud orchestration for carrier networks.

We should choose switches from hardware, software, and proper use-case for the time being.

Next we describe the expectation for the combination of SDN and NFV. SDN defines a way to manage how data planes are controlled by software. On the other hand, NFV as a way to realize network functionality in software, has been under active discussion too at ETSI. The combination of SDN and NFV technologies allows a more flexible and effective carrier network. Figure 10 shows an example of how to deploy SDN and NFV in a carrier network. SDN and NFV are expected to provide backbone networks with optimization by transport networks and multi-layer SDN combination. Utilizing layer-2 and layer-3 network emulation without expensive network devices and traffic engineering based on traffic information measured in real time. Traffic engineering should realize the dynamic path selection based on latency, cost, available resource, etc. The network load is desired to be balanced across multiple links and expected to provide redundancy in the case of some of the link(s) failure. The failover must be executed quickly and in the case of the expected failure, hopefully the failover occurs without packet loss. An access network is expected to build overlay virtual networks on the top of multiple access media for the rapid connection setup and seamless operation with backbone networks.

NFV needs to work together with a cloud infrastructure such as OpenStack and provide software-based appliances running on a virtual machine on demand. OpenFlow switches allow traffic steering in a flexible way with NFV so that network features can be easily added to the existing SDN/NFV based network. NFV is also expected to reduce the cost of CPE in customer premises by virtually realizing CPE functionality on centralized servers. In Fig. 10, in addition to the SDN controller, the NFV controller is deployed to manage NFV virtual appliances to the above features.

A system to orchestrate both SDN and NFV controllers is necessary. We believe combining cloud resource and NFV functionality is the future common model to provide customers service instead of providing a standalone feature of a network as a service. Deploying an application to manage services on an orchestration system is necessary to realize the above service. Technology for such infrastructure which enable making network and cloud resource management work together is important too.

## 5. Conclusion

OpenFlow/SDN is a tool to implement network technologies and enables quick and easy development of new network functions. Studies on networking system itself are important, but the knowledge and perspectives through proof of concept are more important. High expectation of OpenFlow/SDN technologies for carrier networks relies on aspects of new function and Opex/Capex, but these technologies are still immature and it is difficult to distinguish what it can and what it can't do. As for deploying OpenFlow/SDN to commercial networks, it is important to consider not only replacing existing network and equipment but also impact to service and operation. It is necessary to promote investigation of OpenFlow/SDN from these perspectives.

### References

[1] N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner, "OpenFlow: Enabling innovation in campus networks," SIGCOMM Computer Communication Review, vol.38, no.2, pp.69–74, April 2008.
[2] http://www.openflow.org/
[3] http://www.etsi.org/technologies-clusters/technologies/nfv
[4] http://www.intel.com/p/ja_JP/embedded/hwsw/technology/packet-processing
[5] http://www.ntt.com/bhec/
[6] http://osrg.github.com/ryu/
[7] E. Rosen and Y. Rekhter, "BGP/MPLS IP virtual private networks (VPNs)," RFC4364, Feb. 2006.
[8] http://www.interop.jp/2012/pavilion/show_case.html
[9] L. Martini, E. Rosen, N. El-Aawar, and G. Heron, "Encapsulation methods for transport of Ethernet over MPLS networks," RFC4448, April 2006.
[10] http://openvswitch.org/

**Yoichi Sato** received the B.S. degrees in Electrical Engineering from Science University of Tokyo in 1986. He joined the Electrical Communication Laboratories of Nippon Telegraph and Telephone Corporation in 1986. He was engaged in research and develop of Broadband ISDN based on ATM technology. He was transferred Long-distance Division of NTT in 1995. From 1999 he work for NTT Communications Corp. and was engaged in hardware development of MPLS-TP. His current interests include OpenFlow/SDN and cloud computing integration. He received the Young Engineer Award in 1993 from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan.

**Ichiro Fukuda** received his B.E. and M.E. degrees from Waseda University in 1999 and 2001 respectively. He has been working in NTT Multimedia Communications Laboratories, Inc. for 2 years. Prior to joining NTT MCL, he worked for NTT Communications over 10 years. He involved in various development projects including ATM and IP/MPLS technology based network services.

**Tomonori Fujita** received his B.E. and M.E. degrees from Waseda University in 1998 and 2000 respectively. He has worked in Nippon Telegraph and Telephone Corporation since 2000 and has engaged in research on operating systems. He is a member of IPSJ and ACM.