NFV SDN Overlay Network Technology

Technology Reports

Special Articles on Network Functions Virtualisation—Toward a Robust and Elastic Network—

Introduction of SDN Technology for Achieving NFV

The virtualization of network functions by NFV was introduced into the NTT DOCOMO network in March 2016. Since NFV eliminates the binding of communications software to hardware, a method is needed that can dynamically and flexibly change the configuration of the physical network composed of routers and switches. NTT DOCOMO has adopted SDN technology for this purpose. This article describes SDN technology for achieving NFV.

Core Network Development Department
Yusuke Okazaki
Takuya Kitade
Taisuke Yoshida

1. Introduction

The use of Network Functions Virtualisation (NFV)*1 technology is expanding to provide the functions of a carrier network in a virtual manner through software technology on general-purpose hardware instead of dedicated equipment. NFV makes it possible to run a Virtual Machine (VM)*2 independent of hardware characteristics by deploying communications software on a virtual layer (hypervisor) installed on general-purpose hardware.

With virtualization, hardware resources (CPU, memory, HDD, etc.) that have heretofore been constrained by a physical configuration can adopt a logical configuration, which enables resources to be used based on the concept of a resource pool*3. Virtualization also enables the common use of general-purpose hardware instead of expensive dedicated hardware, which facilitates the sharing of hardware resources. This makes for quick and flexible construction of a communications network while enhancing fault resistance.

Yet, it is still difficult to extract the maximum effect of NFV in a network consisting of conventional routers and switches. For example, in a conventional network, communications software runs on dedicated hardware and physical networks are individually prepared. In NFV, however, the resource pool contains different types of communications software on general-purpose hardware, so there is a need to prepare a common physical network from the viewpoint of efficient operation.

In addition, a network that accommodates communications software that runs as VMs under NFV must be able to dynamically and flexibly make changes to the network configuration. That is, it must be able to deal with VMs that are created and deleted on any hardware and with the movement of VMs between different pieces of hardware.

As a solution to the above issues,

*1 **NFV:** Achieving a carrier network on general-purpose hardware through virtualization technology.
*2 **VM:** A computer created in a virtual manner by software.

NTT DOCOMO has decided to adopt Software Defined Networking (SDN) technology.

In this article, we examine the requirements and issues surrounding a network incorporating NFV, describe how SDN provides a solution to these issues, and touch upon the future outlook for SDN.

2. NFV Requirements and Issues

2.1 Sharing of Network Resources

Implementing different types of communications software on general-purpose hardware means that different types of communications software must also be accommodated on the physical network composed of routers, switches, etc. that accommodate general-purpose hardware.

Consequently, for the network too, it must be possible to share and control the physical network in a flexible manner. Developing a method to do so has become an issue—there is a need for technology that can be used to flexibly construct multiple virtual networks on the same physical network.

2.2 Tracking VM Creation/Movement

NFV enables VM creation and VM movement on general-purpose hardware. The network is therefore required to track the creation and movement of VMs. To be more specific, a network path must be set up when a VM is created to give that VM a communications capability. Furthermore, when a VM moves to other hardware, it must carry with it the same IP address*4 and Media Access Control (MAC) address*5 as before the move, which means that the network path prior to the move must be deleted and a new network path to the destination of the move must be set up.

In a conventional network, however, network paths are fixed, so it is not possible to dynamically switch network paths when a VM is created or moved. As a result, a created or moved VM cannot communicate with the network (**Figure 1**). There is therefore a need for dynamic switching of network paths.

3. Solution to Issues by SDN

3.1 SDN Overview

1) SDN Architecture

Conventional network architecture consists of a combination of routers and switches, each of which incorporates all necessary functions in one piece of hardware. SDN architecture takes a second look at this conventional architecture and divides it into a Control Plane (C-Plane)*6 and Data Plane (D-Plane)*7. The C-Plane refers to software-based functions such as path calculation and the D-Plane refers to hardware-based functions such as data transfer.

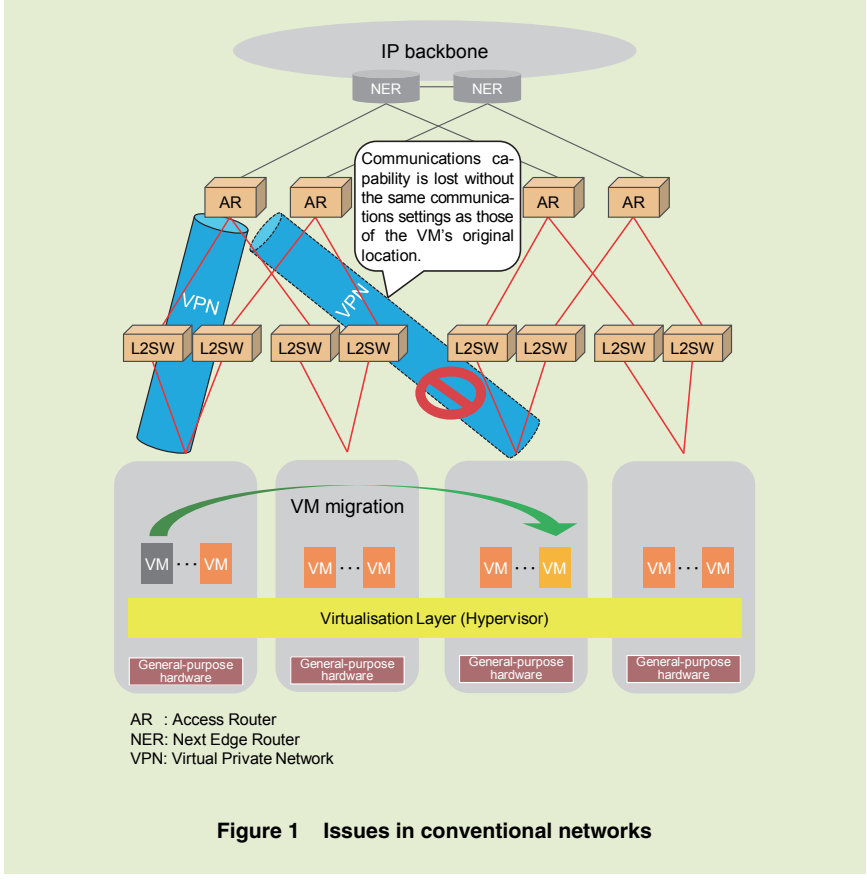

Figure 1 Issues in conventional networks

*3 **Resource pool:** A set of resources achieved by bundling together many units of hardware each possessing certain types of resources (CPU, memory, HDD, etc.). Various types of virtual machines can be created from a resource pool.

*4 **IP address:** A unique identification number allocated to each computer or communications device connected to an IP network such as an intranet or the Internet.

*5 **MAC address:** A 12-digit fixed physical address allocated to an Ethernet board.

*6 **C-Plane:** Network path control function.

*7 **D-Plane:** Data transfer function.

While conventional network devices must be set individually, SDN architecture enables network devices to be centrally controlled via a SDN controller (**Figure 2**).

Centralized control can facilitate total optimization, and specifying the Southbound IF*8 between the C-Plane and D-Plane as open source can enable a multi-vendor environment to be achieved and costs to be reduced.

2) Centralized Control and Hybrid Control

SDN architecture can be broadly divided into two types: centralized control and hybrid control. The features of each are summarized in **Table 1** and their operating schemes are shown in **Figure 3**. In centralized control, the SDN controller collects status information from each switch and performs path calculations. Hybrid control, on the other hand, uses both the SDN controller and a distributed controller mounted on each

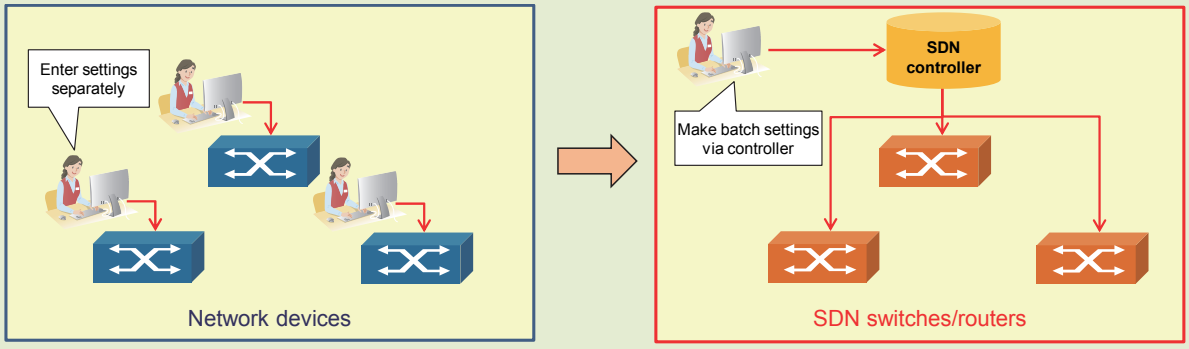

Figure 2 Device control by SDN controller

Table 1 Features of SDN architecture

	Centralized control	Hybrid control
C-Plane	Executed only by SDN controller	Executed by both SDN controller and switches
Path calculation	SDN controller collects status information from each switch and calculates paths.	SDN controller sets transfer policy in each switch and each switch autonomously calculates paths.
Path switching	SDN controller mediates in path switching.	SDN controller does not mediate in path switching.

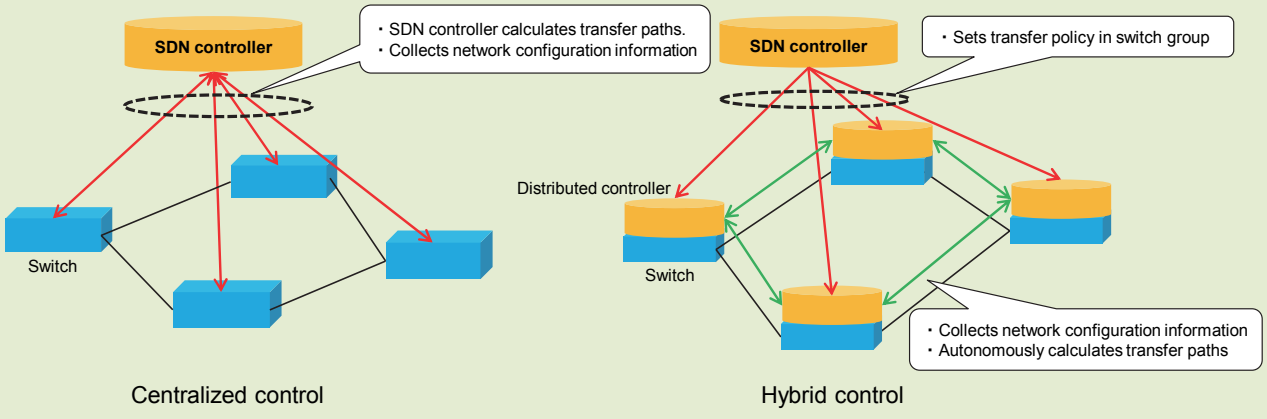

Figure 3 Operation by centralized control and hybrid control

*8 **Southbound IF:** The interface that connects the SDN controller and the network devices that it controls.

switch to control the network.

In the beginning, SDN architecture was mostly of the centralize control type, but in recent years, hybrid control solutions have gained in popularity. In hybrid control, the SDN controller sets a transfer policy in each switch so that each individual switch can then calculate routing information on its own. Compared to centralized control in which the SDN controller calculates all paths, hybrid control has the advantage of reducing the load on the SDN controller, which makes for greater extendibility. Hybrid control also means that data transfer and path calculation can continue if the SDN controller should be down, and it enables high-speed switching within the network through autonomous distributed control. For these reasons, NTT DOCOMO has adopted hybrid control.

3.2 Overlay Network Technology

We use overlay network technology to satisfy the requirement that "multiple types of communications software must be accommodated on the same physical network." Overlay network technology enables multiple logical L2 networks to be created on the same physical L2 network by applying tunneling techniques such as VLAN or Virtual eXtensible LAN (VXLAN)*9 (**Figure 4**).

However, conventional overlay network technology requires that all network devices used in configuring a logical L2 network be individually set. In response to this constraint, the SDN technology to be introduced defines the settings to be made in network devices for configuring logical L2 networks as a "transfer policy" and enables this policy to be set from the SDN controller. This approach enables the settings that need to be made in each network device to be executed in batch, that is, all at the same time.

Having the SDN controller make batch settings in this way provides flexibility in handling the movement of VMs (that include communications software) among general-purpose hardware in NFV (**Figure 5**).

3.3 Tracking of VM Creation and Migration

Next, to satisfy the requirement that "network paths must be dynamically switched when creating or moving a VM," we link the overlay network that uses SDN technology to the Virtualised Infrastructure Manager (VIM)*10 that

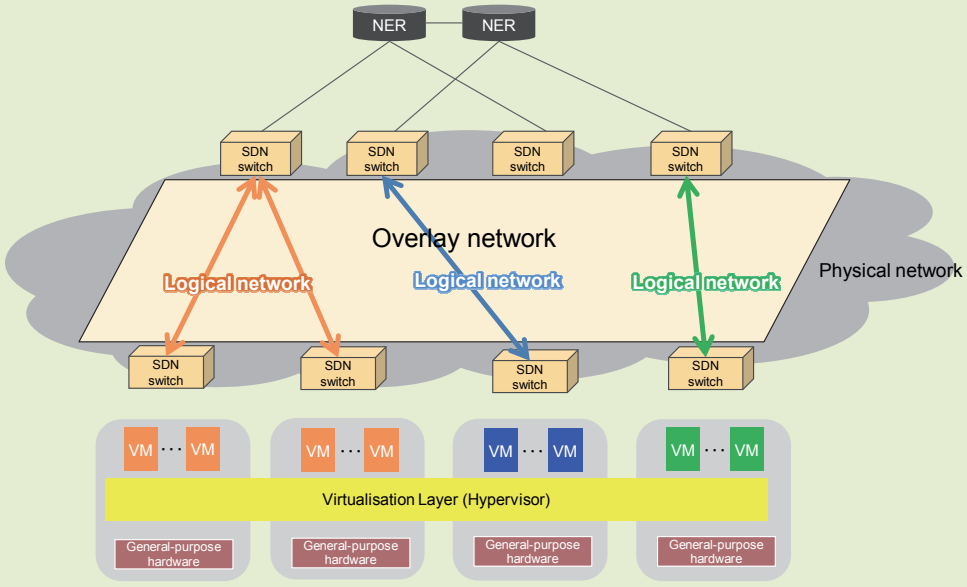

Figure 4 Overlay network

*9 **VXLAN:** Technology for configuring logical L2 networks on top of a network configured on L3. While conventional VLAN could only configure 4,094 L2 networks, VXLAN can configure up to 16,770,000 L2 networks.

*10 **VIM:** Component controlling NFVI (see *14).

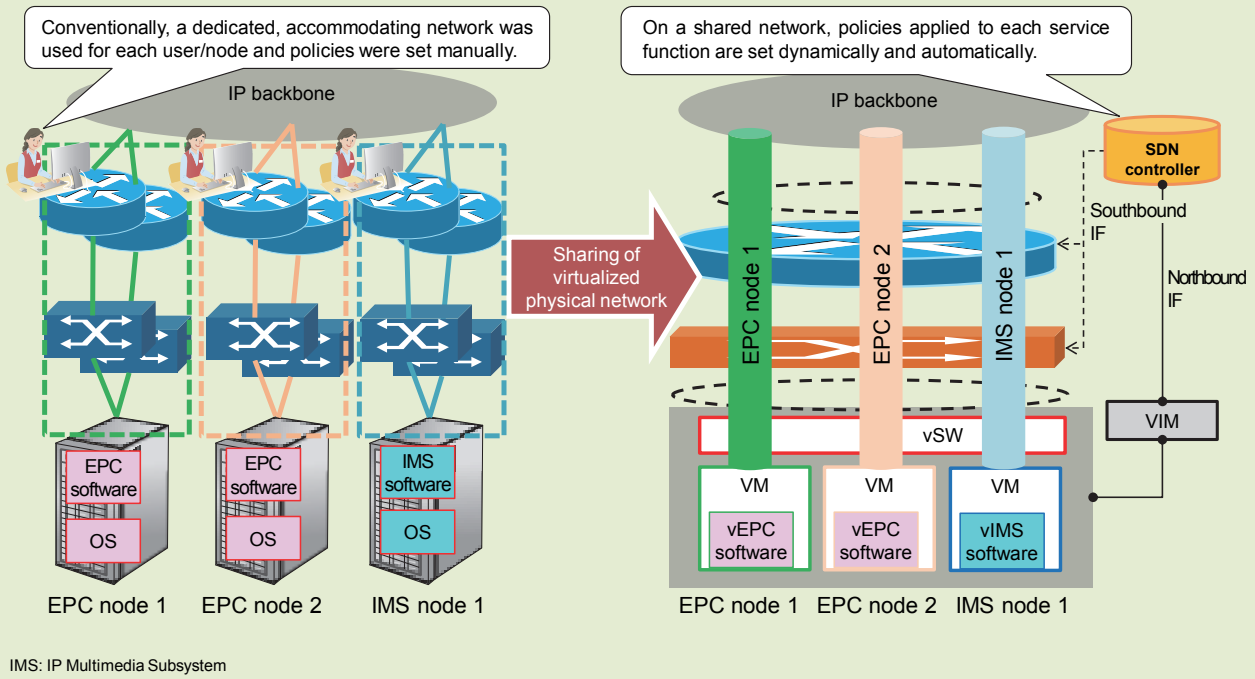

Figure 5 Accommodation of multiple service functions

controls the creation and migration of VMs.

The connection scheme between VIM and the SDN controller is shown in **Figure 6**. Here, installing the SDN-controller OpenStack*11-based ML2 plug-in on the VIM side achieves a Northbound IF*12 for linking purposes. In the resulting configuration, setting of the virtual SWitch (vSW)*13 on the NFV Infrastructure (NFVI)*14 side is performed by VIM without using the SDN controller as an intermediary.

This linking function enables information on VM migration to be conveyed from VIM to the SDN controller and the switching of network paths to be dynamically performed. In the following, we describe linking operations between VIM and the SDN controller at the time of VM creation or migration, that is, at the time of VM healing*15 or scaling*16.

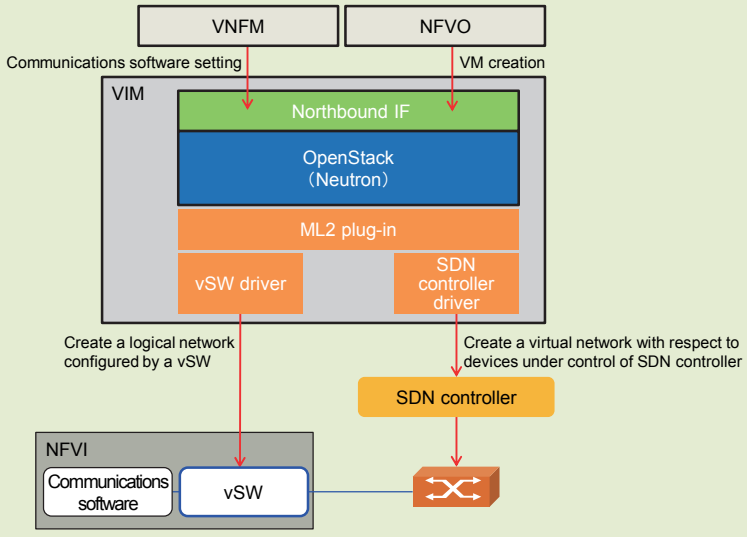

Figure 6 Interface configuration of VIM-SDN controller

1) VM Healing/Scaling Support

A requirement applied to the net-

*11 **OpenStack:** Cloud-infrastructure software that uses service virtualization technology to virtually operate multiple servers on a single physical server and to allocate a virtual server to each cloud service used by the user. OpenStack is provided as open source software.

*12 **Northbound IF:** The interface that connects the SDN controller and upper-level software such as VIM.

*13 **vSW:** A virtual switch achieved by software.

*14 **NFVI:** Physical resources for executing virtual machines. In this article, NFVI is defined as general-purpose hardware.

*15 **Healing:** A procedure for restoring communications software to a normal state in the event of a hardware or VM failure by moving the VM to (or recreating the VM on) hardware operating normally.

work side at the time of VM healing or scaling is that VM migration that holds on to the same IP/MAC addresses must be dynamically supported. This is because session*17 continuity is a requirement characteristic of mobile communications. In SDN, the SDN controller links with VIM and makes on-demand settings to the physical network on receiving notifications of any VM creation or migration from VIM. In this way, the user session can be maintained and communications can continue even when VM switching occurs at the time of a VM failure.

An example of tracking VM migration and switching L2 paths on the network is shown in **Figure 7**. As shown by operation ① in the figure, VIM instructs NFVI to move a VM while simultaneously instructing the vSW side to create and delete L2 paths. A vSW has the role of connecting the NFVI physical port and the VM virtual port. Here, the network connection between the VM and the VM's source NFVI is deleted and a network connection between the VM and the VM's destination NFVI is created.

Once operation ① completes, VIM instructs the SDN controller side to create and delete L2 paths as shown by operation ② in the figure. On receiving this instruction from VIM, the SDN controller enters settings in SDN switches under its control as shown by operation ③. As a result, the L2 path between the VM's source NFVI and SDN switch (1) and that between and SDN switch (1) and SDN switch (0) are deleted and the L2 path between the VM's destination NFVI and SDN switch (2) and that between and SDN switch (2) and SDN switch (0) are created.

In short, operations ① to ③ in the figure result in the simultaneous deletion of pre-VM-migration L2 paths and in the automatic creation of post-VM-move L2 paths.

4. Future Outlook for SDN

4.1 Application of SDN to WAN

Now, in the initial period of NFV deployment, the applicable domain of VM migration is taken to be the hardware existing on the same LAN within a telecom office. In the future, we can envision the migration of VMs across a WAN that interconnects different telecom offices. This capability has the potential of shortening the construction period of telecom offices and of adding/subtracting facilities (as in system recovery after a disaster and resource sharing across telecom offices), all of which will involve changes to WAN settings (**Figure 8**). To this end, the following studies are needed.

(1) Linking between the WAN Infrastructure Manager (WIM)*18 and

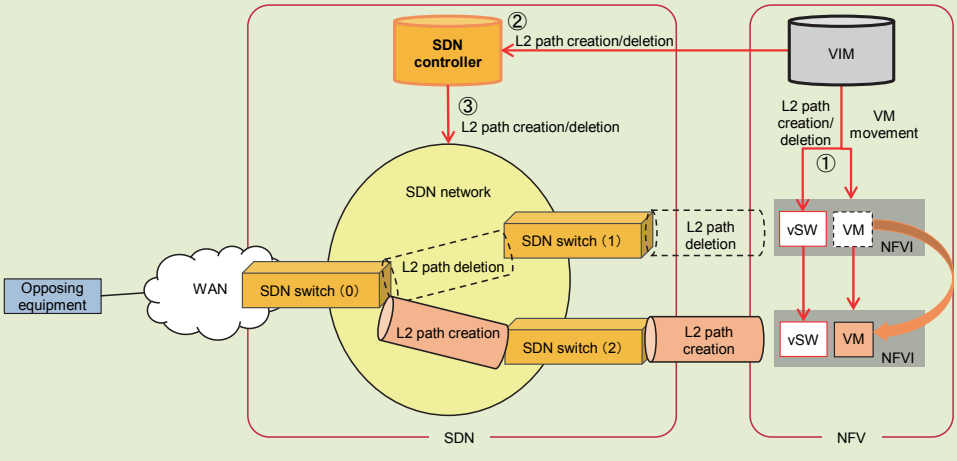

Figure 7 Example of dynamic network settings when moving a VM

*16 **Scaling:** The optimization of processing power by increasing or decreasing VMs that configure communications software whenever processing power is insufficient or excessive according to hardware and VM load conditions.

*17 **Session:** A virtual communication path for transmitting data or the transmission of data itself.

*18 **WIM:** Component controlling NFVIs or VIMs across a WAN.

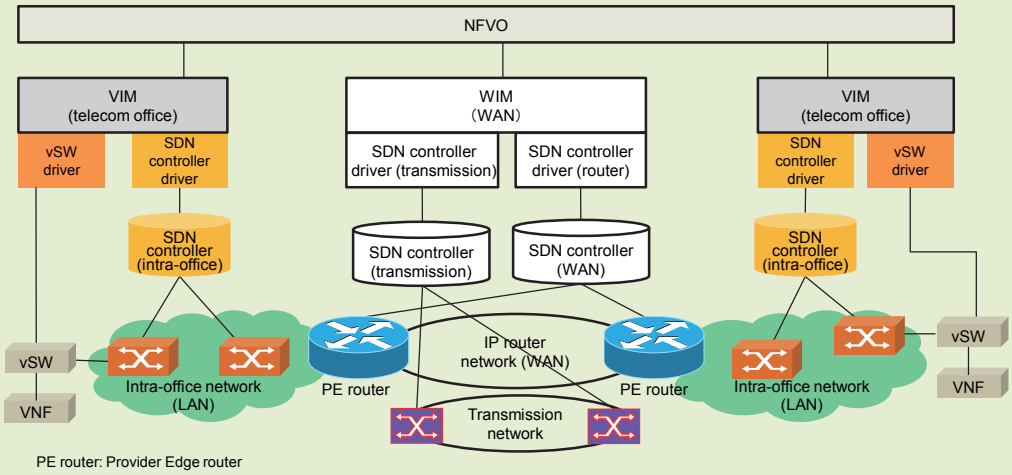

Figure 8 Application of SDN to WAN

NFV Orchestrator (NFVO)*19 and between VIM and NFVO

(2) Linking between router equipment and optical transmission equipment under VIM/WIM using a VIM/WIM plug-in structure

5. Conclusion

In this article, we described network requirements for introducing NFV into the NTT DOCOMO network and SDN as a method for satisfying those requirements. We also touched upon the future outlook for SDN.

This SDN technology was introduced as a network for accommodating virtualised Evolved Packet Core (vEPC)*20, which was commercially deployed in March 2016 [1].

SDN is an essential technology for achieving NFV, and given expectations that virtualization technology will continue to evolve, we consider the further development of SDN to be just as important. Looking to the future, NTT DOCOMO is committed to developing an increasingly flexible and advanced network while keeping an eye on the latest technology trends.

Reference

[1] T. Kamada et al.: "Practical Implementation of Virtualization Platform in NTT DOCOMO Network," NTT DOCOMO Technical Journal, Vol.18, No.1, pp.20-28, Jul. 2016.

*19 **NFVO:** Component managing various types of communications software from creation to deletion and performing operations and management across the entire system.

*20 **vEPC:** An IP-based core network specified by 3GPP for LTE or other access technologies. Communications software to enable EPC to function like a virtual machine.