SDN Solution for Wide Area Networks

● Takuya Miyashita ● Tomohiro Suzuki ● Toshio Soumiya ● Akiko Yamada

The expanding use of cloud computing is driving a growth in the amount of data traffic carried by wide area networks (WANs). Software-defined networking (SDN) is potentially capable of handling this growth efficiently while offering new services. SDN makes networks more flexible, leading to more timely service implementation and more efficient network operation. Such timely service implementation should enhance network value. Although SDN has already been applied to networks within data centers, it has not yet been applied to WANs. This is because WANs support social infrastructure, which demands strict quality and functionality, among other requirements. However, the application of SDN to WANs would promote open-source platforms, which would help eliminate vendor lock-in and support market expansion through open innovation. This paper first describes the challenges inherent in the application of SDN to WANs, presents Fujitsu's SDN solution for WANs, and introduces Fujitsu's projects related to WAN control technology and its open-source initiatives.

1. Introduction

Traffic in wide area networks (WANs) supporting social infrastructure is increasing due to advances in cloud computing and growth of video services. It is said that about 50 billion devices will eventually be connected to the network in the Internet of Things (IoT) era, so this increase in traffic is expected to accelerate in the years to come. This increasing demand for traffic capacity has made providing circuits efficiently and increasing sales by offering new services pressing issues for telecom carriers. There are growing expectations that software-defined networking (SDN) is a way to resolve these issues.

The basic concept of SDN is shown in **Figure 1**. SDN is the general term for the technology and concept of creating a virtual network environment. A variety of definitions exist, but SDN is said to provide the general features listed below.[1),2)]

1) Programmability

Enables not only control of network resources but also creation of applications for controlling those resources

2) Network resource abstraction

Enables abstraction of network resources by using a standard model and presentation to the users of a network different from the underlying physical entities

3) Separation of control functions from hardware and provision of centralized control

Separates the control mechanism from network hardware, enabling the use of general-purpose network equipment and devices

4) Agile service introduction

Enables the provision of network services in a significantly shorter period of time

These features enable the effective use of network bandwidth and the provision of an even higher level of added value through the network.

In this paper, we introduce Fujitsu's approach to SDN. First, we point out the issues in applying SDN to WANs and present Fujitsu's SDN solution. Next, we describe Fujitsu's distributed network control technology, and finally, we introduce open source activities and Fujitsu's contribution.

2. Issues in applying SDN to WANs

In addition to controlling operations within a limited area such as a data center, the following requirements will have to be met in applying SDN to

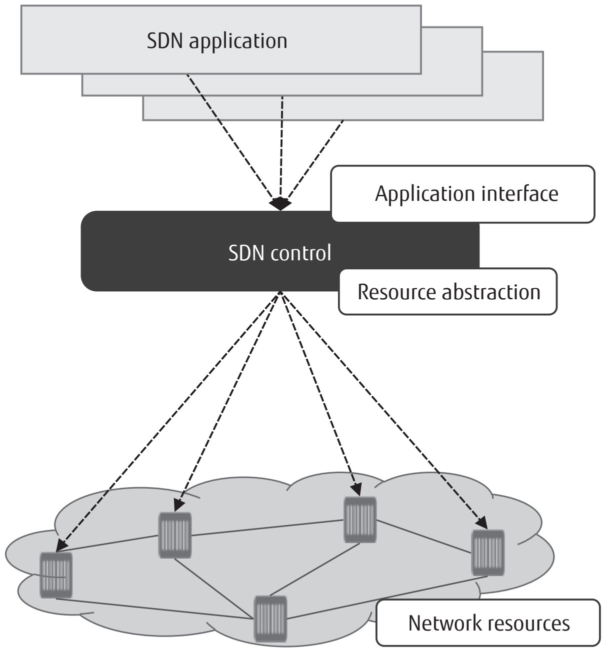

Figure 1
Basic concept of software-defined networking.

WANs.

First, a WAN has a multilayer configuration: a packet layer consisting of routers and switches and a transport layer consisting of transport network equipment. The operating methods for these two layers differ, and there is an operation department for each layer. When a network is being launched, these departments need time to coordinate their activities, and shortening this time period is one requisite to achieving efficient operations. Next, the network itself is a multivendor environment, and the control method for equipment of different vendors differs, which can be a factor in operating errors. To prevent such errors from occurring, there is a need for a control mechanism that makes it unnecessary for operators to worry about the presence of multivendor equipment.

Telecom carriers also expect to generate new revenue with SDN (that is, to monetize SDN). At present, the development of a new network service usually requires system extensions, which calls for careful planning. This makes it all the more difficult to launch a new service in a reasonable period of time. An environment is needed in which a telecom carrier can put a new service through a "small start" and quickly provide a service that can set it apart from its competitors.

Additionally, there are already many devices deployed in a WAN, and replacing all of them with SDN-compliant devices would not be realistic since such a project would produce service interruptions while driving up the cost of device deployment and testing. There is therefore a need to achieve a smooth, low-cost transition from an existing network to one using SDN while maintaining existing services.

A WAN may support a social infrastructure used by a large number of users, so any problems in the functions of the controller when applying SDN can have a major effect on the system and users. Such a controller must therefore exhibit high reliability (carrier-grade quality) to ensure stable network operation.

Due consideration must also be given to operations after completing a migration to an SDN network. Long-term, sustainable operation is a precondition for a WAN, so a mechanism is needed to maintain system development into the future. In short, telecom carriers must be able to expand the network on an open SDN platform.

The above requirements in applying SDN to WANs can be summarized as follows.

1) Efficient control of composite network
2) Quick creation of monetizing opportunities
3) Ensuring carrier-grade quality
4) Low-cost, smooth migration to SDN network
5) Provision of open platform

The following section describes a solution to meeting these requirements.

3. SDN solution for WANs using FUJITSU Network Virtuora NC

Fujitsu has resolved the issues described above by providing FUJITSU Network Virtuora NC (Virtuora NC), a product that serves as an SDN controller for WANs (**Figure 2**).

1) Efficient control of composite network

Virtuora NC enables multilayer control of communication paths and bandwidths on both the packet layer and transport layer. For the packet layer, it manages network topology. It monitors the layer by using the Simple Network Management Protocol (SNMP)[3] and controls the layer by using a command-line interface (CLI), the Network Configuration Protocol (NETCONF),[4] and the OpenFlow[5] protocol. For the transport layer, it manages the physical connections with the packet

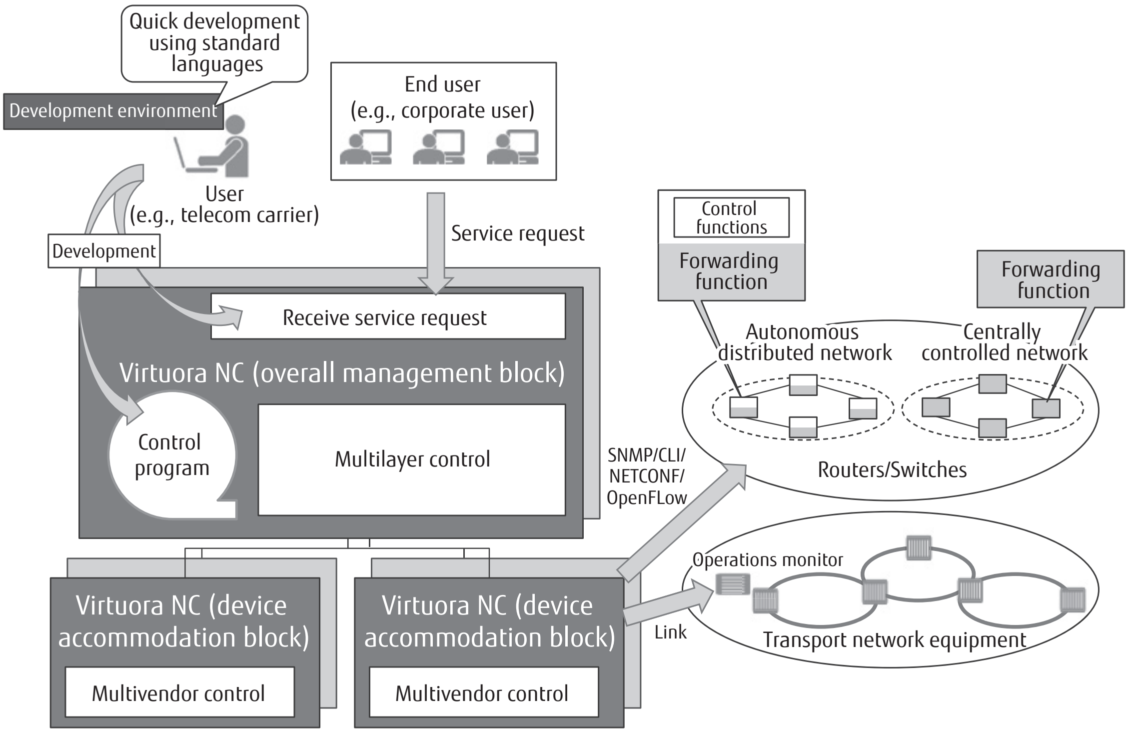

Figure 2
Virtuora NC solution.

layer and establishes logical circuits on top of the transport network by linking with the operations monitor of transport network equipment. Furthermore, as the target of control is hardware of other companies, Virtuora NC selects and applies a control protocol appropriate for those devices and adjusts the settings to enable operation. Provision of the above functions resolves the issue of controlling a composite multilayer network.

2) Quick creation of monetizing opportunities

Virtuora NC provides a development environment using standard languages that enables quick response to network-related service requests from end users. Using this development environment, the telecom carrier itself can define an interface for receiving service requests from end users and can develop a control program for the hardware associated with a request. These capabilities make for early provision of services without having to interface with vendors or other parties. The provision of such a development environment resolves the issue of how to create monetizing opportunities quickly.

3) Ensuring carrier-grade quality

Virtuora NC uses a redundant and hierarchical architecture that enhances reliability. A configuration consisting of active and standby systems supports business continuity by enabling switching to the standby system in the event of a failure in the active system. Additionally, the use of a hierarchical structure consisting of functional blocks for accommodating network devices and a functional block for overall management means that an increase in the number of accommodated devices can be handled by simply adding more functional blocks. This architecture can cope with system failures and upgrades and thus provide stable communications and ensure carrier-grade quality. However, processing in a high-load control plane (control signals) is still an outstanding problem, and we present Fujitsu's approach to solving this problem in the next section.

4) Low-cost, smooth migration to SDN network

Virtuora NC can accommodate both centrally controlled networks and autonomous distributed networks. For autonomous distributed networks, it uses SNMP, CLI, and NETCONF to manage the packet layer, and it links with the existing operations monitor of transport network equipment to manage the transport layer. For centrally controlled networks, it uses the OpenFlow protocol to control operations. In this way, both types of networks can be uniformly accommodated by a single Virtuora NC implementation. This not only enables a stepwise and smooth migration to a centrally controlled network but also reduces operating costs after SDN deployment.

5) Provision of open platform

The development environment described under "Quick creation of monetizing opportunities" above supports an open platform through the use of standard languages. This provision of an open platform is described later in the section "Open source activities."

4. Distributed network control technology

A key feature of SDN is centralized network control. Consequently, when migrating to SDN from a WAN consisting of switches for forwarding large volumes of packets, the load generated by an increase in the number of users will concentrate at the SDN controller, preventing the smooth provision of services. Another problem is that a fault in the controller itself can prevent managed switches from being controlled.

Given this background, the authors undertook the research and development of cluster-based distributed network control technology that can logically operate multiple control modules as a single controller and centrally manage an SDN network on a scale of 1,000 switches.[6),7)]

1) Issues

In contrast to a centralized controller, a cluster-based controller requires that multiple controllers be linked and operate without conflict. However, as processing must continue even if a controller fails, it is difficult to automatically execute a countermeasure if the load concentrates at or a failure occurs in another controller. As a result, a delay will occur in the processing of the switch group managed by a failed controller, preventing continuous control.

2) Cluster-based distributed network control technology

- Architecture

The architecture of our proposed cluster-based distributed network controller is shown in **Figure 3**. It has an application section in which such functions as path computation are performed and a controller section in which network devices are set. Each section is treated as a collection of modules or components that can be independently executed as instances on virtual machines (VMs) or physical servers. This enables instances to be scaled out as the network grows.

When an unknown packet arrives at a switch, it is passed to the controller component managing that switch. The controller component analyzes the packet content and transfers the packet to the appropriate application via the messaging system. The application is then executed. For example, the path computation application transfers path-setting information to each controller component, and each component sets the

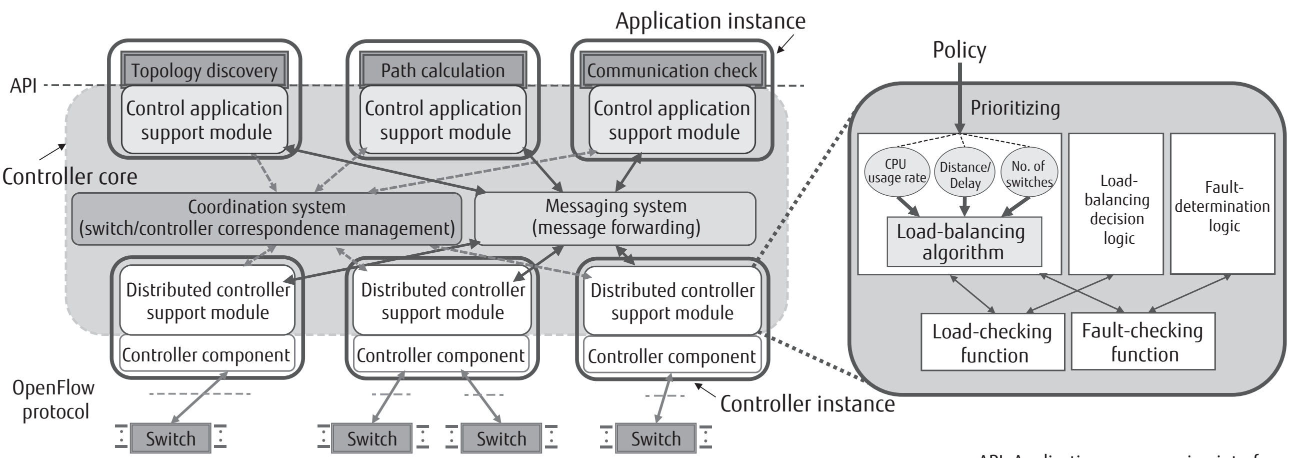

Figure 3
Architecture of cluster-based distributed network controller.

path in the switches it controls by using the OpenFlow protocol.

- Load-balancing technology

We developed a load-checking function that is mounted in the distributed controller support module (right side of Figure 3). This function makes it possible to collect load information (CPU usage rate, number of switches, etc.) for each controller component. For example, a distributed controller support module selected as "leader" by the coordination system on the basis of the number of modules managed or other criteria can detect a load imbalance by periodically checking load information. Then, if the load-balancing decision logic deems load balancing to be necessary, the module decides on the switch or switches to be changed and determines the load balancing method in accordance with a policy specifying the CPU usage rate, number of managed switches, etc. It then registers the relationship between controller components and changed switches with the coordination system. Load balancing is then achieved by reassigning switches in accordance with that updated information.

- Uninterrupted recovery technology

We also developed a fault-checking function that is mounted in the distributed controller support module (right side of Figure 3). Similar to the load-checking function, this function operates in the distributed controller support module selected as "leader." It detects the occurrence of a fault in a controller component and decides on a new controller component to manage the switch or switches connected to the failed controller. Then, on the basis of controller-component load information such as CPU usage rate and number of switches, the function updates the controller/switch correspondence information, resulting in an automatic distribution of load. Operation of the distributed controller support modules that have suffered no failure in coordination with updated information enables the controller components that manage switches to be switched appropriately and service operation to be continued without interruption. Using load-balancing technology in this way to determine how best to switch controller components prevents a dramatic rise in load at a controller component and a halt in processing.

In addition, the coordination system can detect a failure in the leader module itself through a session disconnection. It then selects a new leader module and again determines which controller components manage which switches.

3) Results

Applying these techniques to a cluster-based distributed controller enables the load to be distributed to other controllers in the cluster if the load should suddenly concentrate at a particular controller. It also enables network services to continue without interruption in the event of a failure at a single controller, thereby achieving stable and reliable WAN operation.

4) Outlook for the future

Users demand that an SDN platform be of the open source type, which requires that distributed network control technology be implemented on an open source platform. The following section describes activities in this regard.

Part of the research and development introduced in this section was supported by the Ministry of Internal Affairs and Communications, Japan.

5. Open source activities

The OpenDaylight platform[8] and Open Network OS (ONOS)[9] are attracting attention as open-source SDN platforms.

OpenDaylight is being developed under the Linux Foundation. It supports the YANG[10] model-driven interface, which enables a variety of network devices to be accommodated.

The development of ONOS, meanwhile, is centered on Open Networking Lab (ON.Lab), a research institution whose parent organizations include Stanford University and the University of California, Berkeley. ONOS is oriented to an SDN platform for telecom carriers and features an architecture with high performance, high reliability, and scalability. Proof of concept (PoC) activities to demonstrate the value and feasibility of SDN on ONOS are progressing through the implementation of diverse use cases for telecom carriers.

ONOS can be broadly divided into three elements (**Figure 4**).

1) Distributed control section

This section provides functions for managing the network on a distributed system, which promotes high reliability and scalability.

2) Intent framework

This framework provides an interface between applications and the distributed control section. It

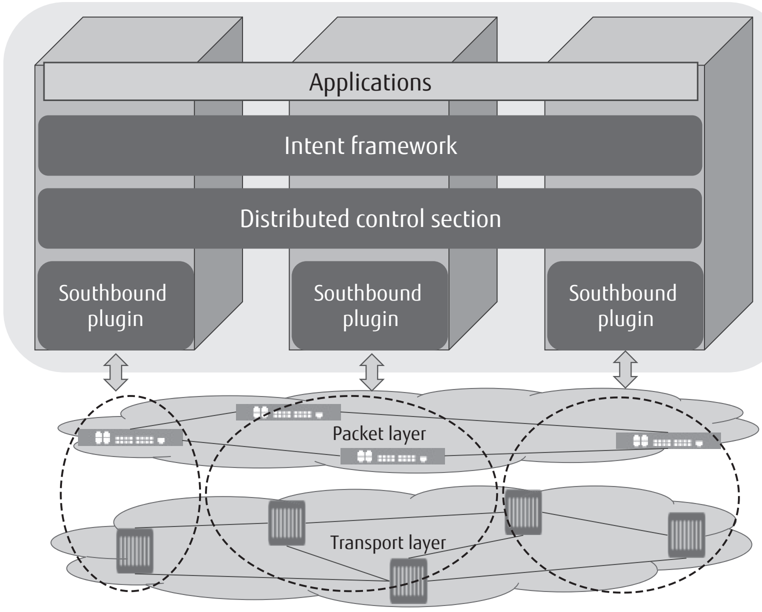

Figure 4
Open Network OS architecture.

features a clear separation between request and behavior. In other words, an application requests the system to perform an action, and the intent framework translates that request into "how" the system should behave to meet that request. Enhancing interface abstraction in this way supports diverse service needs.

3) Southbound plugins

These plugins provide a mechanism for absorbing interface diversity between network equipment and ONOS. They support protocols such as OpenFlow, NETCONF, Path Computation Element Protocol (PCEP), and Transaction Language 1 (TL1).

ONOS also supports a module mechanism based on the Open Services Gateway initiative (OSGi), resulting in an architecture that takes future maintainability and extendibility into account.

In addition, ONOS use case activities are unique in their focus on constructing an SDN ecosystem that promotes a codependent relationship with business enterprises. Participants take on roles based on their respective experiences and viewpoints. ON.Lab takes on a leadership role with respect to the technical aspects of the architecture while telecom carriers define use-case requirements. Vendors also participate by providing source code, performing system evaluations, and making other technical contributions. Contributions from individuals are also welcome in addition to corporate participation.

While all sorts of use cases are now being studied, here we introduce the Packet-Optical use case submitted by Fujitsu as part of its participation in ONOS. The objective of this use case is to provide on-demand bandwidth through multilayer coordination. This is because services can be rolled out in an agile manner appropriate for the cloud era by automatically coordinating the setting of packet-processing devices with the setting of optical transport network equipment and because operations can be made more efficient by unifying the operations that straddle the packet layer and transport layer. This use case was demonstrated at the Open Networking Summit (ONS)[11] held in June 2015.

The construction of SDN systems on WANs using an open source platform has only just begun. As in the past, there is a need for collaboration not only with telecom carriers and vendors but also with the open source community. Going forward, Fujitsu plans to promote collaboration with the open source community and apply SDN to social systems on open source platforms.

6. Conclusion

This paper described the issues in applying SDN to WANs, presented Fujitsu's solution and its control technology for WANs, and introduced Fujitsu's participation in open source activities.

The SDN market for WANs is expected to undergo a full-scale expansion in 2016. Fujitsu plans to expand its product use cases aimed at this market and to respond to the diverse SDN needs of telecom carriers.

References

1) ONF: SDN Architecture issue 1, June 2014.
2) ITU-T: Framework of software-defined networking (Y.3300), June 2014.
3) SNMP: RFC 1157. *https://tools.ietf.org/html/rfc1157/*
4) NETCONF: RFC 4741. *https://tools.ietf.org/html/rfc4741/*
5) OpenFlow. *https://www.opennetworking.org/sdn-resources/openflow/*
6) S. Shimizu et al: Study of Distributed Network Controller and Development of Benchmark Tool for the Controller in SDN, Proceedings of the IEICE General Conference 2013 (in Japanese).
7) K. Hikichi et al.: Study on Scalability Improvement Technique in Distributed SDN Controllers, IEICE Technical Committee on Network Virtualization, Mar. 2015 (in Japanese).
8) OpenDaylight. *http://www.opendaylight.org/*
9) ONOS. *http://onosproject.org/*
10) YANG. *https://wiki.opendaylight.org/view/OpenDaylight_Controller:MD-SAL*
11) ONS. *http://opennetsummit.org/*

Takuya Miyashita
Fujitsu Ltd.
Mr. Miyashita is currently engaged in the planning and development of SDN for wide area networks.

Tomohiro Suzuki
Fujitsu Ltd.
Mr. Suzuki is currently engaged in the planning and development of SDN for wide area networks.

Toshio Soumiya
Fujitsu Laboratories Ltd.
Mr. Soumiya is currently engaged in the research and development of virtual network control technology.

Akiko Yamada
Fujitsu Laboratories Ltd.
Ms. Yamada is currently engaged in the research and development of distributed network control technology.