Research and Development on Network Virtualization Technologies in Japan

Akihiro NAKAO

Since 2008, we have conducted our research on continuously evolve-able network virtualization infrastructure, proposing the concept of “slice”, i.e., a set of isolated programmable resources so as to implement new generation network protocols and services. We have proposed unification of recently called SDN and NFV technologies and the concept of deep programmability to extend programmability deeply into data plane, in collaboration among 6 institutions till 2010 and 7 institutions since 2011. This article gives briefly overview of these research activities and discusses our directions in near future.

1 Introduction

The Internet currently operating on TCP/IP network protocols has its origin in ARPANET developed in the late 1960s. Although the Internet started as an experimental platform for studying inter-networking of various network protocols, after its commercialization in 1990, it has improved and been enhanced to fulfill a variety of requirements as a social infrastructure. At present, it is a critical information social infrastructure indispensable to society.

The Internetworking Principle proposed in 1974 by Cerf and Kahn has existed for more than 40 years as the core of the Internet architecture. It is simply remarkable. However, since around 2000, the Internet has been widely acknowledged to have become “ossified,” i.e., it has become very hard to introduce or add new functions to the Internet; ”Successful and widely adopted technologies are subject to ossification, which makes it hard to introduce new capabilities,” so cited a U.S. National Research Council report[1]. In this way, the view that the Internet is inherently making evolution difficult has been acknowledged as a problem.

To address this problem, researchers around the world have therefore initiated research projects since around 2007, taking approaches such as reviewing the Internet architecture to redesign it with a clean slate. In other words, erase the blackboard and design a brand-new network as a “Clean Slate” network. In the U.S., the Global Environment for Networking (GENI) Project[2] was initiated, funded by the National Science Foundation, to construct a testbed for studies on new network architectures. Afterwards, under the coordination of BBN Technologies, which assumed the role of the GENI Project Office (GPO), studies on large-scale network-virtualization started across the U.S. In 2008 in Japan, the National Institute of Information and Communications Technology (NICT) assumed a similar role of the GENI Project Japan Office and started promoting R&D projects aiming at the actualization of a new generation network. NICT’s actions triggered the R&D of network-virtualization technologies critical to the verification of a variety of protocols for the foundation of new generation networks.

In Japan, the author of this paper Dr. Akihiro NAKAO, a member of the GENI Community with experience working under Prof. Larry Peterson for the PlanetLab Project[3] until 2005 that had a significant impact on the initial design of GENI, served as a project leader in NICT to launch a Virtualization Node (VNode) project. In this project, NICT, the University of Tokyo, NTT, NEC, Hitachi, and Fujitsu collaborated to implement a platform based on VNodes from 2008 to 2010 at a faster pace than GENI. Then, in 2011, NICT, for the purpose of continuing research, contracted R&D to the six research institutes of the University of Tokyo, NTT, KDDI, NEC, Hitachi and Fujitsu; since then, NICT and the six contractors have been collaborating with a so-called “All Japan” formation, to promote the research and development and work as a counterpart in Japan of GENI.

The joint or contracted research projects conducted from 2008 to 2015 actualized a number of proposals that included innovative ideas ahead of GENI. We believe that some of these ideas had a significant impact on the GENI community. The most prominent among them is the

VNode[4] architecture, which is a building component for a network-virtualization platform. In 2008, the VNode architecture design was proposed. In 2010 after the first implementation, it was introduced in a keynote presentation at the 8th GENI Engineering Conference (GEC)[5]. One year later, GENI proposed the GENI Rack architecture, which is similar to the VNode architecture.

Briefly, the VNode architecture consists of two parts mutually connected by a backplane: Redirector, a virtual network component, and Programmer, a network function implementer. The most innovative part of this architecture is that virtual network-connectivity implementation technologies and network-function implementation technologies can be scalably designed and implemented, and they are not constrained by each other. The VNode concept, originally presented in 2008, is the core of the architecture that ensures the creation of network functions through slicing a network. Some network operators have already employed this concept where Software Defined Networking (SDN), Network Function Virtualization (NFV) or their combination (orchestration) is discussed.

Another significant technology that is leading GENI even today is the technology for ensuring the programmability of the data plane. Since the start of our R&D activities, we have focused on the actualization of the next generation network. So, we have implemented a communication platform that ensures the configuration of the data plane where protocols independent of the conventional network protocols, the so-called non-IP protocols, or more precisely, protocols for layer L2 or higher, can be processed flexibly by means of programs. On the other hand, the network virtualization technologies developed in GENI project still remain at the stage where the handling of TCP/IP or Ethernet protocols are allowed, partly because they had to depend on OpenFlow[6]. Therefore, in regard to the concept of a "clean slate," although it was proposed by Stanford University and GENI, it would not be an exaggeration to say that the network virtualization technologies have been developed in Japan in advanced manner because our technologies aim at protocol modification in L2 or higher, although our R&D has been greatly influenced and encouraged by that of GENI project in the U.S.

Our concept, compared to the closest idea of Protocol Oblivious Forwarding (POF)[7] which is used in SDN today, deals with deeper programmability and ensures easier programming because POF deals with limited network functions and uses the pattern matching/action programming model. We proposed "Deeply Programmable Networking (DPN)" which has deeper programmability and ensures easier programming. reference We believe that with the application of DPN a node that can handle independent conventional TCP/IP protocols as Content Centric Networking or Information Centric Networking[8] will prove its true value due to the data-plane programmability enabled through DPN. In addition to DPN, we have an excessive number of technologies to enumerate well ahead of GENI that have made a significant impact such as the method for applying different network-virtualization technologies to fulfill different requirements for edge or core networks, the virtualization of wireless networks, the virtualization of terminals, and creating slices in an application-by-application manner.

It is worthy to note here that the projects we have promoted have had a great impact on the U.S. network-virtualization research community. Network-virtualization research conducted in Japan such as in the Architecture Journal of GENI[2] has attracted a great deal of attention and received very high evaluations.

The rest of this paper is organized as follows. Section **2** introduces the overview and the brief history of network virtualization research world-wide. Section **3** highlights deep programmability, the focus of the research and development of network virtualization technologies in Japan. Finally, Section **4** briefly concludes and addresses remarks for future.

2 Overview of network-virtualization research

In early 2000, the first-generation platforms for verifying and evaluating new network protocols were established. They were overlay systems based on IP networking, on which an experimental environment for virtual networks was implemented. A representative example is PlanetLab[3] from Princeton University. It enabled dynamic configuration into seamless slices of virtualized network-computing node resources on an overlay network. In those days, a slice was defined as a unit component with allocated resources such as computing power/storage on servers or resources existing in namespaces. Since then, a variety of network research activities have been conducted on the system involving more than 700 nodes deployed worldwide in 2000 and more than 1,000 nodes currently. There was another important achievement called Super Charging PlanetLab that enhances the performance and programmability of PlanetLab nodes using network processors[9]. In Japan,

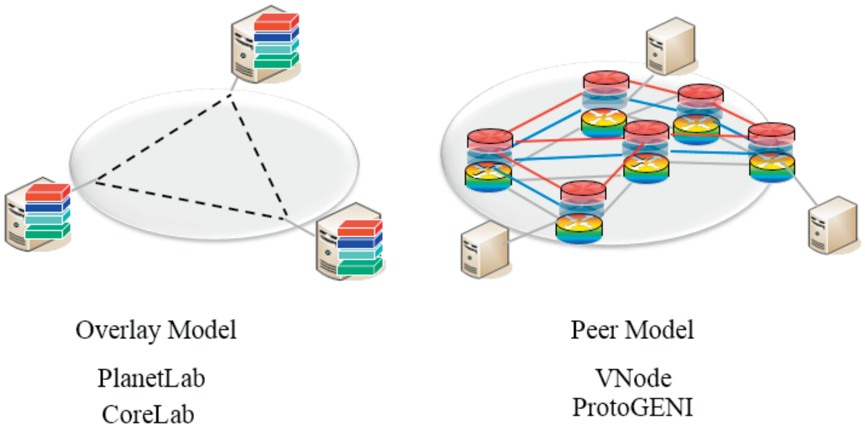

Fig. 1 Overlay model and peer model

CoreLab was developed by the University of Tokyo and NICT based on PlanetLab technologies. It was implemented on the JGN-X network in 2007[10].

However, such overlay platforms have limitations in underlay network controls and are not appropriate for our research goal, i.e., the R&D of a clean slate network design. Hence, in 2008, NICT and the University of Tokyo as well as industry partners initiated an R&D project on network virtualization technologies as an industry-academia-government collaborative project as shown in Fig. 1.

In the project, two critical requirements are identified: *programmability* and *resource isolation*. For programmability, while conventional virtual networks implement a slice by simply multiplexing L2/L3 networks using VLAN tags or tunnels, a slice should be defined as an aggregation of independent programmable resources such as networks, computing power, storage or namespace, and such programmability ensures the handling of any type of protocol. For resource isolation, each resource used to provide such a program-execution environment should be independently allocatable.

One of the design principles in the network-virtualization node was to ensure sustainable evolvability. Following this principle, node development was conducted separately in two parts[11]: As shown in Fig. 2, Redirector is in charge of the virtualization of network links , and Programmer is in charge of the virtualization of node functions. Redirector configures virtual networks while maintaining inter-node links and bandwidth, which is equivalent to the current SDN technologies. The Programmer performs functions, equivalent to NFV, for converting packets, filtering packets, or controlling communication paths or caching/signal processing. It can be said to integrate/enhance the functions of SDN and NFV because it can define new protocols and program their conversion/control. What has been described so far indicates that, as early as in the 2008 development phase of the joint research project, we had proposed an architecture equivalent to that of SDN/NFV. Furthermore, our architecture, by employing virtualization technologies that enable the placement of multiple slices on physical resources, ensures multiplexing of more than one network function, and can adapt to the evolution of technologies expected in the near future.

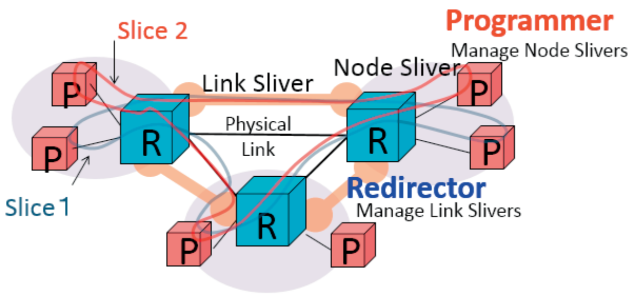

Fig. 2 Basic configuration for network-virtualization (extracted and edited from a paper published by the author in 2008)

At the same time, while our development was on-going, in the U.S., the GENI project in which a total of around thirty top U.S. universities and research institutes participated, pushed forward the development of a testbed based on network-virtualization technologies for promoting R&D of a Clean Slate Network[2]. In Europe, the development of a similar testbed was promoted under the FP7 project[12]. GENI's research achievements were presented at a GEC meeting that is held three times a year. We, participating from the very beginning, gave presentations on the status of the network-virtualization technology development in Japan, as one of the achievements of the joint research on virtualization nodes that started in 2008 at NICT. In 2011, we implemented a prototype network-virtualization node,

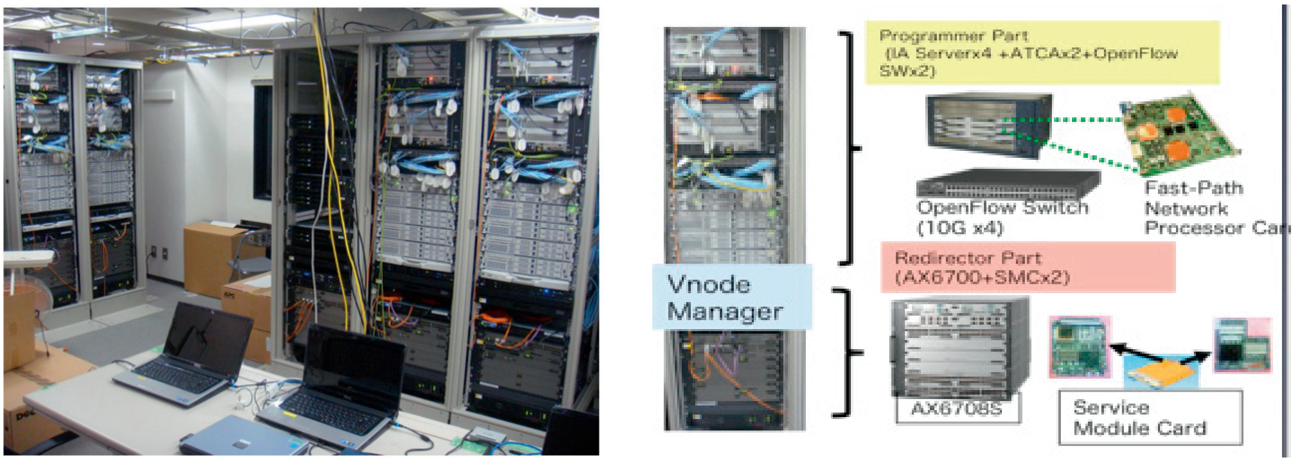

Fig. 3 Prototype implementation of network virtualization node in 2010: Initial version with 4 units

VNode, at NICT's experimental facility in Hakusan to promote the development of the testbed based on network-virtualization technologies as shown in Fig. 3 ahead of GENI Rack developed by the GENI project. GENI Rack has two variations. One is the InstaGENI Rack, which can be said to be an evolved type of PlanetLab, with an architecture requiring fewer resources and a large number of distributed nodes. The other is the ExoGENI Rack, which has an architecture that requires more resources but has a small number of distributed nodes and is similar in its architecture to VNode.

In 2011, for the purpose of promoting network-virtualization technology R&D, the "Network Virtualization Platform Technology Research and Development Supporting Next-Generation Network" was started as NICT contracted NTT, KDDI, the University of Tokyo, NEC, Fujitsu, and Hitachi for R&D. This research project had the following themes: (a) network-virtualization platform implementation, (b) development of a system for allocating and integrating services into the slices that are created on the platform; and (c) application development on the platform. Work on theme (c) was conducted concurrently with that on themes (a) and (b). Work on theme (a) took place from 2011 to 2014. During that period, more than 125 scheduled periodic research discussion meetings in total were held at NICT's Hakusan research site. At each of these meetings, detailed and profound discussions were held under the leadership of the University of Tokyo regarding technology and NICT regarding project management.

The status of the network-virtualization technology development was publicly presented at the Next-Generation Symposium held by NICT. In addition, for the purpose of promoting international collaboration, NICT introduced invited participants from research institutes around the world, through panel discussions at the network-virtualization symposium held by the network-virtualization technical committee established by the Institute of Electronics, Information and Communication Engineers (IEICE)[13].

With regard to international relations, participating in the abovementioned GEC every time since 2008, we opened a booth at its Live-Network Demonstration, one of their prime exhibitions, to show the validity of the functions we developed. In addition to the exhibitions, we gave presentations at the plenary sessions four times, GEC 7 in 2009, GEC 10 in 2011, GEC 13 in 2012, and GEC 15 in 2013. The participants were impressed with the degree of evolution and the advancements in our network-virtualization activities. Moreover, in 2014, at the demonstration session, our FLARE technology[14], which will be described later, was awarded "Best Demonstration." Hence, what has been described so far indicates that our activities have been attracting a very high level of interest in the GENI community, and that our technologies have been leading others and making a huge impact on researchers around the world.

On the other hand, OpenFlow, which has become the basis for actualizing SDN today, was introduced in its initial development phase. At GEC, for example, live demonstration sessions were held, and hands-on tutorials were offered to the participants to promote the technology. To catch up with the U.S. in the public relations campaign-race on network-virtualization technologies (we are a little bit behind right now), we have been putting efforts into the promotion of our technologies since FY2013 through a variety of measures, for example, offering tutorials and hands-on sessions at IEICE or Network Virtualization Study Group events.

3 Development of network virtualization technologies supporting deep programmability

OpenFlow, proposed by Stanford University, has already been acknowledged worldwide as the most popular SDN technology. Furthermore, industry has prematurely decided to employ it in the business field. It is an epoch-making architecture where the data plane and control plane that conventional network devices have both are separated and the liberated control plane is controlled by programs written in computer programming languages. However, because data plane in the OpenFlow architecture is rigidly contained in hardware, and not allowed to handle packets other than IP packets, deeper programmability especially in data plane was to be pursued.

On the other hand, being aware since as early as at the start of the project that the data plane should have such programmability as the control plane, we designed a virtualization node such that is able to execute through programs the processing of frames on L2 or higher. We refer to these frames as Any Frames. In other words, we proposed DPN (Deeply Programmable Networking) where even the data plane has an SDN feature as shown in Fig. 4. However, it is not easy to attain high-speed performance with a data-plane programmable node. This is the challenge for DPN. To address this problem, we employed the following configuration approach for VNode design: SlowPath programming environment that consists of Intel Architecture (IA) servers, and FastPath comprising network processors that perform network processing.

Programming of the network processors used for the VNode's FastPath is relatively difficult because the processors must execute high-speed processing in kernel mode and such difficulty in programming may be problematic. However, due to the advancements in IA processor technology and the availability of technologies for network interface controller (NIC)-offloading packet processing, upgrading VNode to increase the processing speed has become realistic. These activities however have been suspended because the study phase and the R&D contracts on the virtualization node came to a close at the end of FY2014. A joint research project was started in FY2015, and the University of Tokyo and NTT are developing their own virtualization nodes. Of course, an increase in speed to some extent should be achieved by using the Data Plane Development Kit (DPDK)[15] provided as an open source solution by Intel Corp. and other similar tools; however, such tools are applicable to a limited number of cases. An example of this is DPDK because a specific type of conventional NIC (physical layer (PHY)) must be used and DPDK cannot handle frames other than conventional Ethernet frames.

FLARE Node[14], derived from VNode, ensures simple realization of DPN. We developed it in conjunction with a research project on edge network virtualization in VNode project as described in the next chapter of this special issue.

The FLARE Node is configured with SlowPath and FastPath. The network processor for FastPath employs an architecture enabling use from the user space of the Polling Mode Driver (PMD), which ensures easy programming. FLARE provides easy programmability and ensures high-scalability and high throughput by employing many-core-processor multiprocessing and recently using Intel DPDK and other technologies as shown in Fig. 5. Hence, in FLARE, use of such a DPN platform that can program the data plane and easily achieve expandability of the data-plane components and their API (Southbound Interface) is

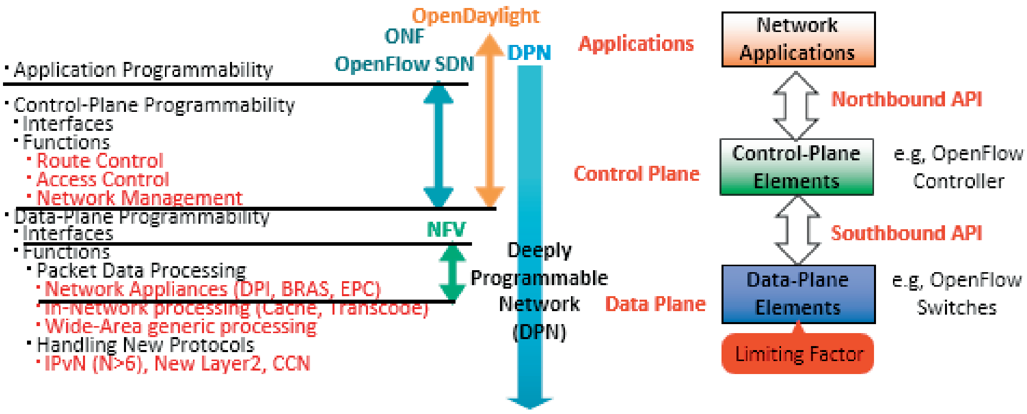

Fig. 4 SDN and DPN

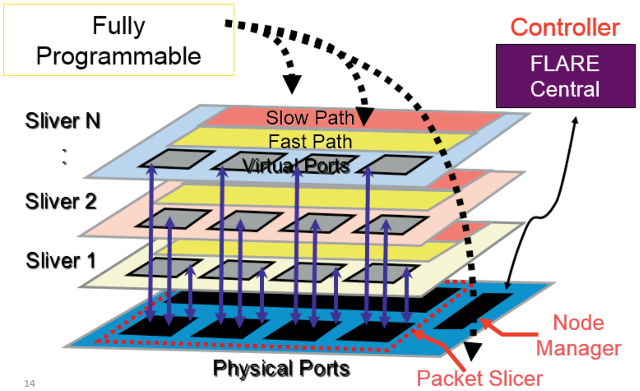

Fig. 5 Small-scale network virtualization node using many-core processors

ensured. This means, for example, that, in addition to the byte string defined in the packet header, software can be used to control the byte string in a user specified field. The closest idea to ours mentioned above is Protocol Oblivious Forwarding (POF)[7]. However, POF has limitations. Its network function is limited, and it employs the pattern-match-and-action program model that is used in OpenFlow. On the other hand, FLARE supports deeper programmability than that provided by POF, and is much more easily programmable. Furthermore, POF or OpenFlow can be implemented on FLARE as a network function element.

While VNode uses high-speed Generic Routing Encapsulation tunneling for processing arbitrary types of frames (so called any frames or arbitrary frames), FLARE is able to process any frames without tunneling because the PHY is designed to handle any frames. This means that an edge network is not required to prepare tunnels to implement new protocols or services, and it leads to reduction of overhead in achieving high network performance. Therefore, FLARE is expected to be used in many cases, particularly in edge networks, to serve Internet of Things (IoT)/Machine-to-Machine (M2M) applications.

In summary, FLARE employs a node architecture that features deep programmability (Deeply Programmable Node Architecture). FLARE can be implemented so that many types of processors such as general purpose processors, many-core-processors or GPGPUs (general purpose graphics processor units) are integrated. As for the present version of FLARE, there are two types of implementations: (1) an integration of EZchip Tile processors, Intel general purpose servers, and GPGPUs, and (2) Intel general purpose servers by Intel DPDK. We plan to add implementations of other types in near future. In addition, in the current situation where different types of traffic from IoT, sensors, and mobile to high-precision images are exchanged, we note the usefulness of operating network functions in application-specific ways.

Regarding application communications, network devices in general, even an advanced type of SDN switch, have no way to access directly the information existing in the application layer except by conducting DPI (deep packet inspection). This is because, conventionally on the Internet, the application context is lost at the socket interface at the time when session/datagram abstraction is performed. So, on the Internet, application-specific controls are not applicable, only universal control is employed.

To enable application controls in the network layer, an architecture is required that ensures the definition of an additional field in a packet for application specific in-network control. We proposed defining a trailer that can add application information at the end of a packet instead of defining an additional header. This method, if combined with SDN-type network control, ensures packet-traffic control through application identification, which is not achievable using the conventional network equipment[16].

We implemented this method in a mobile network environment and held a demonstration at the GEC20 Evening Demo Session. The demonstration was highly praised, and we were awarded the “Best Demo Award.”[17]

4 Perspectives for the future

Since 2008, network-virtualization technologies have been developed as fundamental technologies supporting new generation networks. At the initial stage of development, these technologies were acknowledged primarily as tools to realize a clean slate experimental environment, and the method of subdividing a network into slices to build a variety of application-specific networks has been developed as a candidate of new generation network infrastructures. We believe that the network virtualization technologies we developed have had a great impact on other research activities, especially, GENI’s R&D in the U.S. because of their innovativeness and uniqueness.

Without doubt, as more and more of the things around us become connected to networks, as proposed in the concept of IoT or Internet of Everything (IoE), networks will not be able to fulfill application requirements if they continue to perform “one-size-fits-all” control on a variety of traffic types with different characteristics. Therefore, we believe that the network virtualization technologies, combined with traffic-classification technologies, will support the core of new generation networks. In addition, technologies for enabling easy programming of high-speed processing are indispensable for reducing development time and cost. At present, for such programming, general servers are used. However, advanced network processor and reconfigurable ASIC technologies that may outperform the conventional processor architectures may become available as another factor contributing to the realization of network-virtualization technologies.

Considering such situations, it is reasonable to predict that the seamless consolidation of networking and computing would advance. We believe that edge networks, at least, as the idea of Mobile Edge Computing (MEC) shows, will play a crucial role and networks in near future will evolve to become "intelligent and functional pipes" where cloud computing, network functional elements at edges and user equipment collaborate naturally, instead of being "dumb pipes" connecting cloud computing and user equipment. Such network evolution will lead to a world where, as a result of seamless integration of networks and computing, computing is available any place on a network. The whole of the Internet will function as a huge distributed computer system.

We believe that our near future R&D targets should include (1) the development of use cases, a.k.a., applications of seamless integration of networks, edges, and computers, especially in the areas of IoT, IoE, M2M, and 5G mobile networks, (2) fundamental technologies to be developed to support the abovementioned use cases such as high-speed data-plane processing with many-core processors and/or general purpose processors, or high-speed SDN switches utilizing reconfigurable ASICs, and (3) advanced traffic classification and slicing technologies, e.g., according to various metrics such as application/device/context, etc. Also, considering the tradeoff between the extent of programmability and cost performance, although software based solutions will surely increase their scope because they reduce capital expense and operational expense and enable flexibility in infrastructure, we have to bear in mind that we have to apply the right technology to the right place.

On the other hand, with regard to the application to commercial networks, business issues are always challenges such as standardization, interoperability, operation and management, security measures, economic efficiency, and energy conservation. Since we believe energy conservation and interoperability, in particular, are critical issues for commercialization, we should continue our activities by participating in international standardization organizations and consortiums.

Acknowledgments

The author expresses his deep appreciation to NICT who promoted the research and development of the Next-Generation Network and gave him much support. The author also sincerely thanks the research organizations that contributed to the joint research projects.

References

1 Looking Over the Fence at Networks: A Neighbor's view of Networking Research, National Academy Press, 2001.

2 Mark Berman, Jeffrey S. Chase, Lawrence H.Landweber, Akihiro Nakao, Max Ott, Dipankar, Raychaudhuri, Robert Ricci, and Ivan Seskar, "GENI:A Federated Testbed for Innovative Network Experiments," Computer Networks 61, pp.5-23, 2014

3 Larry Peterson and Timothy Roscoe, "The design principles of PlanetLab," SIGOPS, Vol. 40, pp.11-16, Jan. 2006.

4 Akihiro Nakao, "VNode: A Deeply Programmable Network Testbed through Network Virtualization," 3rd IEICE Technical Committee on Network Virtualization, March 2012.

5 https://www.geni.net/?p=1739

6 Nick McKeon, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner, "OpenFlow: Enabling Innovation in Campus Networks," ACM SIGCOMM, vol. 38, pp.69-74, April 2008.

7 Haoyu Song, "Protocol-oblivious forwarding: unleash the power of SDN through a future-proof forwarding plane," HotSDN'13, pp.127-132, Aug. 2013

8 Van Jacobson, Diana K. Smetters, James D. Thornton, Michael F. Plass, "Networking Named Content," CoNEXT'09, Dec. 2009.

9 Jonathan S. Turner, Patrick Crowley, John DeHart, Amy Freestone, Brandon Heller, Fred Kuhns, Sailesh Kumar, John Lockwood, Jing Lu, Michael Wilson, Charles Wiseman, and David Zar, "Supercharging planetlab: a high performance, multi-application, overlay network platform", *SIGCOMM Comput. Commun. Rev.* 37, 4 (August 2007), 85-96 , 2007.

10 Akihiro Nakao, Ryota Ozaki, Yuji Nishida, "CoreLab: An Emerging Network Testbed Employing Hosted Virtual Machine Monitor," ACM ROADS 2008 (December 2008).

11 Akihiro Nakao, "VNode: A Deeply Programmable Network Testbed Through Network Virtualization", The 3rd domestic conference, IEICE Technical Committee on Network Virtualization (NV), Mar. 2, 2012.

12 https://ec.europa.eu/research/fp7/index_en.cfm

13 http://www.ieice.org/~nv/

14 Akihiro Nakao, "Flare: Deeply Programmable Network Node Architecture," http://netseminar.stanford.edu/10_18_12.html.

15 http://dpdk.org/

16 Akihiro Nakao, Ping Du, and Takamitsu Iwai,"Application Specific Slicing for MVNO through Software-Defined Data Plane Enhancing SDN", IEICE TRANSACTIONS on Communications, Vol.E98-B, No.11, pp.2111-2120, 2015.

17 http://groups.geni.net/geni/wiki/GEC20Agenda

Akihiro NAKAO, Ph.D.

Professor, Interfaculty Initiative in Information Studies, Graduate School of Interdisciplinary Information Studies, The University of Tokyo
Network, Computer Science