IEICE TRANS. COMMUN., VOL.E93–B, NO.3 MARCH 2010

INVITED LETTER Special Section on New Generation Network towards Innovative Future Society

Network Virtualization as Foundation for Enabling New Network Architectures and Applications

Akihiro NAKAO[†a)], ***Member***

SUMMARY Network virtualization has become a common research topic that many researchers consider a basis for defining a new generation network architectures. In this paper, we attempt to clarify the concept of network virtualization with its brief history, to introduce the benefit of network virtualization for the future network, to posit our strong belief in that the future network should adopt a form of a meta-architecture that accommodates multiple competing multiple architectures, and to identify challenges to achieving this architecture.

key words: *new generation network architecture, network virtualization, overlay networks, testbed*

1. Introduction

Network Virtualization has become one of the buzz words in research communities all over the world, especially since the recent advances in a multitude of server (operating system) virtualization technologies [1]–[6]. One reason why this term has become popular is that we see the benefit from extending the notion of virtualization of computational resources to that of network resources for enabling finer-grain resource control and another is that various technologies such as programmable switches [7]–[9] and powerful multicore network processors [10]–[12] have become all mature enough to enable isolation of both computational and network resources at routers and switches.

We observe that the term network virtualization has not been clearly defined, although it has been mentioned and discussed in many articles and various occasions. To make matter worse, the term is often abused and confused with the existing concepts such as overlay networks and virtual private networks (VPNs). Also, its benefit has often been stressed only as a fundamental technology for constructing testbeds for future Internet architectures.

In the light of these observations, in this paper, we attempt (1) to clarify our definition of the concept of network virtualization, summarizing its history, (2) to introduce the benefit of network virtualization and to posit our strong belief that the future network architecture be a form of a meta architecture that accommodates multiple concurrent architectures, and (3) to identify challenges to achieving a planetary scale network-virtualized meta-architecture. We briefly conclude in the end.

Manuscript received October 13, 2009.
Manuscript revised November 21, 2009.
[†]The author is with The University of Tokyo, Tokyo, 113-0033 Japan.
a) E-mail: nakao@iii.u-tokyo.ac.jp
DOI: 10.1587/transcom.E93.B.454

2. Definition of Network Virtualization

In our view, we define network virtualization as *a technique for isolating computational and network resources through virtualization to allocate them to a logical(virtual) network for accommodating multiple independent and programmable virtual networks*[*].

There are several differences between a traditional concept of VPNs and network virtualization. While VPNs only offer apparent and dedicated connectivity in the current network architecture, network virtualization aims to achieve the additional features: (1) *programmability:* a virtual network may be equipped with programmable control plane, (2) *topology awareness:* a virtual network may be topology-aware rather than offering only connectivity, (3) *quick reconfigurability:* a virtual network may be quickly provisioned and reconfigured, (4) *resource isolation:* a virtual network may be allocated a set of computational and network resources, and (5) *network abstraction:* a virtual network may accommodate a new architecture different from the current Internet architecture. As shown in Sect. 4, there are also fundamental differences between so called overlay network and network virtualization.

In our view of network virtualization, we define a loose network abstraction model (Fig. 1) in order to support non-IP network architectures. In this model, a *slice* is defined as an isolated set of computational and network resources allocated and deployed across the entire network. A slice consists of primitives such as a *link sliver* that conveys traffic, a *node sliver* that forks traffic with equipped programmability, and an *interface* that connects a link sliver and a node sliver.

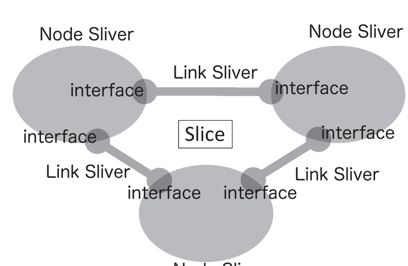

Fig. 1 Network abstraction model in network virtualization.

[*]We can always add isolation of the other kinds of resources such as storage, but for the sake of simplicity, we intentionally do not include them.

Slivers must be isolated from each other, so that there exists no cross-talk between them.

In general, data transmission between end systems can be modeled as a series of forwarding decisions at node slivers and transmissions over link slivers. No matter what format of data is transmitted and conveyed on a slice, a node sliver can be programmed to parse, route and forward the data through a link sliver to another node sliver. In other words, the abstraction model allows us to define an arbitrary data format, whether to transmit data, e.g., via circuits or packets, how to route data, etc., within a slice. This nuts-and-bolts view of network virtualization shows that a slice may define an arbitrary network architecture that consists of protocols, i.e., message formats and rules for actions, and the organization of the protocols, e.g., a traditional layered model and the other types of architectures [13]–[15].

A final note in this section is that although the term includes virtualization, our primary goal is to "isolate" resources for an individual logical network using virtualization techniques as a means, thus, virtualization itself is not necessarily our goal.

3. Benefit of Network Virtualization

We posit that, according to our definition of network virtualization, we enjoy the benefit of implementing *multiple* network architectures and services in isolated logical networks on top of a *single* shared physical infrastructure. The benefit serves two purposes: first, in the long run, we can define a *meta-architecture* to accommodate multiple architectures concurrently, and second, in the short term, we can construct testbeds to experiment with multiple disruptive network architectures and services concurrently without interference among those experiments.

One may wonder if we really need such a meta-architecture to concurrently hold multiple network architectures. However, the recent trend shows that many parties are defining their own new network architectures departing from the current Internet. For example, academic research activities reveal that due to the ossification of the current Internet, we must exercise *clean-slate thinking* to design new network architectures and services [16]. Also, in industries, several enterprises such as Google, Amazon, and Yahoo now own their networks to connect large data centers internally and to and from the Internet backbones for efficiently delivering contents and software services. The fear is that these elephants may define their own future networks before academia defines and agrees upon the future Internet.

We also posit that not only in the context of future network architectures and testbeds for defining them, but also in the current Internet, carriers and operators may benefit from network virtualization by operating multiple existing architectures concurrently. The point is that carriers and operators may be able to offer a virtual network with isolated resources per user and even per application, thus may open the door to new business models. We have already seen such business models in a limited form, e.g., MVNO, but network virtualization accelerates the isolation of resources and provides user- and application-specific virtual networks.

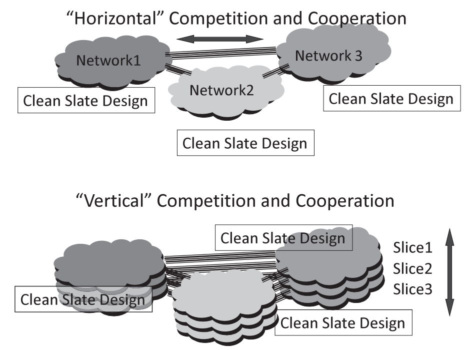

Fig. 2 Horizontal vs. Vertical coopetitions.

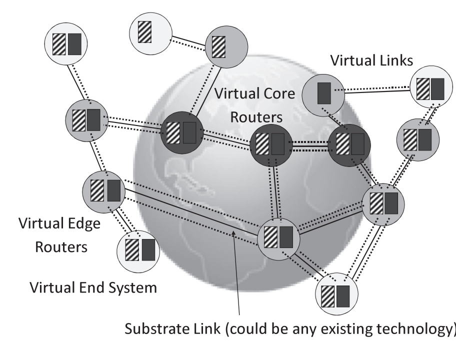

Fig. 3 Planetary scale network virtualization (End systems (white circles), edge routers (light gray circles) and core routers (dark gray circles) are all virtualized so computational resources are isolated into different slices (solid and striped boxes). Substrate links are virtualized so network resources are also isolated.).

We expect in such a meta-architecture that we would introduce a competition principle between multiple clean-slate designs. Through network virtualization, each network could implement its own clean-slate design and compete and cooperate (*coopetition*) with another network of different design. However, with network virtualization, we isolate the resources of the entire network of networks into *slices* and implement a clean-slate design per slice, thus create *vertical coopetitions* among multiple architectures (Fig. 2). In addition, the vertical coopetition allows us to switch from one (production) architecture implemented in a slice (the solid slice in Fig. 3) to another experimental one (the striped slice in Fig. 3), thus, achieves a continuously evolvable network architecture.

4. A Brief History of Network Virtualization

In this section, we introduce a brief history of network virtualization to understand its background.

Overlay network is probably the most recent ancestor of network virtualization in terms of the purposes of resolving the ossification issue of the current Internet and of defining innovative network architectures and services. While most overlay networks are distributed applications, a few examples such as RON [17] and Scalable Routing Overlay [18], [19] have proposed mechanisms for improving the current routing architecture in the Internet. In this context, PlanetLab [20], [21] has played a very important role in enabling such overlay networks to experiment with new architectures. PlanetLab has also first introduced the concept of *slice* mentioned in Sect. 2.

Within the PlanetLab community, the concept of "Internet In A Slice (IIAS)" has stemmed from the routing overlay, where we could implement our own internets, but limited to enabling experimental routing at that time. The IIAS concept has been implemented as VINI [22] later. A fundamental difference between overlay network and network virtualization is that overlay networks are realized through *virtualization of computational resources at the network edges* and also is *unable to achieve rigorous isolation and provision of network resources* between computational resources, while network virtualization aims at *isolating computational resources inside networks (e.g., routers and switches) as well as network resources between them.* In a sense, IIAS (and routing overlays) drives the concept of network virtualization since it is hard to benefit from IIAS without isolation through network virtualization.

There exist several research efforts advocating that network virtualization be the foundation for defining and enabling the future network that accommodates multiple concurrent architectures, such as Cabo [23], Diversified Internet [24], and NVLAB and AKARI [25], [26]. Also, GENI [16], SFA [27], [28] and several other testbed efforts in Europe and Japan inherit the concept of slice to isolate resources for experimenting with disruptive network architectures simultaneously.

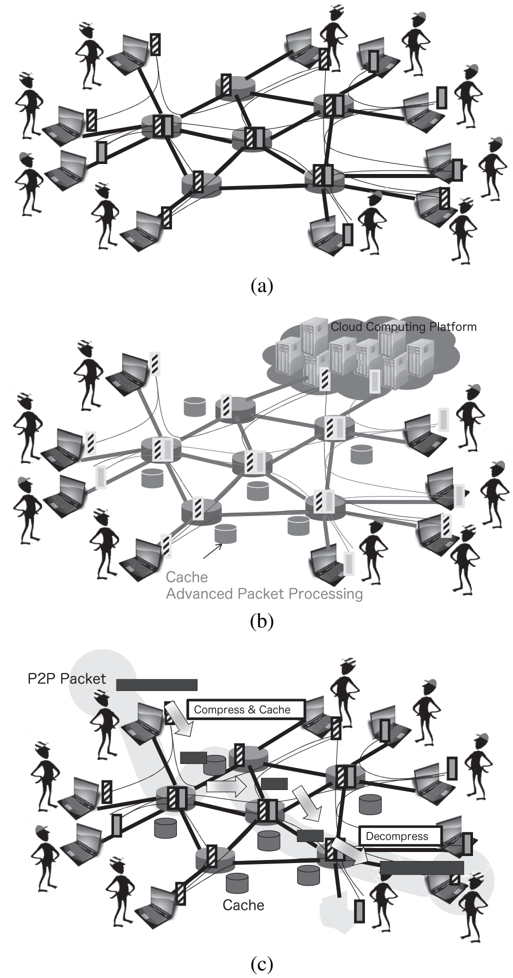

Fig. 4 Applications of network virtualization.

5. Challenges

In this section, we identify seven challenges in realizing a meta-architecture through network virtualization that supports multiple concurrent architectures.

Isolation: Isolation of resources is one of the most important challenges. Security isolation ensures there is no cross-talk between slices of resources and performance isolation defines strict QoS guarantee of a given slice. Most operating system virtualization achieves both isolation mechanisms, but extending such isolation concepts to network resources is a challenge.

Performance: Performance issues are also an important challenge. The most server virtualization techniques of today may perform sufficiently for the control plane such as signaling and routing protocols, but they fall short for the data plane that requires fast network I/O for a slice. We expect further advance in virtualization in network processors [10]–[12] and FPGA [8] to enable advanced per-packet processing in a slice.

Scalability: It is frequently wondered how many slices, i.e., how many different architectures we must support in our meta-architecture. PlanetLab reports that there are over a thousand slices where researchers conduct their experiments, so this gives a ballpark figure to support in our meta-architecture. OpenFlow [7] is arguably one of the promising network substrates for multiplexing slices within an enterprise network, but an interesting question is how it scales with its centralized control architecture. Also, another interesting question is how to resolve conflicts in FlowSpace [7].

Flexibility: In our meta-architecture, we must be able to flexibly implement experimental architectures in network nodes such as routers and switches. This problem is tightly coupled with operating system virtualization. Resource containers[4], [5] sacrifices flexibility over performance and scalability, while Hosted and Hypervisor approaches [1]–[3], [6] leave room for performance improvement [29]. Further advance in virtualization technologies is expected.

Evolvability: A network virtualization mechanism must keep up with the progress in substrate technologies such as optical and wireless networks. We must define a narrow waist virtualization layer that will not become unusable with the latest substrate technologies.

Management: Network management issue is another hard challenge to resolve, when we have thousands of virtual networks to control. We must devise a viable method to scale management in the meta-architecture. Note, however, that supporting at least two slices would allow us to define a production network architecture and an experimental one running concurrently and to get around the ossification issue we are facing today.

Applications: What kind of architecture should be implemented in a slice is a whole different and important challenge. Research communities are putting forth clean slate thinking, but the first things to try out may be those architectures that propose architecturally incremental improvements from the current Internet, but require changes in routers and switches. We are striving to discover such applications of network virtualization [26]. For example, Fig. 4(a) shows that a network operator may provide a sliced network as a service (NaaS). People with solid caps will be allocated to the solid slice of resources, and those with striped caps will be assigned to the striped slice. Another application in Fig. 4(b) combines the idea of NaaS with cloud computing where a cloud service provider can allocate not only computing resources but also a slice that offers dedicated access from a group of users, possibly with advanced packet processing like caching within the slice. Finally, Fig. 4(c) shows that we can confine P2P traffic in a slice and perform packet processing such as (de)compression and caching [30].

6. Conclusion

In this paper, we attempt to clarify the concept of network virtualization with its brief history, introduce the benefit of network virtualization for the future network, posit our strong belief in that the future network should adopt a form of a meta-architecture that accommodates competing multiple architectures, and identify challenges to achieve this architecture. Interesting topics await us in the field of the research for defining our future network.

References

[1] "VMWare," http://www.vmware.com/
[2] "Xen," http://www.xen.org/
[3] P. Barham, B. Dragovic, K. Fraser, S. Hand, T. Harris, A. Ho, R. Neugebauer, I. Pratt, and A. Warfield, "Xen and the art of virtualization," Proc. 19th ACM Symposium on Operating Systems Principles (SOSP), pp.164–177, New York, NY, USA, 2003.
[4] "Linux VServer," http://www.linux-vserver.org/
[5] "Open VZ," http://wiki.openvz.org/Main_Page
[6] "Kvm," http://kvm.qumranet.com/kvmwiki
[7] "OpenFlow Switch," http://www.openflowswitch.org/
[8] "NetFPGA," http://www.netfpga.org/
[9] J.S. Turner, P. Crowley, J.D. DeHart, A. Freestone, B. Heller, F. Kuhns, S. Kumar, J.W. Lockwood, J. Lu, M. Wilson, C. Wiseman, and D. Zar, "Supercharging planetlab: A high performance, multi-application, overlay network platform," Proc. ACM SIGCOMM, pp.85–96, 2007.
[10] "Netronome Systems," http://www.netronome.com/
[11] "Cavium Networks," http://www.cavium.com
[12] "RMI Cooperation," http://www.rmicorp.com
[13] J.D. Touch, Y.S. Wang, and V. Pingali, "A recursive network architecture," ISI Technical Report ISI-TR-2006, 2006.
[14] V. Jacobson, D. Smetters, J. Thornton, M. Plass, N. Briggs, and R. Braynard, "Networking named content," Proc. ACM CoNEXT, pp.1–12, 2009. http://www.parc.com/content/attachments/networking-named-content-preprint2.pdf
[15] T. Koponen, M. Chawla, B.G. Chun, A. Ermolinskiy, K.H. Kim, S. Shenker, and I. Stoica, "A data-oriented (and beyond) network architecture," Proc. ACM SIGCOMM, pp.181–192, New York, NY, USA, 2007.
[16] T. Anderson, L. Peterson, S. Shenker, and J. Turner, "Overcoming the Internet impasse through virtualization," Tech. Rep. GENI Design Document GDD-05-01, GENI Planning Group, April 2005.
[17] D. Andersen, H. Balakrishnam, F. Kaashoek, and R. Morris, "Resilient overlay networks," Proc. 18th ACM Symposium on Operating Systems Principles (SOSP), pp.131–145, Oct. 2001.
[18] A. Nakao, L. Peterson, and A. Bavier, "A routing underlay for overlay networks," Proc. ACM SIGCOMM, pp.11–18, Aug. 2003.
[19] A. Nakao, L. Peterson, and A. Bavier, "Scalable routing overlay networks," Proc. ACM SIGOPS, vol.40, no.1, pp.49–61, 2006.
[20] L. Peterson, T. Anderson, D. Culler, and T. Roscoe, "A blueprint for introducing disruptive technology into the Internet," ACM SIGCOMM Computer Communication Review, vol.33, no.1, pp.59–64, 2003.
[21] L. Peterson, S. Muir, T. Roscoe, and A. Klingaman, "PlanetLab architecture: An overview," Technical Report PDN–06–031, PlanetLab Consortium, May 2006.
[22] A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford, "In VINI veritas: Realistic and controlled network experimentation," Proc. ACM SIGCOMM, pp.3–14, New York, NY, USA, 2006.
[23] N. Feamster, L. Gao, and J. Rexford, "How to lease the Internet in your spare time," ACM SIGCOMM Computer Communications Review, pp.61–64, Jan. 2007.
[24] J. Turner, P. Crowley, and S. Gorinsky, et al., "An architecture for a diversified Internet," NSF NeTS FIND Initiative, http://www.nets-find.net/Funded/DiversifiedInternet.php
[25] "Akari project," http://akari-project.nict.go.jp/eng/index2.htm
[26] "Network Virtualization Lab," http://www.nvlab.org
[27] L. Peterson, S. Sevinc, J. Lepreau, R. Ricci, J. Wroclawski, T. Faber, S. Schwab, and S. Baker, "Slice-based facility architecture," Tech. Rep., http://svn.planet-lab.org/attachment/wiki/GeniWrapper/sfa.pdf, Feb. 2009.
[28] L. Peterson, S. Sevinc, S. Baker, T. Mack, R. Moran, and F. Ahmed, "PlanetLab implementation of the slice-based facility architecture," Tech. Rep., http://svn.planet-lab.org/wiki/GeniWrapper#geniwrapper, Feb. 2009.
[29] "CoreLab Testbed," http://www.corelab.jp
[30] A. Nakao, K. Sasaki, and S. Yamamoto, "A remedy for network operators against increasing P2P trafc: Enabling packet cache for P2P applications," IEICE Trans. Commun., vol.E91-B, no.12, pp.3810–3820, Dec. 2008.