This paper is a joint publication of the ECAR Campus Cyberinfrastructure (ECAR-CCI) and Communications Infrastructure and Applications (ECAR-CIA) working groups. ECAR working groups bring together higher education IT leaders to address core technology challenges. Individuals at EDUCAUSE member institutions are invited to collaborate on projects that advance emerging technologies important to colleges and universities. More information can be found at the ECAR working groups website.

Introduction

Everywhere you turn, you see an article about software-defined networking or a vendor promoting its SDN strategy, which may lead you to believe that SDN is already a mainstream technology. The reality, however, is that you are not alone if you feel you don’t have a strong enough understanding of SDN. Even though the technology has existed for well over a decade, it’s still considered an emerging conglomeration of technologies that do not yet interoperate in clear ways. Standards are evolving, the effects are not well understood, and the optimal use cases and payoffs are poorly documented. Additionally, most institutions lack a clear and immediate need to deploy SDN at their site. The hype around SDN ranges from its being the silver bullet that solves researcher large-scale data-transfer needs to a technology that finally makes high-performance networking cheap using open-source software and commodity hardware. Reality, as you might expect, differs somewhat from the hype.

With the advent of complex IT services spanning multiple highly virtualized data centers and the trend of moving services to software as a service (SaaS) in the cloud, there is now a greater emphasis on the robustness and flexibility of the network underlying these services. Outsourced e-mail and collaboration services such as Google Apps, Microsoft Office 365, and Blackboard, with video and other third-party building blocks and modules integrated, have moved away from a hub-and-spokes network model to a peer-to-peer approach across the Internet.

SDN may do for networks what virtualization has done for servers. Essentially, splitting the control and data planes can provide applications a level of network abstraction similar to the way hypervisors abstracted computer hardware for virtual machine operating systems.

What Is SDN?

SDN is a new approach to designing, building, and operating networks that allows system administrators and network engineers to respond quickly to ever-changing network requirements, thereby optimizing resources. It does this by approaching computer networking in a way that decouples the bit-forwarding function (“data plane”) from programmability (“control plane”) in order to manage the network services in a simpler way and enable network end users and applications to customize the network to their needs (see
This approach further allows for the injection of new commands centrally to allow for more flexibility and faster responses.

**APPLICATION PLANE**

- App 1: (e.g., Lync)
- App 2: InfoSec Logs
- App 3: Research
- ... App N

**CONTROL PLANE**

- SDN Controller Cluster

**DATA PLANE**

- Network Switch
- Network vSwitch
- Network Router
- ... Network Device

**Northbound OpenFlow API**

**Southbound OpenFlow API**

Figure 1. SDN architecture

From an industry-wide perspective, the most successful SDN implementations are mostly internal to mammoth infrastructure operators (e.g., Google, Amazon, etc.) that are reconfiguring to optimize huge traffic flows within and across their data centers.¹

**SDN Benefits**

What type of benefits will SDN provide the higher education campus at large?

- **Granular Traffic Control**: The separation of the control and data planes allows for easier and much more seamless management of campus networks. Because administrators can change the network policies through a central controller, prioritizing and shaping traffic becomes less labor intensive and less error prone. This is especially helpful for managing traffic loads in a more efficient manner, such as in cloud computing and data centers, but it can also be used for managing the user experience for wireless and mobile networks. Gartner estimates that the automated provisioning of network services associated with a workload can reduce provisioning times for new applications by more than 80% while ensuring that manual provisioning mistakes are reduced.²

- **Security**: The automated and highly controllable configuration process of SDN may result in a more secure network. SDN-capable switches can filter—and optionally also modify—network packet headers. As a result, they can act as programmable firewalls, as well as intrusion-detection, intrusion-
prevention, and data-loss-prevention services on demand. However, SDN can also add both new security provisions and new security vulnerabilities to network architecture.

- **Virtualization**: SDN supports the dynamic allocation and replication of virtual loads, machines, and applications across data centers anywhere on the Internet.

- **Potentially Reduced Costs**: Another potential benefit of SDN is in the reduction of network hardware costs. With the implementation of SDN, expensive routers and switches are replaced with less expensive switches, reducing the capital expenditures on the overall network. Yet the move to centralized control also means that software licensing and SDN controller costs become more expensive. Further analysis is needed to determine if in fact the campus saves money by using SDN; at present, reduced network costs should not be a significant consideration for implementing SDN. Overall cost savings, especially when including staffing expenses, are not a certainty in the near term. However, in the long term, operational savings may be achieved by automating the orchestration of provisioning and deploying networks.

- **Infrastructure for Innovation**: Finally, SDN offers an important intangible benefit: the creation of an innovation platform for future services. It provides researchers and students with a platform that enables the development of next-generation network applications and a network infrastructure that is agile and flexible enough for a multitude of disciplines, each with its own unique requirements.

**Current State**

When conventional TCP/IP networks systems were first being planned, the concept of central control of the switches was considered, but the prevalent view was that the robustness of distributed routing protocols made a much better foundation for adaptive switching and routing. Although this has been successful over the years, it means that current conventional networks are managed as a collection of semiautonomous routers and switches. With SDN, the opportunity exists to leverage the simplicity and optimality of central routing decisions while using the conventional network as an out-of-band means to communicate with the various switches of a network.

To meet yesterday’s network topology needs, vendors concentrated on providing network routers and switches that focused on maximizing data-link speed, reducing latencies, increasing reliability and security, and offering a command interface for network administrators to manage the network. In that approach, each router or switch had to be configured for its place in the topology and reconfigured for desired changes either manually or through administrative scripts. As network complexity increases, the scalability of administering the network becomes problematic. The rise of IT services that span the wider Internet means that the ability to customize this complex network is also a challenge.

With server virtualization taking hold of data centers over the past three to four years, network function virtualization (NFV) has also come into focus as a primary, cost-effective way that network service providers can ramp up reliability, speed up deployment of new network services, or scale up or down existing ones. Virtual network functions such as load balancing, domain name service (DNS), or firewalls can now run on virtual machines instead of specialized hardware-based appliances. This virtualized environment can enable network service providers to raise their availability metrics with little additional effort and cost.

Virtual data centers have also popularized the idea of network overlays that enable the mobility or isolation of virtual machines between cloud infrastructures and across data centers. Overlays encapsulate
virtual networks within actual networks, simulating virtual network adjacency across the Internet. Today, when SDN is not in place, overlays are configured independently.

**Desired Future State**

In the early days of computing, applications took the limitations of the computer they ran on as a given. There were parameters—such as the number of CPUs or the amount of available memory—that were out of the application’s control. Over time, applications grew more complex and powerful, and operating systems were designed with the capability to provide simulated CPU, RAM, and other environmental configurations through virtualization; in some cases, applications could also drive interactions between the virtual and actual resources to control or influence the amount and configuration of actual resources provisioned for their virtual environment. As applications become more network-aware, their distributed functional requirements will drive their need to control their network environment in order to offer more timely and reliable service. With SDN, these applications will be able to make application programming interface (API) calls to request an underlying network designed to their needs, just as they can make calls to the operating system for its system needs.

While NFV and network overlays do not require SDN, applying the fundamental SDN concept of a standards-based central controller to programmatically configure these functions will vastly improve their management.

**Transitional State**

SDN democratizes the network by delegating the ability to configure it to the IT service using it. This comes with inherent risks because the network is complex and most people do not have deep expertise in running it. Great care must be taken in distributing the control of the network. Here we discuss some of the critical issues that higher education institutions must consider as they transition from their legacy networks to a software-defined network.

**Planning for SDN**

SDN is not yet a mature technology with settled standards, plug-and-play vendor products on the market, a deep pool of subject-matter experts and technologists to recruit from, and network-aware applications that can take advantage of a software-definable network. Without careful planning, a transition to SDN can be expensive and risky while providing few concrete benefits.

Because SDN deploys a control plane for the network, it must be protected against unauthorized use. The modern identity and access management platforms that higher education institutions are currently deploying support research and collaboration through federated identities using such standards as secure assertion markup language (SAML). To date, there is no integration between SDN protocols and SAML to enable Internet-wide yet granular technical and policy control over who can do what to the network through the SDN controller.

Higher education institutions should define a clear vision for the benefits they expect to derive from SDN. Clearly defined goals will inform the scope of the project, help manage the cost of the rollout, and limit its impact on the legacy production network. In addition, higher education institutions must also consider other important criteria, such as the degree to which their network control will be centralized; whether the
SDN solution is open or proprietary; the scalability and the true interoperability of the solution; and the role of hardware versus software in the SDN architecture.

**Assessing the State of the SDN Standards**

SDN started with the concept of decoupling the control plane of the network from its data plane. The logical next step was to provide a standard protocol for applications to request services from the SDN controller (via what are called the “northbound APIs”) and for the controller to program the network switches (via what are called the “southbound APIs”). The OpenFlow (OF) protocol was developed at Stanford University in 2009, and in 2011 the Open Networking Foundation (ONF) was launched as an industry consortium to take control of and commercialize the OF protocol. The quantum leap in value for SDN comes with representational state transfer (RESTful) APIs in the northbound OF protocol, allowing network-aware applications to design their own underlying networks through the SDN controller. Despite the clear value that standardization offers, however, not all vendors have fully aligned with OF API. Especially in the southbound APIs, vendors have adopted a variety of existing protocols, such as XMPP, Netconf, XML-RPC, Yang, and others to maximize their SDN controller’s ability to configure and manage their switches, leading to limited interoperability between vendor products. Caveat emptor!

**SDN Switches**

In an OF switch, all control functions are run on the SDN controller. In an OF-enabled switch, SDN-controlled forwarding as well as traditional switching and routing functions are supported, allowing the device to operate in hybrid SDN and legacy modes. This feature presumably makes it easier to implement a slow rollout of SDN and leverage existing infrastructure while exploring the benefits of SDN to a campus. As the control of switching functions such as forwarding, prioritization, rate limiting, virtual interface plumbing, and others are implemented in a sort of “network middleware,” the switching hardware will tend toward commoditization.

**SDN Controllers**

Multiple initiatives are in progress to build open-source SDN controller platforms. The Linux Foundation is host to the OpenDaylight Project (ODP), which is community led and industry supported. ODP’s goal is to support the OF standard and a comprehensive set of multivendor functionalities, from security to virtual network abstractions. ODP is primarily focused on SDN implementation in the data center.

Over the past year, the Open Networking Lab, essentially an SDN think tank of researchers at Stanford University and the University of California, Berkeley, has been working on the Open Networking Operating System (ONOS), which is scheduled to be unveiled in December 2014. ONOS will feature full support for the OF standard and greater scaleout and high availability built in, with support for controller state across a cluster of controllers, as well as a richer set of northbound and southbound APIs. ONOS appears to be more focused on SDN across the WAN.

Other open-source and commercial controller offerings exist, such as Cisco’s XNC, Juniper’s Contrail and its open release OpenContrail, VMware’s NSX, NT&T’s RYU, the open-source Floodlight OF controller, and others, each with its own strengths, weaknesses, and history.
Evaluating Vendor Offerings

Vendor partnerships can be valuable at a time when SDN products are not fully interoperable and the standards are still in a state of flux. Each institution must evaluate its needs for a solution that make sense for the unique state of its network. SDN is not a one-size-fits-all solution.\(^4\) It’s important to consider and evaluate both incumbent and nonincumbent vendors, as well as open-source options, to find the best fit for the goals and key drivers for SDN at your institution.

Rollout Strategy

SDN standards are still evolving, and the leading vendors continue to explore their own strategies for adapting to the new technologies while maintaining their market share and positioning themselves for growth. As a result, many SDN-enabled products are not cross-vendor interoperable. In addition, as an emerging technology, SDN supports a vendor-scape that includes many interesting start-ups innovating in this space, and they should be given due consideration.

It is too early, and too risky, to consider a forklift upgrade of institutional legacy networks in favor of one SDN solution or another. Does your organization have the expertise in programming fully secured virtual networks so that they do not affect the underlying production network adversely and do not expose sensitive or regulated data inadvertently? How mature and safe are the SDN controller codebases, and how will your organization secure the new, open, and complex network core from malicious attacks that could debilitate your entire network?

With tools such as network overlays and with strategic, point upgrades to specific network gear and careful selection of SDN controllers, institutions can prudently explore SDN without harming the legacy production network, while gaining valuable subject-matter experience and expertise in managing a software-defined network.

The bottom line remains simple: Clearly define and articulate the goal of your SDN initiative; initiate a full diligence effort to talk to vendors, researchers, and other innovators in the SDN technology space; and start small and allow your team to gain expertise and experience before defining a strategy that matches your wider institutional goals.

Conclusion

We believe that software-defined networking has a permanent place in the future of networking. Even now, in the early days of the technology, one can point to many successful SDN implementations in data centers and national networking (e.g., Internet2’s AL2S service). The use of SDN to better manage and more rapidly reconfigure data-center, enterprise, and national networks is an obvious fit. The future possibility of lower costs from network hardware commoditization and open-source controller software is real but difficult to predict. Less clear is when SDN technology and implementations will evolve to a point where a researcher is able to run an application using a standard campus network port that magically requests end-to-end interrealm resources from “the network,” uses these resources in real time for large-scale work, and then relinquishes the resources back to the network.
For More Information

Organizations

- **OpenDaylight**: OpenDaylight is an open platform for network programmability to enable SDN and create a solid foundation for Network Function Virtualization (NFV) for networks of any scale.

- **Open Networking Foundation**: ONF is a member-driven organization dedicated to the promotion and adoption of SDN through open standards development.

- **Open Networking Lab**: Open Networking Lab is a nonprofit organization that was formed to “pursue our vision of what networking could be for the public good” by building tools and platforms to accelerate SDN and educate the public and provide thought leadership on SDN.

- **The ONOS Project**: The ONOS Project is an open source network operating system by the Open Networking Lab.

- **OpenContrail**: OpenContrail is an Apache 2.0 licensed SDN controller.

- **SDNCentral**: SDNCentral is a “centralized source of news and resources” for network virtualization and SDN.

- **SDN Hub**: SDN Hub is a consortium of SDN enthusiasts promoting development and adoption of SDN solutions.

Articles, Papers, and Videos


Authors

Celeste Anderson
Director, External Networking Group
University of Southern California

Asbed Bedrossian
Director, Enterprise Middleware and Infrastructure
University of Southern California

James A. Jokl
Associate Vice President and Chief Enterprise Architect
University of Virginia

Timothy Lance
President and Board Chair
NYSERNet, Inc.

Contributors

Special thanks go to all the members of the ECAR Campus Cyberinfrastructure (ECAR-CCI) and Communications Infrastructure and Applications (ECAR-CIA) working groups, and notably to Guy T. Almes (Director, Academy for Advanced Telecommunications and Learning Technologies, Texas A&M University) and Clifford A. Lynch (Executive Director, Coalition for Networked Information) for their substantial contributions to this document.

Notes

1. For instance, Urs Hölzle, senior vice president of technical infrastructure at Google, was quoted as saying, “In utilization alone, we are hoping for a 20 percent to 30 percent reduction” in costs with the implementation of an SDN system; see Quentin Hardy, “Google Opens Up About Its Network,” New York Times, April 18, 2012.

