

Software-Defined Networking

Novel architecture allows programmers to quickly reconfigure network resource usage.

THE DEMANDS OF networked computer systems have changed dramatically since the early days of basic file sharing, peripheral sharing, or the hosting of companywide applications on a server. Today, organizations increasingly are using significantly more advanced computing environments to meet their needs, including cloud-based networks, virtualized desktops and servers, and remote data-storage devices, technologies that require significantly more computing resources, labor, and planning to properly deploy and maintain.

Enter software-defined networking (SDN), a new networking architecture that is designed to use standardized application programming interfaces (APIs) to quickly allow network programmers to define and reconfigure the way data or resources are handled within a network. The use of an API allows network applications (such as e-mail systems, cloud computing services, or telephony applications) to easily interface and reconfigure the network and its components (such as switches, racks of servers, virtual machines, and other end devices), or pull specific data, based on their particular requirements.

While SDN is not yet a household concept, it has garnered significant attention from major players in the virtualization and cloud computing space, another burgeoning segment of the computing world. Indeed, just over a year ago in July 2012, VMware Inc. agreed to acquire Nicira Networks, a Silicon Valley-based SDN startup that had flown under the radar for nearly five years, for \$1.26 billion.

“Networking has remained stuck in the mainframe era for 15 years,” says Andrew Harding, senior director of product marketing for Big Switch Networks, an SDN controller vendor.

Harding notes that, while significant advances in other areas of technology have occurred (such as the shift from basic mobile phones to the world of smartphones using open APIs, like those found in the Android ecosystem), networking architecture and protocols have not kept pace, until now.

Configuring Networks

The promise of SDN can be thought of as being somewhat analogous to how mobile applications are built to interact with each other. In the mobile world, an application can make an API available to other developers for use in their own applications, without permanently modifying the first application. For example, Google Maps offers an API that

allows applications to layer specific data on top of Google Maps data, such as a restaurant’s location, by pulling in the relevant content. In essence, these location-based service applications are configuring Google Maps data for their own use, without requiring any change in the way Google Maps configures its data.

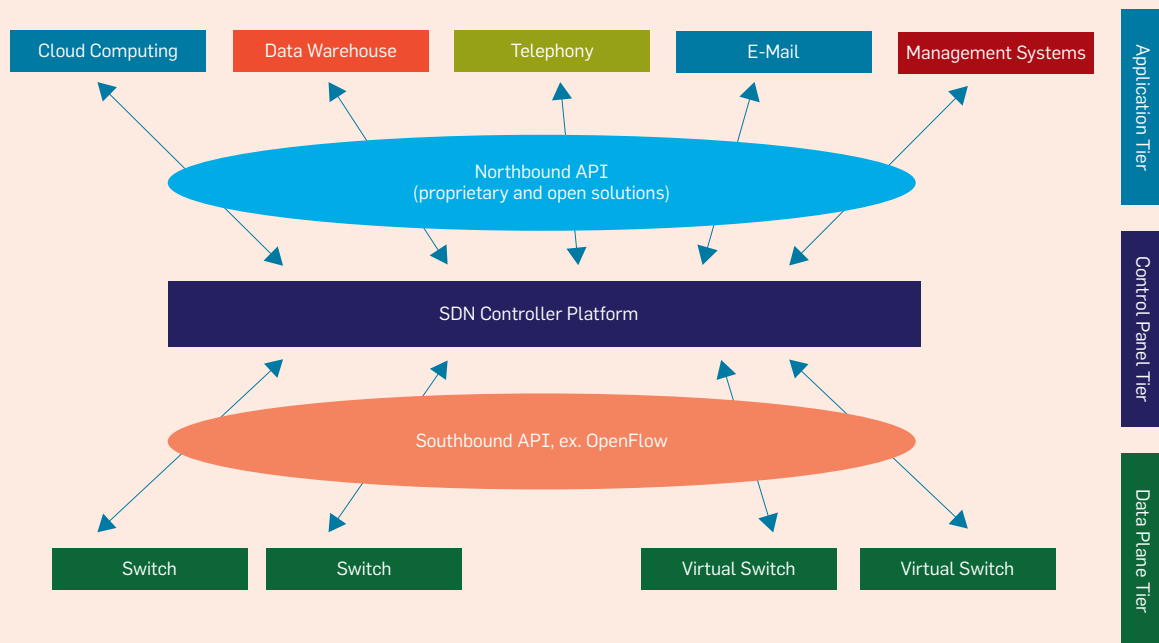
Similarly, in an SDN environment, applications can “reach through” to the network switches via an API and reconfigure the resources of the network to suit their needs. This ability to quickly reprogram or provision the network is achieved due to the way routers or switches are deployed.

Historically, the transmission of data has been dominated by the use of dedicated switches or routers to direct



Bruce Davie, principal engineer at VMware, notes SDN is “a mechanism, not a panacea.”

An example of SDN architecture.



packets between servers or other connected devices. These switches consist of two “planes,” or layers of the router interface: the data or forwarding plane, which handles the routing of data packets to its network destination; and the control plane, which creates the routing tables that determine how packets are sent to a destination. The control plane is also responsible for managing the connections between switches, handling errors and exceptions, and defining quality of service for different types of packets.

SDN decouples the link between the switch itself and the data-routing instructions, while adding an application programming interface between the two. These now “virtual” switches are not tied to any single piece of hardware or groups of devices, and thus can allow higher-level applications to pull data or reconfigure network resources from any connected device on the network.

“We’re still stuck in this manual configuration era,” Harding says. “But [now] you can program the network, [there are] a million applications you can pursue.”

Network Interfaces

SDN can be defined as a three-tiered “stack” architecture, in which appli-

cations and high-level instructions sit in the top tier, a controller sits in the middle directing data traffic, and a third tier resides at the bottom, containing physical and virtual switches.

Each control device within a network is equipped with one or more interfaces, which enables the device to communicate with other components. In networking parlance, these interfaces are described directionally, with each direction related to the relationship between devices. For example, a northbound interface describes the communication with a higher-level component, while a southbound interface allows a particular network component to communicate with a lower-level component.

Northbound APIs

In SDN, the northbound API interface on the controller enables applications and the overall management system to program the network and request services from it. This application tier often includes global automation and data management applications, as well as providing basic network functions such as data path computation, routing, and security.

Currently, no formalized standards have been ratified for northbound

APIs, with several dozen open and proprietary protocols being developed using different northbound APIs. The lack of a standard API is likely due to the varied nature of applications sitting above the controller, which can include managing cloud computing systems, network virtualization schemes, and other disparate or specialized functions.

Nevertheless, work on open northbound APIs is being done for specific vertical applications. OpenStack, a cloud computing effort backed by Arista Networks, Big Switch Networks, Brocade, VMware, and other SDN vendors, has developed the Quantum API, which is a vendor-agnostic API for defining logical networks and related network-based services for cloud-based systems. Several vendors have developed plugins for Quantum, which has helped it to become the default networking API for OpenStack, one of the largest open source cloud management platforms.

Southbound APIs

Though not explicitly required by SDN, OpenFlow is a protocol often used as the southbound API that defines a set of open commands for data forwarding. These commands allow routers to

discover the network's topology and define the behavior of physical and virtual switches, based on application requests sent via the northbound APIs. Note, however, that while commonly used in SDN architectures, OpenFlow is not a requirement of SDN, and organizations may opt to use other types of southbound APIs for the control of switches and devices.

According to Dan Pitt, executive director of the Open Networking Foundation, a trade organization working to promote software-defined networking and the use of the OpenFlow protocol, the open protocol can assist organizations in scaling and reconfiguring their networks, while supporting the growing trend of network virtualization.

"The perpetuation of manual configuration through command-line interfaces has long held networking back from the advances in virtualization enjoyed by the computing world, and has led to high operating costs, long delays in updating networks to meet business needs, and the introduction of errors," Pitt says. "Eliminating the need to tie applications to specific network details like ports and addresses makes it possible to evolve the network's physical aspects without the delay and cost of both rewriting the applications and manually configuring the network devices."

The early benefits of SDN are largely going to stem from the use of network virtualization, which allows for more dynamic network segmentation and utilization.

Faster Hardware

Another key benefit with SDN, and its architectural configuration of splitting the data control plane from the routing tables, is the ability to incorporate faster, more powerful hardware. For example, a network switch can be used to handle the forwarding of data packets, while a separate virtual server can be configured to run the network control plane. This split configuration permits the network development and runtime environment to be located on a more advanced, speedier platform, rather than being relegated to the lower-end, slower management processors used on hardware switches and routers.

Benefits and Challenges

The early benefits of SDN are largely going to stem from the use of network virtualization, which allows for more dynamic network segmentation and utilization. SDN permits a more efficient use of network resources to support virtual machines (controlled by hypervisors, or the software used to support virtualization), as well as greater flexibility, via OpenFlow virtual switches. In essence, virtual machines that normally would be dedicated to static IP addresses can now be dynamically shared across virtual switches, allowing greater flexibility, as well as reduced operational expenses due to improved data efficiency and density.

While current real-world implementations are essentially pilot programs, the Open Networking Foundation is touting early efficiency gains and cost reductions. "When Genesis Hosting adopted SDN in their hosting facility, they gained a reduction in network administration costs of 50% and a reduction in IP address usage of 60%," Pitt says. "When Google converted its G-Scale WAN that interconnects its data centers worldwide to a 100% OpenFlow network, they reduced overprovisioning [to achieve] utilization levels above 95% with zero loss."

That said, not everyone is sold on the benefits of OpenFlow to drive SDN and other advanced networking appli-

Milestones

BCS Recognizes Two Oxford CS Lecturers

BCS, The Chartered Institute for IT, has recognized two University of Oxford computer science lecturers for outstanding contributions to the field. Boris Motik has been named the recipient of the BCS Roger Needham Award, while Samson Abramsky F.R.S. is being awarded the BCS Lovelace Medal.

The Roger Needham Award, sponsored by Microsoft Research Cambridge, is awarded for a distinguished research contribution in computer science by a U.K.-based researcher within 10 years of receiving their Ph.D.

Motik was recognized for making major contributions to the design and standardization

of the OWL2 ontology language, widely used in industrial research and applications. His research, on the development of techniques for intelligent management of large amounts of data, involves building advanced data management systems that can exploit background knowledge about an application domain in order to improve common tasks such as information production and search. Such systems are used in a broad range of applications, such as annotation of healthcare records, intelligent information management and retrieval in tourism, and provision of context-enabled services in consumer devices

such as smartphones.

The BCS Lovelace Medal was established in 1998 in honor of Lady Augusta Ada Byron, Countess of Lovelace and daughter of Lord Byron. The medal is presented annually to individuals who have made a significant contribution to the advancement of information systems.

Since the 1980s, Abramsky has helped to set the modern research agenda for computer science. His contributions in each of the past three decades had a major impact on the field; they include domain theory in logical form, game semantics, and categorical quantum mechanics. He has shown

the ability to change research fields and to establish new interdisciplinary approaches. His work over the past decade has shown that methods and concepts developed in theoretical computer science can be applied very directly in quantum information, and to the foundations of quantum mechanics.

Both Motik and Abramsky are based at the University of Oxford's Department of Computer Science. Motik will present the 2013 BCS Needham lecture in November, and Abramsky will deliver his Lovelace lecture in 2014.

For further information see: www.bcs.org/academy-awards.

cations. Pere Monclus, co-founder and CTO of PLUMgrid, a networking platform vendor, asks whether “OpenFlow is enough” to support the applications of today and tomorrow.

Monclus argues that the current rate of new versions of APIs (such as OpenFlow) being released is perhaps occurring too quickly, without allowing for innovations to drive the market. “If you want to go create an environment that you can develop applications that [will extend the functions of] the data plane, then OpenFlow is like a closed environment, because it already assumes that I know what you’re going to need in terms of a data plane,” Monclus says.

Furthermore, SDN is faced with a major challenge from traditional networking vendors, which have long relied on sales of their proprietary networking hardware and software, and could see a significant decline in the sales if SDN is rapidly adopted. Additionally, many organizations simply do not have the time, expertise, or capital to invest in a completely new networking architecture, particularly smaller organizations with limited IT staff and budgets.

That said, networking giants including Cisco, Microsoft, Hewlett-Packard, and IBM (among many others) joined in April 2013 with SDN vendors such as Big Switch Networks, Brocade, VMware, Arista, and PLUMgrid, to announce OpenDaylight, a new open source software project designed to create a collection of software for building networks that is largely based around the concepts of SDN. Despite the stated commitment to creating an industrywide, open source effort to promote SDN, there has been legitimate criticism that the larger networking companies have joined the consortium to push their own proprietary APIs to retain control over some of the hardware and software that may be used in future SDN implementations.

Still, as a result of the industry’s activities surrounding SDN, there is plenty of positive buzz about the technology’s ability to reduce the time and cost of rolling out new applications, routing and transferring data, and reconfiguring networks virtually. This enthusiasm could result in gains in a decidedly non-virtual way: in March 2013, IDC projected the SDN mar-

Many organizations simply do not have the time, expertise, or capital to invest in a completely new networking architecture.

ket would reach \$3.7 billion by 2016, capturing a 35% share of the switching market, up from what the market tracker quantified as “negligible” penetration in 2012. 

Further Reading

SDN Inventor Martin Casado’s 2005 Thesis on SDN: http://yuba.stanford.edu/~casado/vns_sigcse.pdf

“Ending The Confusion Around Software Defined Networking (SDN): A Taxonomy,” Gartner, <http://www.gartner.com/resId=2367616>.

Open Networking Foundation, October 2012 Interoperability Event Technical Paper, v0.4: <https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow-test/onf-testing-interop-oct-2012-tech-doc-v0-4.pdf>

Das, S., Parulkar, G., McKeown, N. Why OpenFlow/SDN Can Succeed Where GMPLS Failed, *ECOC Technical Digest, 2012 OSA*, <http://yuba.stanford.edu/~nickm/papers/ECEOC-2012-Tu.1.D.1.pdf>

Kobayashi, M., Seetharaman, S., Parulkar, G., Appenzeller, G., Little, G., van Reijndam, J., Weissmann, P., McKeown, N. Maturing of OpenFlow and Software Defined Networking through Deployments, *Elsevier August 14, 2012*, http://yuba.stanford.edu/~nickm/papers/openflow_deployment_journal_paper_aug2012.pdf

Video Content

Open Networking Summit 2013: Day One Highlights: <https://www.youtube.com/watch?v=pDY9pjVhMew>

Origins and Evolution of OpenFlow/SDN - Martin Casado, Open Networking Summit 2011: <https://www.youtube.com/watch?v=4Cb91JT-Xb4>

Keith Kirkpatrick is Principal of 4K Research & Consulting, LLC, based in Lynbrook, NY.

© 2013 ACM 0001-0782/13/09 \$15.00



Association for
Computing Machinery

ACM Conference Proceedings Now Available via Print-on-Demand!

Did you know that you can now order many popular ACM conference proceedings via print-on-demand?

Institutions, libraries and individuals can choose from more than 100 titles on a continually updated list through Amazon, Barnes & Noble, Baker & Taylor, Ingram and NACSCORP: CHI, KDD, Multimedia, SIGIR, SIGCOMM, SIGCSE, SIGMOD/PODS, and many more.

For available titles and ordering info, visit: librarians.acm.org/pod



Copyright of Communications of the ACM is the property of Association for Computing Machinery and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.