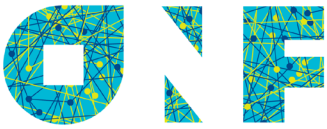


OPEN NETWORKING
FOUNDATION

Software-Defined Networking: The New Norm for Networks

ONF White Paper
April 13, 2012

Table of Contents

Executive Summary

Traditional network architectures are ill-suited to meet the requirements of today's enterprises, carriers, and end users. Thanks to a broad industry effort spearheaded by the Open Networking Foundation (ONF), Software-Defined Networking (SDN) is transforming networking architecture.

In the SDN architecture, the control and data planes are decoupled, network intelligence and state are logically centralized, and the underlying network infrastructure is abstracted from the applications. As a result, enterprises and carriers gain unprecedented programmability, automation, and network control, enabling them to build highly scalable, flexible networks that readily adapt to changing business needs.

The ONF is a non-profit industry consortium that is leading the advancement of SDN and standardizing critical elements of the SDN architecture such as the OpenFlow™ protocol, which structures communication between the control and data planes of supported network devices. OpenFlow is the first standard interface designed specifically for SDN, providing high-performance, granular traffic control across multiple vendors' network devices.

OpenFlow-based SDN is currently being rolled out in a variety of networking devices and software, delivering substantial benefits to both enterprises and carriers, including:

- Centralized management and control of networking devices from multiple vendors;
- Improved automation and management by using common APIs to abstract the underlying networking details from the orchestration and provisioning systems and applications;
- Rapid innovation through the ability to deliver new network capabilities and services without the need to configure individual devices or wait for vendor releases;

© Open Networking Foundation. All rights reserved.

- Programmability by operators, enterprises, independent software vendors, and users (not just equipment manufacturers) using common programming environments, which gives all parties new opportunities to drive revenue and differentiation;
- Increased network reliability and security as a result of centralized and automated management of network devices, uniform policy enforcement, and fewer configuration errors;
- More granular network control with the ability to apply comprehensive and wide-ranging policies at the session, user, device, and application levels; and
- Better end-user experience as applications exploit centralized network state information to seamlessly adapt network behavior to user needs.

SDN is a dynamic and flexible network architecture that protects existing investments while future-proofing the network. With SDN, today's static network can evolve into an extensible service delivery platform capable of responding rapidly to changing business, end-user, and market needs.

The Need for a New Network Architecture

The explosion of mobile devices and content, server virtualization, and advent of cloud services are among the trends driving the networking industry to reexamine traditional network architectures. Many conventional networks are hierarchical, built with tiers of Ethernet switches arranged in a tree structure. This design made sense when client-server computing was dominant, but such a static architecture is ill-suited to the dynamic computing and storage needs of today's enterprise data centers, campuses, and carrier environments. Some of the key computing trends driving the need for a new network paradigm include:

- **Changing traffic patterns:** Within the enterprise data center, traffic patterns have changed significantly. In contrast to client-server applications where the bulk of the communication occurs between one client and one server, today's applications access different databases and servers, creating a flurry of "east-west" machine-to-machine traffic before returning data to the end user device in the classic "north-south" traffic pattern. At the same time, users are changing network traffic patterns as they push for access to corporate content and applications from any type of device (including their own), connecting from anywhere, at any time. Finally, many enterprise data centers managers are contemplating a utility computing model, which might include a private cloud, public cloud, or some mix of both, resulting in additional traffic across the wide area network.

© Open Networking Foundation. All rights reserved.

- **The "consumerization of IT":** Users are increasingly employing mobile personal devices such as smartphones, tablets, and notebooks to access the corporate network. IT is under pressure to accommodate these personal devices in a fine-grained manner while protecting corporate data and intellectual property and meeting compliance mandates.
- **The rise of cloud services:** Enterprises have enthusiastically embraced both public and private cloud services, resulting in unprecedented growth of these services. Enterprise business units now want the agility to access applications, infrastructure, and other IT resources on demand and à la carte. To add to the complexity, IT's planning for cloud services must be done in an environment of increased security, compliance, and auditing requirements, along with business reorganizations, consolidations, and mergers that can change assumptions overnight. Providing self-service provisioning, whether in a private or public cloud, requires elastic scaling of computing, storage, and network resources, ideally from a common viewpoint and with a common suite of tools.
- **"Big data" means more bandwidth:** Handling today's "big data" or mega datasets requires massive parallel processing on thousands of servers, all of which need direct connections to each other. The rise of mega datasets is fueling a constant demand for additional network capacity in the data center. Operators of hyperscale data center networks face the daunting task of scaling the network to previously unimaginable size, maintaining any-to-any connectivity without going broke.

Limitations of Current Networking Technologies

Meeting current market requirements is virtually impossible with traditional network architectures. Faced with flat or reduced budgets, enterprise IT departments are trying to squeeze the most from their networks using device-level management tools and manual processes. Carriers face similar challenges as demand for mobility and bandwidth explodes; profits are being eroded by escalating capital equipment costs and flat or declining revenue. Existing network architectures were not designed to meet the requirements of today's users, enterprises, and carriers; rather network designers are constrained by the limitations of current networks, which include:

- **Complexity that leads to stasis:** Networking technology to date has consisted largely of discrete sets of protocols designed to connect hosts reliably over arbitrary distances, link speeds, and topologies. To meet business and technical needs over the last few decades, the industry has evolved networking protocols to deliver higher performance and reliability, broader connectivity, and more stringent security.

© Open Networking Foundation. All rights reserved.

Protocols tend to be defined in isolation, however, with each solving a specific problem and without the benefit of any fundamental abstractions. This has resulted in one of the primary limitations of today's networks: complexity. For example, to add or move any device, IT must touch multiple switches, routers, firewalls, Web authentication portals, etc. and update ACLs, VLANs, quality of services (QoS), and other protocol-based mechanisms using device-level management tools. In addition, network topology, vendor switch model, and software version all must be taken into account. Due to this complexity, today's networks are relatively static as IT seeks to minimize the risk of service disruption.

The static nature of networks is in stark contrast to the dynamic nature of today's server environment, where server virtualization has greatly increased the number of hosts requiring network connectivity and fundamentally altered assumptions about the physical location of hosts. Prior to virtualization, applications resided on a single server and primarily exchanged traffic with select clients. Today, applications are distributed across multiple virtual machines (VMs), which exchange traffic flows with each other. VMs migrate to optimize and rebalance server workloads, causing the physical end points of existing flows to change (sometimes rapidly) over time. VM migration challenges many aspects of traditional networking, from addressing schemes and namespaces to the basic notion of a segmented, routing-based design.

In addition to adopting virtualization technologies, many enterprises today operate an IP converged network for voice, data, and video traffic. While existing networks can provide differentiated QoS levels for different applications, the provisioning of those resources is highly manual. IT must configure each vendor's equipment separately, and adjust parameters such as network bandwidth and QoS on a per-session, per-application basis. Because of its static nature, the network cannot dynamically adapt to changing traffic, application, and user demands.

- **Inconsistent policies:** To implement a network-wide policy, IT may have to configure thousands of devices and mechanisms. For example, every time a new virtual machine is brought up, it can take hours, in some cases days, for IT to reconfigure ACLs across the entire network. The complexity of today's networks makes it very difficult for IT to apply a consistent set of access, security, QoS, and other policies to increasingly mobile users, which leaves the enterprise vulnerable to security breaches, non-compliance with regulations, and other negative consequences.

© Open Networking Foundation. All rights reserved.

- **Inability to scale:** As demands on the data center rapidly grow, so too must the network grow. However, the network becomes vastly more complex with the addition of hundreds or thousands of network devices that must be configured and managed. IT has also relied on link oversubscription to scale the network, based on predictable traffic patterns; however, in today's virtualized data centers, traffic patterns are incredibly dynamic and therefore unpredictable.

  Mega-operators, such as Google, Yahoo!, and Facebook, face even more daunting scalability challenges. These service providers employ large-scale parallel processing algorithms and associated datasets across their entire computing pool. As the scope of end-user applications increases (for example, crawling and indexing the entire world wide web to instantly return search results to users), the number of computing elements explodes and data-set exchanges among compute nodes can reach petabytes. These companies need so-called hyperscale networks that can provide high-performance, low-cost connectivity among hundreds of thousands—potentially millions—of physical servers. Such scaling cannot be done with manual configuration.

  To stay competitive, carriers must deliver ever-higher value, better-differentiated services to customers. Multi-tenancy further complicates their task, as the network must serve groups of users with different applications and different performance needs. Key operations that appear relatively straightforward, such as steering a customer's traffic flows to provide customized performance control or on-demand delivery, are very complex to implement with existing networks, especially at carrier scale. They require specialized devices at the network edge, thus increasing capital and operational expenditure as well as time-to-market to introduce new services.

- **Vendor dependence:** Carriers and enterprises seek to deploy new capabilities and services in rapid response to changing business needs or user demands. However, their ability to respond is hindered by vendors' equipment product cycles, which can range to three years or more. Lack of standard, open interfaces limits the ability of network operators to tailor the network to their individual environments.

  This mismatch between market requirements and network capabilities has brought the industry to a tipping point. In response, the industry has created the Software-Defined Networking (SDN) architecture and is developing associated standards.

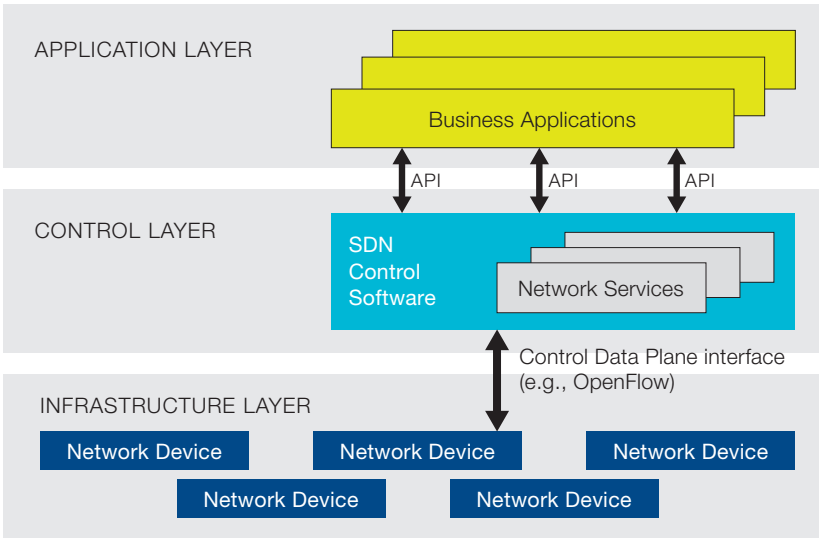
© Open Networking Foundation. All rights reserved.

Introducing Software-Defined Networking

Software Defined Networking (SDN) is an emerging network architecture where network control is decoupled from forwarding and is directly programmable. This migration of control, formerly tightly bound in individual network devices, into accessible computing devices enables the underlying infrastructure to be abstracted for applications and network services, which can treat the network as a logical or virtual entity.

Figure 1 depicts a logical view of the SDN architecture. Network intelligence is (logically) centralized in software-based SDN controllers, which maintain a global view of the network. As a result, the network appears to the applications and policy engines as a single, logical switch. With SDN, enterprises and carriers gain vendor-independent control over the entire network from a single logical point, which greatly simplifies the network design and operation. SDN also greatly simplifies the network devices themselves, since they no longer need to understand and process thousands of protocol standards but merely accept instructions from the SDN controllers.

FIGURE 1
Software-Defined Network Architecture



Perhaps most importantly, network operators and administrators can programmatically configure this simplified network abstraction rather than having to hand-code tens of thousands of lines of configuration scattered among thousands of devices. In addition, leveraging the SDN controller's centralized intelligence, IT can alter network behavior in real-time and deploy new applications and network services in a matter of hours or days,

© Open Networking Foundation. All rights reserved.

rather than the weeks or months needed today. By centralizing network state in the control layer, SDN gives network managers the flexibility to configure, manage, secure, and optimize network resources via dynamic, automated SDN programs. Moreover, they can write these programs themselves and not wait for features to be embedded in vendors' proprietary and closed software environments in the middle of the network.

In addition to abstracting the network, SDN architectures support a set of APIs that make it possible to implement common network services, including routing, multicast, security, access control, bandwidth management, traffic engineering, quality of service, processor and storage optimization, energy usage, and all forms of policy management, custom tailored to meet business objectives. For example, an SDN architecture makes it easy to define and enforce consistent policies across both wired and wireless connections on a campus.

Likewise, SDN makes it possible to manage the entire network through intelligent orchestration and provisioning systems. The Open Networking Foundation is studying open APIs to promote multi-vendor management, which opens the door for on-demand resource allocation, self-service provisioning, truly virtualized networking, and secure cloud services.

Thus, with open APIs between the SDN control and applications layers, business applications can operate on an abstraction of the network, leveraging network services and capabilities without being tied to the details of their implementation. SDN makes the network not so much "application-aware" as "application-customized" and applications not so much "network-aware" as "network-capability-aware". As a result, computing, storage, and network resources can be optimized.

Inside OpenFlow

OpenFlow is the first standard communications interface defined between the control and forwarding layers of an SDN architecture. OpenFlow allows direct access to and manipulation of the forwarding plane of network devices such as switches and routers, both physical and virtual (hypervisor-based). It is the absence of an open interface to the forwarding plane that has led to the characterization of today's networking devices as monolithic, closed, and mainframe-like. No other standard protocol does what OpenFlow does, and a protocol like OpenFlow is needed to move network control out of the networking switches to logically centralized control software.

SDN USE CASES

The ONF is guided by prominent enterprises and service providers, systems and applications developers, software and computer companies, and semiconductor and networking vendors. This diverse cross-section of the communications and computing industries is helping to ensure that SDN and associated standards effectively address the needs of network operators in each segment of the marketplace, including:

THE ENTERPRISE

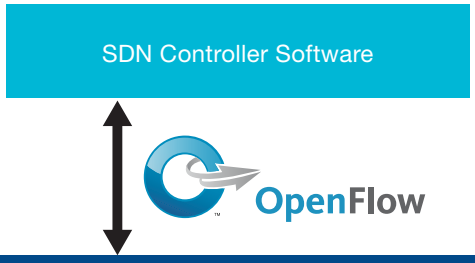
- Campus – SDN's centralized, automated control and provisioning model supports the convergence of data, voice, and video as well as anytime, anywhere access by enabling IT to enforce policies consistently across both the wired and wireless infrastructures. Likewise, SDN supports automated provisioning and management of network resources, determined by individual user profiles and application requirements, to ensure an optimal user experience within the enterprise's constraints.
- Data center – The SDN architectures facilitates network virtualization, which enables hyper-scalability in the data center, automated VM migration, tighter integration with storage, better server utilization, lower energy use, and bandwidth optimization.
- Cloud – Whether used to support a private or hybrid cloud environment, SDN allows network resources to be allocated in a highly elastic way, enabling rapid provisioning of cloud services and more flexible hand-off to the external cloud provider. With tools to safely manage their virtual networks, enterprises and business units will trust cloud services more and more.

continued on next page

© Open Networking Foundation. All rights reserved.

OpenFlow can be compared to the instruction set of a CPU. As shown in Figure 2, the protocol specifies basic primitives that can be used by an external software application to program the forwarding plane of network devices, just like the instruction set of a CPU would program a computer system.

FIGURE 2
Example of OpenFlow Instruction Set



OpenFlow-enabled Network Device

Flow Table comparable to an instruction set

MAC src	MAC dst	IP Src	IP Dst	TCP dport	...	Action	Count
*	10:20:.	*	*	*	*	port 1	250
*	*	*	5.6.7.8	*	*	port 2	300
*	*	*	*	25	*	drop	892
*	*	*	192.*	*	*	local	120
*	*	*	*	*	*	controller	11

SDN USE CASES

continued from previous page

CARRIERS AND SERVICE PROVIDERS

SDN offers carriers, public cloud operators, and other service providers the scalability and automation necessary to implement a utility computing model for IT-as-a-Service, by simplifying the roll-out of custom and on-demand services, along with migration to a self-service paradigm. SDN's centralized, automated control and provisioning model makes it much easier to support multi-tenancy; to ensure network resources are optimally deployed; to reduce both CapEx and OpEx; and to increase service velocity and value.

The OpenFlow protocol is implemented on both sides of the interface between network infrastructure devices and the SDN control software. OpenFlow uses the concept of flows to identify network traffic based on pre-defined match rules that can be statically or dynamically programmed by the SDN control software. It also allows IT to define how traffic should flow through network devices based on parameters such as usage patterns, applications, and cloud resources. Since OpenFlow allows the network to be programmed on a per-flow basis, an OpenFlow-based SDN architecture provides extremely granular control, enabling the network to respond to real-time changes at the application, user, and session levels. Current IP-based routing does not provide this level of control, as all flows between two endpoints must follow the same path through the network, regardless of their different requirements.

The OpenFlow protocol is a key enabler for software-defined networks and currently is the only standardized SDN protocol that allows direct manipulation of the forwarding plane of network devices. While initially applied to Ethernet-based networks, OpenFlow switching can extend to a

© Open Networking Foundation. All rights reserved.

much broader set of use cases. OpenFlow-based SDNs can be deployed on existing networks, both physical and virtual. Network devices can support OpenFlow-based forwarding as well as traditional forwarding, which makes it very easy for enterprises and carriers to progressively introduce OpenFlow-based SDN technologies, even in multi-vendor network environments.

The Open Networking Foundation is chartered to standardize OpenFlow and does so through technical working groups responsible for the protocol, configuration, interoperability testing, and other activities, helping to ensure interoperability between network devices and control software from different vendors. OpenFlow is being widely adopted by infrastructure vendors, who typically have implemented it via a simple firmware or software upgrade. OpenFlow-based SDN architecture can integrate seamlessly with an enterprise or carrier's existing infrastructure and provide a simple migration path for those segments of the network that need SDN functionality the most.

Benefits of OpenFlow-Based Software-Defined Networks

For enterprises and carriers alike, SDN makes it possible for the network to be a competitive differentiator, not just an unavoidable cost center. OpenFlow-based SDN technologies enable IT to address the high-bandwidth, dynamic nature of today's applications, adapt the network to ever-changing business needs, and significantly reduce operations and management complexity.

The benefits that enterprises and carriers can achieve through an OpenFlow-based SDN architecture include:

- **Centralized control of multi-vendor environments:** SDN control software can control any OpenFlow-enabled network device from any vendor, including switches, routers, and virtual switches. Rather than having to manage groups of devices from individual vendors, IT can use SDN-based orchestration and management tools to quickly deploy, configure, and update devices across the entire network.
- **Reduced complexity through automation:** OpenFlow-based SDN offers a flexible network automation and management framework, which makes it possible to develop tools that automate many management tasks that are done manually today. These automation tools will reduce operational overhead, decrease network instability introduced by operator error, and support emerging IT-as-a-Service and self-service provisioning models.

© Open Networking Foundation. All rights reserved.

In addition, with SDN, cloud-based applications can be managed through intelligent orchestration and provisioning systems, further reducing operational overhead while increasing business agility.

- **Higher rate of innovation:** SDN adoption accelerates business innovation by allowing IT network operators to literally program—and reprogram—the network in real time to meet specific business needs and user requirements as they arise. By virtualizing the network infrastructure and abstracting it from individual network services, for example, SDN and OpenFlow give IT—and potentially even users—the ability to tailor the behavior of the network and introduce new services and network capabilities in a matter of hours.
- **Increased network reliability and security:** SDN makes it possible for IT to define high-level configuration and policy statements, which are then translated down to the infrastructure via OpenFlow. An OpenFlow-based SDN architecture eliminates the need to individually configure network devices each time an end point, service, or application is added or moved, or a policy changes, which reduces the likelihood of network failures due to configuration or policy inconsistencies.

  Because SDN controllers provide complete visibility and control over the network, they can ensure that access control, traffic engineering, quality of service, security, and other policies are enforced consistently across the wired and wireless network infrastructures, including branch offices, campuses, and data centers. Enterprises and carriers benefit from reduced operational expenses, more dynamic configuration capabilities, fewer errors, and consistent configuration and policy enforcement.
- **More granular network control:** OpenFlow's flow-based control model allows IT to apply policies at a very granular level, including the session, user, device, and application levels, in a highly abstracted, automated fashion. This control enables cloud operators to support multi-tenancy while maintaining traffic isolation, security, and elastic resource management when customers share the same infrastructure.
- **Better user experience:** By centralizing network control and making state information available to higher-level applications, an SDN infrastructure can better adapt to dynamic user needs. For instance, a carrier could introduce a video service that offers premium subscribers the highest possible resolution in an automated and transparent manner. Today, users must explicitly select a resolution setting, which the network may or may not be able to support, resulting in delays and interruptions that degrade the

© Open Networking Foundation. All rights reserved.

user experience. With OpenFlow-based SDN, the video application would be able to detect the bandwidth available in the network in real time and automatically adjust the video resolution accordingly.

Conclusion

Trends such as user mobility, server virtualization, IT-as-a-Service, and the need rapidly to respond to changing business conditions place significant demands on the network—demands that today's conventional network architectures can't handle. Software-Defined Networking provides a new, dynamic network architecture that transforms traditional network backbones into rich service-delivery platforms.

By decoupling the network control and data planes, OpenFlow-based SDN architecture abstracts the underlying infrastructure from the applications that use it, allowing the network to become as programmable and manageable at scale as the computer infrastructure that it increasingly resembles. An SDN approach fosters network virtualization, enabling IT staff to manage their servers, applications, storage, and networks with a common approach and tool set. Whether in a carrier environment or enterprise data center and campus, SDN adoption can improve network manageability, scalability, and agility.

The Open Networking Foundation has fostered a vibrant ecosystem around SDN that spans infrastructure vendors large and small, including application developers, software companies, systems and semiconductor manufacturers, and computer companies, plus various kinds of end users. OpenFlow switching is already being incorporated into a number of infrastructure designs, both physical and virtual, as well as SDN controller software. Network services and business applications already interface with SDN controllers, providing better integration and coordination between them.

The future of networking will rely more and more on software, which will accelerate the pace of innovation for networks as it has in the computing and storage domains. SDN promises to transform today's static networks into flexible, programmable platforms with the intelligence to allocate resources dynamically, the scale to support enormous data centers and the virtualization needed to support dynamic, highly automated, and secure cloud environments. With its many advantages and astonishing industry momentum, SDN is on the way to becoming the new norm for networks.

Open Networking Foundation / 2275 E. Bayshore Road, Suite 103 / Palo Alto, CA 94303 / www.opennetworking.org

Open Networking Foundation, the ONF symbol, and OpenFlow are registered trademarks of the Open Networking Foundation, in the United States and/or in other countries. All other brands, products, or service names are or may be trademarks or service marks of, and are used to identify, products or services of their respective owners.

© Open Networking Foundation. All rights reserved.