Software-Defined Networking: Discover How to Save Money and Generate New Revenue

The Talk of the Industry

Software-defined networking (SDN) can help businesses and service providers reduce network operations complexity and costs. The basis for these benefits is the abstraction of the control functions from the network forwarding devices (such as switches, routers, and Layer 4 to 7 appliances). Abstraction helps simplify development of new and modification of existing services and applications, and that simplification drives costs down and contributes to faster revenue returns. Many people in the industry advocate the relocation of control functions to centralized general-purpose servers, typically x86 class, in data centers. This approach promises to lower equipment costs through the use of lower-priced, mass-produced hardware for the forwarding devices.

In addition, the SDN community is discussing how service providers can use SDN to make money. These discussions typically include three main methods:

- Repurposing existing revenue-generating network services onto a software-defined network that yields greater flexibility and a lower cost structure, resulting in higher margins
- The ability to modify or spin up services much more quickly, capturing new opportunities to enhance revenue
- A lower-cost, more flexible network, which can promote more application innovation in a dynamic service environment, opening new market opportunities

Is the Industry on the Right Track?

The SDN economics promise is indeed tantalizing. It presents an opportunity to take advantage of mass-produced, low-cost hardware and much faster software development, using the experience of web proliferation as the benchmark. Service providers, technology vendors, and enterprises are therefore looking at SDN technologies closely. Some service providers are already aggressively implementing them.

What are some practical implementations and lessons learned? How can SDN reduce expenses and make money for you?

Today, you and all other network operators still need hardware at several locations across the services infrastructure.

These locations include the customer premises, fixed and mobile access network layers, flow aggregation points, provider edge, high-capacity network core, data center, and interconnection and exchange points. Wherever in the world you operate, you have to manage a great variety of topologies and geographies. That means coping with the economic practicalities of where fiber runs can be made and where network, storage, and computing gear can be located. You must also take into account the effects of local energy availability, real estate costs, and more.

Consider a New Operational Model

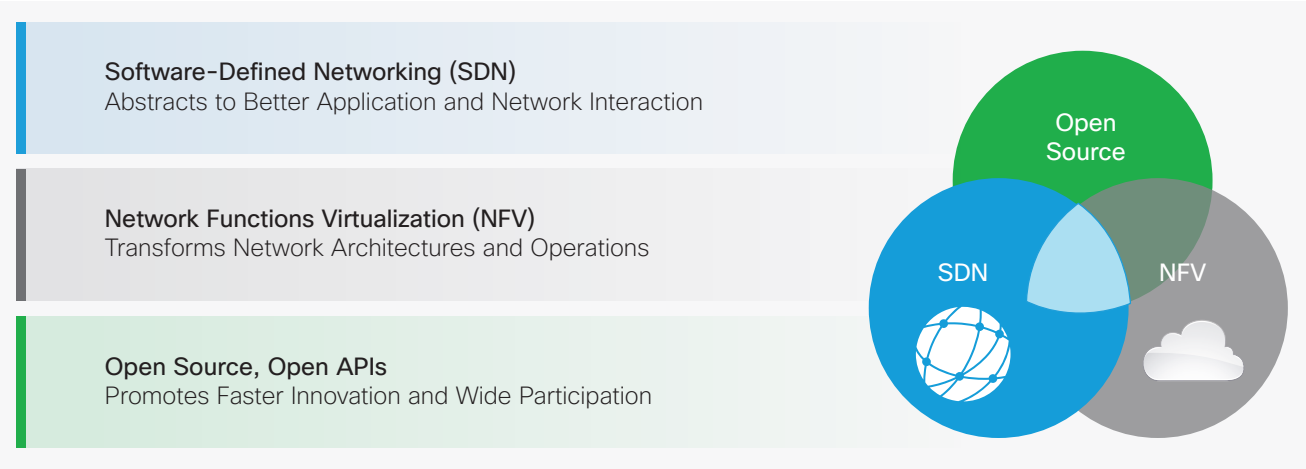
For SDN to be successful, a new operational model must be applied across all service provider environments and infrastructures, with a clear mission to achieve significant and measurable reduction in operational costs and complexity. Your main focus should be on your desired business outcomes: lower costs, less operational complexity, and greater agility. Focusing on centralizing network functions into big data centers and using mass-produced generic hardware will not necessarily yield optimal economics for you.

The problems of cost, rigidity, and complexity are mainly caused by the control and management frameworks. These legacy frameworks have not included suitable data-abstraction models, at least not those that serve the needs of the services your customers expect now and in the future. These frameworks have also been burdened with too many proprietary interfaces through which to manage network equipment.

To reduce cost, complexity, and rigidity, we must break up the control and management problem into smaller pieces, make it modular. Furthermore, we need to develop good data-abstraction models that can be easily used across various software modules and layer interaction points. Fortunately, some very good work has already been done in the Internet Engineering Taskforce (IETF). For example, the YANG data-modeling language builds a consistent abstraction of network elements that can be programmed with the network configuration (NETCONF) protocol.

Another example is the effort in the European Telecommunications Standards Institute (ETSI) working groups related to network functions virtualization (NFV). Indeed, it would be very difficult for SDN to achieve its goals without a high degree of virtualization of network functions. The core NFV and SDN work activities in the industry are highly complementary. Also complementary are the various open source projects such as OpenDaylight and Open Platform for NFV (OPNFV).

Figure 1. Complementary Technology Initiatives



Fulfilling the promise of SDN also requires a commitment by you to look at your development operations and production environments with a new, more entrepreneurial perspective. This perspective values the flexibility of embracing new operational and business models. It encourages openness to new technologies and independent application development. To support application openness and speed, we recommend that you seriously consider a declarative model between the applications and the network. (Refer to the Declarative Programming link under in the section "For More Information.") Such a model will allow users to naturally describe applications requirements and initiate their deployment across the network without having to consider the underlying complexities and differences across networks. It is up to the intelligence of the network to maintain the service-level requirements of the applications, regardless of the scale. The network has to be highly resilient in stress conditions and converge quickly upon failure in both the data plane and control plane.

Developing Your Software Ecosystem

Many network service providers are looking with interest at the models that companies such as Google, Apple, and Amazon have deployed. These models, though varied, provide platforms that broad ecosystems of third parties can use to quickly develop and deploy their own applications. A successful SDN provides the same opportunity to open up the network environment to broader software and application development and distribution, helping to address a myriad of niche and mass-market needs and to generate new revenue. You can use improved programmability and application interaction to do a similar thing, providing a broad platform that is inviting to third parties. This can result in compelling services and applications for their users that benefit you too.

Standard, simple, open APIs, such as Representational State Transfer (REST), support simpler integration with the rest of the service framework, with fewer problems. Each application developer that partners with a service provider designs software for these APIs abstracted from the underlying network. The developers simply don't need to know the intricacies of networking, just what their applications need from it.

Your SDN will create and enforce policies that are used to configure network and service infrastructure to meet the needs of the overriding applications. An application can ask the network to provide certain quality-of-service (QoS) characteristics, security, and storage without needing detailed knowledge of network complexities, such as data forwarding, data collection, topology, accessibility, loading, and scale. The SDN infrastructure responds to and interacts with applications by configuring the service or by indicating that resources are not available to provide a satisfactory user experience. Essentially, applications are abstracted from service control, which is, in turn, abstracted from network infrastructure. That is the essence of SDN simplification.

Keeping the data model and APIs simple and open in your SDN environment is vital to maintaining a vibrant, fast-moving, and lucrative software ecosystem. This becomes the foundation for orchestration and automation of services. These are what really drive dramatic reductions in operational costs.

Adaptable Infrastructure, Flexible Architectures

SDN benefits are also dependent on how SDN architectures and technologies are applied. A network supporting subscribers in a major city might be very different from one serving rural customers. Architectural requirements might also vary tremendously between different countries and the type of service they support. Thus you will need to establish an SDN environment that is flexible enough to enhance your business outcome, however the physical hardware and virtualized infrastructure are deployed. This flexibility is crucial to keeping costs down and responding to new and individual revenue opportunities.

The underlying network resources need to be flexible and adaptable in regard to service instances and where they are located. These resources need to support rapid turnup and turndown of services. These resources include computing, storage, network forwarding, network functions, bandwidth capacity, and physical versus virtual instances. The operations control systems should be able to automatically determine (in real time or near real time) the best placement for the network functions used for the services. They can then meet the needs of the applications in use, while complying with business policies.

Flexible architectures also help you migrate to SDN, because network transformation can happen quickly, but not overnight. Interim migration steps are likely very necessary. There is a huge investment in existing infrastructure, so such equipment (which might have lifecycles ranging from 3 to 10 years) has to be either controlled as SDN components or migrated to an SDN infrastructure. This migration should be mapped out carefully to avoid yet more complexity and its associated extra costs.

In summary, the primary points to remember include:

- Openness, operational simplicity, and flexibility applied across the whole services infrastructure
- Building a flexible service infrastructure that can adapt to the needs of different customers, applications, business models, and geographies

These are hallmarks of SDN that will result in the right business outcomes for you by cutting your costs and expanding your markets.

For More Information

- Declarative Programming
- Automating a New Class of Carrier Cloud Services
- Cisco Application Centric Infrastructure
- Cisco Evolved Services Platform
- SDN for Service Providers
- Network Functions Virtualization

C11-732672-02 06/16