White Paper

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

Software-defined networking (SDN) can help businesses and service providers reduce networking complexity, as well as deployment and operational costs. These benefits are based on abstraction of the control functions from the network forwarding devices (such as switches, routers, and Layer 4 to 7 appliances). Abstraction helps simplify development or modification of new or 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 offers lower 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 promises greater flexibility and a lower cost structure, resulting in higher margins
- The ability to modify or spin up services much faster, 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 markets

Is the Industry on the Right Track?

These SDN economics are indeed tantalizing. They present an opportunity to take advantage of mass-produced, low-cost hardware and much faster software development, using the experience of the web explosion as the benchmark. Service providers, technology vendors, and enterprises are therefore looking at SDN technologies closely.

But what are some practical implementations and lessons learned? How can SDN reduce expenses and make money for providers?

Today, 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. Service providers around the world have to manage a great variety of topologies and geographies. That means they must cope with the economic practicalities of where fiber runs can be made and where network, storage, and computing gear can be located. They 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 measurable reduction in operational costs and complexity. The main focus should be on the desired business outcomes: lower costs, less complexity, and greater agility. The focus should not be on centralizing network functions into big data centers and using mass-produced generic hardware, although this will yield optimal economics in some scenarios.

The problems of cost, rigidity, and complexity can be traced to the control and management frameworks. To date, these frameworks have not included suitable data-abstraction models.

- Step one in reducing cost, complexity, and rigidity is to break up the control and management problem into smaller pieces, otherwise known as modularity.
- Step two is to develop good data-abstraction models that can be easily used across various software modules and layer interaction points. Fortunately, some very good work is already underway in the Internet Engineering Taskforce (IETF). For example, the YANG data modeling language builds a consistent abstraction of network elements and programs them 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) that are defining the various functional modules and interfaces for a virtualized network infrastructure. 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.

Fulfilling the promise of SDN also requires a commitment by service providers to look at their 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, service providers should seriously consider a declarative model between the applications and the network. (Refer to Declarative Programming link under 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, and to be highly resilient in stress conditions and converge quickly upon failure in both the data and control plane.

Developing Your Software Ecosystem

Many service providers are looking with interest at the models that companies like 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. Service providers can use their improved programmability and application interaction to do a similar thing, providing a broad platform that is inviting to third parties and resulting in compelling services and applications for their users.

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 service providers' SDN creates and enforces policies that are used to configure network and service infrastructure. As a result, the 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.

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 may be very different from one serving rural customers. Architectural requirements may also vary tremendously between different countries and the type of service they support. Thus providers must establish an SDN environment that is flexible enough to enhance the business outcome, however the physical hardware and virtualized infrastructure are deployed. This flexibility is key to keeping costs down and responding to new and individual revenue opportunities.

The underlying consumable resources need to be flexible and adaptable in regard to service instance location and allow rapid turn up and turn down of services. (These resources include computing, storage, network forwarding, network functions, bandwidth capacity, and physical versus virtual instances. Thus the operations control systems should be able to automatically determine (in real time or near real time) the best placement for the services, so they can meet the needs of the applications in use, while complying with business policies.

Flexible architectures also help providers migrate to SDN, because network transformation can happen fast, but not overnight. Interim steps are necessary. There already is a huge investment in existing infrastructure, so such equipment (which may have lifecycles ranging from 3 to 10 years) either has to be controlled as SDN components or migrated to an SDN infrastructure. This migration should be mapped out carefully to avoid complexity, which would add costs and defeat the ultimate objective.

In summary, the key 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 - saving money and making money.

For More Information

- Declarative Programming
- Automating a New Class of Carrier Cloud Services
- Cisco Application Centric Infrastructure
- Can Software Defined Networking Enhance Operator Monetization?

- Cisco Evolved Services Platform
- SDN for Service Providers
- Network Functions Virtualization

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at www.cisco.com/go/offices.

Printed in USA C11-732672-01 10/14