Executive Summary

Legacy network architectures are not constructed to meet the requirements of modern enterprises. Government IT CIOs and Leaders face challenges unique to Government Enterprises. Regulations such as FITRA, which mandate the optimization of data centers and IT resources, are one example. Operations can also be a challenge. By some estimates, Government Agencies spend three quarters of their IT budget maintaining Legacy Networks and IT systems. Today's networks must be dynamic, adaptive, and responsive in order to compute, store, and manage application resource needs. Software Defined Networking (SDN) allows organizations to meet rapidly developing requirements-bypassing long procurement and deployment cycles. Because SDN benefits include end-to-end visibility and control, software can be used to intelligently route data across the optimal network paths. This routing of intelligence can be used to build automation tasks that activate or deactivate based on thresholds such as network load, latency and jitter. SDN architectures accomplish this by decoupling network functions (control, data transport, security, and load balancing) from dedicated hardware and specific vendors. The network, which functions as the transport medium, is abstracted from the application and function. Much like a highway system that can support various transport containers; cars, vans, buses, and freight-bearing vehicles, SDN provides the mechanisms for moving data and providing services, without binding it to specific hardware, software, or vendors.

SDN accomplishes this by abstracting the network infrastructure from the applications. The result; unparalleled ability to program, automate, control, and secure the network. Some of the benefits of this include:

- Logical centralized management that eliminates the need to touch multiple devices, with varied operating systems. Additionally, granular control allows for micro-segmentation of policy and security configurations.
- Enhanced Application and Feature Velocity-the capacity to rapidly develop, test, and deploy new applications or enhance existing applications for better user/customer experience and utility.

- Improved cost by reducing Capex and Opex
- Increased security by allowing for security and policy configuration to be vertically integrated and pushed from a central location.

This white paper presents an explanation of SDN, summarizes the state of the industry, explores some key benefits derived from an SDN deployment, and presents some SDN uses cases.

Technology Overview

Software Defined Networking (SDN) has no widely accepted definition. There are, however, generally agreed upon architectural principles that define what a software defined network looks like:

- The separation of the network control and forwarding (packet handling) functions.
- The abstraction of network functions (at all layers) from a dedicated hardware device.
- Programmability from layers 2-7 on the Open Systems Architecture (OSI) stack.
- Abstraction of Application Layer from being tied to network

Additionally, there is general agreement that SDN implementations have these characteristics:

- SDN controllers are independent elements from any forwarding network elements.
- All system elements are based (preferably) on open interfaces and standards.

Figure 1, illustrates these principles and elements:

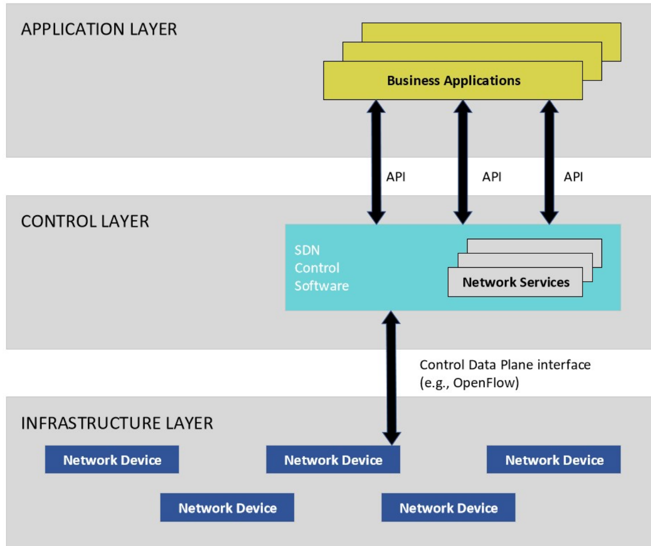

Fig. 1

A software defined network, then, is a network which: has the control and forwarding planes separated, the network functions abstracted from any underlying vendor specific hardware with little or no proprietary protocol or interfaces, and is programmable from the infrastructure to the application layer (2-7 OSI Model).

Benefits of SDN

SDN evolved in response to the limitations of traditional network architectures. Traditional networks are defined by vendor dependent, static network stacks; every role such as security, routing, switching, etc. is performed by a separate piece of physical hardware. This leads to issues such as:

- Complexity in management and change that leads to stasis making it challenging to easily add new features and functions.
- Inconsistent policy implementation due to differences between vendors and user error practically guarantee mistakes or misconfigurations.

- No method for dynamic, real-time scaling of capacity or feature enhancements.
- Integration with higher-layer software (e.g. applications) can be costly and time consuming.

An SDN-based architecture alleviates the above issues and brings a broader list of other potential benefits:

- Logical centralized management that eliminates the need to touch multiple devices, with varied operating systems. Additionally, granular control allows for micro-segmentation of policy and security configurations.
- Enhanced Application and Feature Velocity-the capacity to rapidly develop, test, and deploy new applications or enhance existing applications for better user/customer experience and utility.
- Improved cost by reducing Capex and Opex
- Increased security by allowing for security and policy configuration to be vertically integrated and pushed from a central location.

A brief look at a couple of use cases illustrates some of these benefits:

SDN General Use Case - Data Center Operations

Data Center networks works require the ability to dynamically access network resources, as the hosted applications demand it. As compute and storage resources are spun up, taken down, moved, and reallocated based on user and endpoint demand, the network needs to respond. Traditional network architectures require manual intervention for even the most basic tasks.

A typical task encountered in daily data center operations is the need to move a Virtual Machine (VM) between compute resources. This might require a VLAN move or other network adjustments, such as inter server bandwidth or flow reallocations. A traditional network would require a network operator to manually adjust VLAN assignments, possibly across heterogeneous vendors with different command structures, manually reallocate individual network flow resources, and adjust any security measures (Access Control Lists, etc.). Not only is there a delay related to executing these tasks, but there is a good possibility of configuration errors being introduced. In the SDN model these functions can be automated. The network programmability and logically centralized management functions of an SDN network simplify this task. A network operator can initiate the VLAN and security changes from one place and push it out to all network elements. Unlike management and configuration in a traditional network, a network operator programs all the network changes- VLAN assignments, network flows, and any security requires from on central SDN controller. The SDN control software then makes the necessary changes to the underlying hardware. This reduces the chance of configuration errors or security gaps. Figure 2, shows an automated virtual machine (VM) migration.

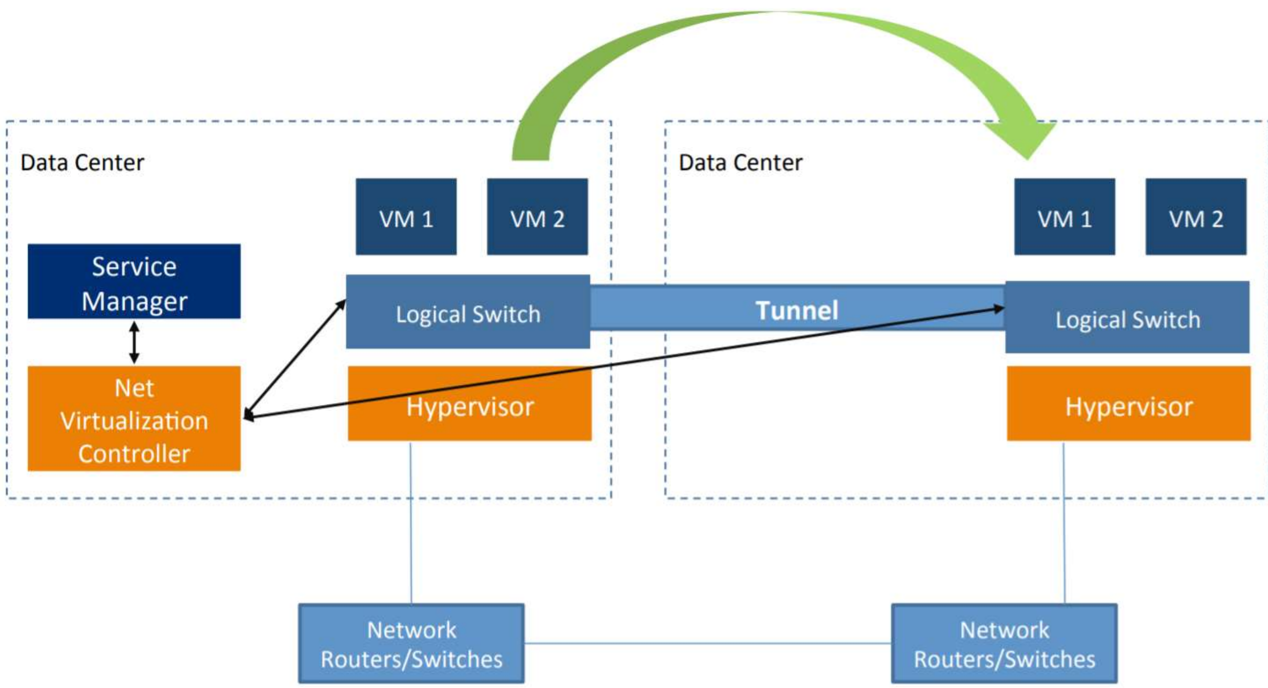

Fig 2.

SDN General Use Case - Big Data Optimization

The explosive growth of Big Data and how it has enabled an organization's ability to collect data has yielded two things: an increase in the amount of information (infrastructure state, application, geo-location, and end-user information) and a dramatic growth in the number of sources for that information. The need to access, analyze, and act on this information has resulted in innovations in the way software, computation, and storage environments operate. Container technology, hyper virtualization, and fully composable software processes store and operate on data more efficiently than ever. Network architectures and operations have not kept pace. Traditional network architectures with fixed resources and which require manual intervention do not have the agility or intelligence to respond to fluid and on-demand resource needs. Traditional networks rely on routing and switching protocols that are aimed at optimizing data paths across networks at a macro level, but what's needed to truly optimize a Big Data environment is to find the right path for each application -- and in a perpetually fluid environment. SDN, which is centrally controlled with fully automated programmability, can respond to the real time data grooming needs of a Big Data environment.

For example, assume an application container has been deployed which needs access to imagery data stored at various storage locations across an enterprise, and needs the information rapidly to make real-time decisions. A traditional network would be unable to respond in real time. Conversely, SDN offers features like affinity networking groups, which allow the network controller to dynamically specify the routes and optimize the paths that lead directly to the storage locations of imagery data, while relocating network forwarding resources. None of this require the manual network operator intervention required in legacy networks.

Conclusion

Software Defined Networking (SDN) is a flexible and dynamic network architecture. It allows organizations to accelerate service delivery, speed application deployment, and stream-line deployment cycles. More importantly, SDN allows an organization to evolve its network into a programmable service delivery platform capable of responding rapidly to changing operational and end-user needs.