ESG Lab Review

Big Switch Networks and Dell EMC: Next-generation Data Center Networking

Date: June 2017 **Author:** Jack Poller, ESG Analyst with Brian Garrett, VP ESG Lab

Abstract

This ESG Lab Review documents hands-on testing of Big Switch Networks' next-generation data center networking solutions using Dell EMC Open Networking Ethernet switches with a focus on demonstrating the value of software-defined automation, simplified operations, visibility, and resilience in VMware, OpenStack, and containerized data center environments. ESG also looked at scalability and performance as well as potential savings through reduced operational expenditures (OpEx).

The Challenges

Networking remains a core "nervous system" of the IT infrastructure, and few IT initiatives will leave the networking infrastructure untouched. Software-defined data center initiatives including private clouds, server virtualization, and hyperconverged infrastructure (HCI)—leveraging VMware SDDC (vSphere, NSX, and vSAN), OpenStack, and containers—workloads are increasing the cost and complexity of configuring, visualizing, and securing the network.

A recent ESG survey of data center networking professionals indicates that the top most-cited three networking challenges were meeting budget constraints (38%), implementing network security (35%), and integration between network operations and other IT domains (29%).[1] Continuing on the path initially forged by server virtualization technology, a growing number of organizations are addressing these challenges with software-defined networking (SDN) and open networking (white-box or branded white-box) switches. When asked about these new technologies, nearly two-thirds of organizations (64%) indicated that they are committed to SDN as a long-term strategy and more than one-third (38%) reported that they have already deployed open networking switches to some extent.

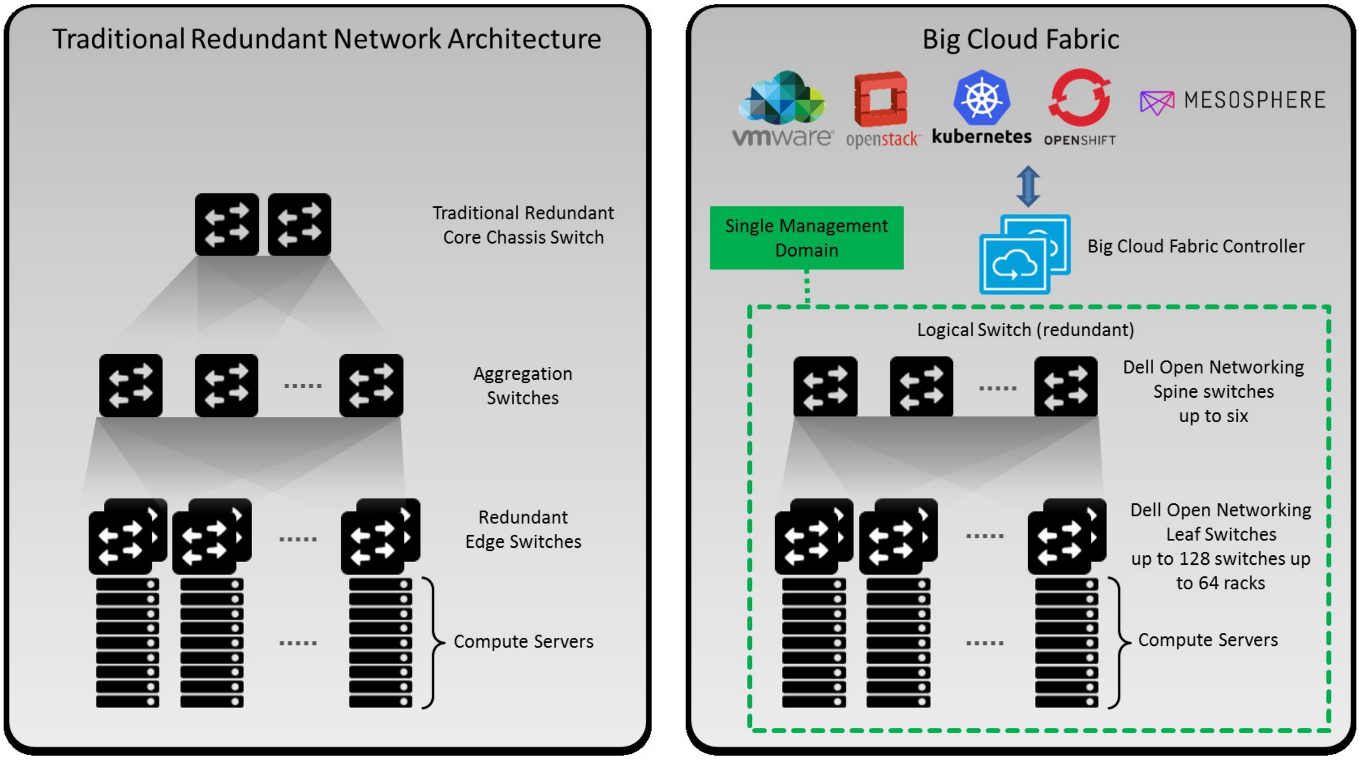
The Solution: Big Cloud Fabric

Big Switch Networks is a next-generation data center networking company that enables data center transformation with SDN controller software (Big Cloud Fabric) running on industry-standard Dell EMC open networking switches with built-in network packet broker technology for centralized network monitoring and visibility (Big Monitoring Fabric). Big Cloud Fabric (BCF) is a networking fabric designed to support physical, virtual, and container workloads and choice of orchestration software, as has become prevalent in the hyperscale data center. The fabric provides layer 2 switching (L2), layer 3 routing (L3), and layer 4-7 service insertion and chaining. Big Cloud Fabric delivers scalability, application agility, and cost reductions through hardware and software disaggregation and automation.

The traditional data center architecture, shown on the left in Figure 1, uses a three-tiered model where network administrators deploy top of rack edge switches, aggregation switches, and chassis-based core switches where each physical component is separately managed. Network components are redundantly cross-connected, which can create loops. Complex L2 and L3 protocols are typically used to enforce loop-free topologies, increasing the difficulty of managing these networks.

[1] Source: ESG Research Report, *Trends in Data Center Networking*, February 2016.

Figure 1. Traditional Redundant Network Architecture versus Big Cloud Fabric Architecture



The Big Cloud Fabric architecture, shown on the right in Figure 1, uses a leaf-spine topology, where all devices are exactly the same number of segments away from each other, and contain a predictable and consistent amount of latency. The spine layer is comparable to the backplane of the chassis switch or router, while the leaf layer is comparable to the combination of the line cards in the chassis router along with all of the aggregation and edge switches. Every physical leaf switch is interconnected with every spine switch (full-mesh). The HA-pair of BCF controllers is comparable to the supervisor cards in the core routers or chassis-based switches, and manages all BCF components.

The Big Cloud Fabric controller cluster, running on redundant industry-standard servers, is responsible for the control, management, and policy of the switching fabric. The switching fabric is implemented as a single management domain using Dell EMC Open Network Ethernet switches running Big Switch Networks' own Switch Light operating system. The fully redundant architecture eliminates single points of failure, and can run without the Big Cloud Fabric controller.

Big Cloud Fabric is available in two editions. The P-Fabric Edition is a leaf and spine physical fabric. The P+V Fabric Edition extends the leaf and spine fabric of physical switches to virtual leaf switches, providing a fully integrated networking fabric solution for OpenStack Clouds and container environments.

Big Switch designed Big Cloud Fabric for the enterprise-scale data center leveraging the design principles of hyperscale data center networks. Enterprise-scale data centers benefit from:

- **Centralized Management**—All BCF components reside within a single management domain, controlled by the BCF controller. Administrators access the BCF controller from the graphical user interface (GUI), the command line interface (CLI), or programmatically with a REST interface. Each change to the BCF fabric is automatically propagated to all physical and virtual switch components as necessary.
- **Streamlined Multi-tenant Configuration**—All configuration is based on the model of logical tenants and network segments, where each tenant has administrative control of a logical L2 and L3 network and policy that connects edge ports under the tenant's control. The BCF controller translates logical tenant configuration into the correct optimized forwarding and policy rules for each leaf and spine switch in the fabric.

- **SDN with HW/SW Disaggregation**—Using Open Ethernet (white-box or branded white-box) switch hardware in fabric designs, and software-defined networking (SDN) concepts to separate the control and data planes eliminates the need for complex, hard to manage, and proprietary chassis switches. This translates into both CapEx and OpEx savings.
- **Scale-out and Elasticity**—BCF's leaf-spine topology enables users to start at the minimum size to meet immediate needs, and to scale by adding additional switches as the environment grows. The BCF controller automatically extends the configuration to the new switches, accelerating deployment and reducing the potential for manually introduced errors.
- **Resilience**—The BCF will continue to work with link failures, node failures, and even in the rare situation where the BCF controller has failed. Swapping a failed switch node is similar to changing a line card in a chassis router. When the replaced switch boots, it is automatically configured by the BCF controller.
- **Workflow Automation and Integration**—The BCF controller exposes REST APIs to integrate with network, security, and audit workflow automation systems. Integration of tenant provisioning with OpenStack HEAT templates enables audit reviews of templates rather than each application instance, simplifying and speeding application deployment.
- **VMware Integration**—Big Cloud Fabric seamlessly integrates with VMware environments to deliver physical networking automation and simplicity to any virtualized workload. BCF provides a single network management console via a tab in vCenter. This API integration between BCF and vCenter is the basis for deeper integration with and networking support for additional VMware solutions, including vSphere, VIO, Horizon, VMware NSX (including VTEP support for end-to-end virtual and physical network visibility), vSAN, vRealize, and vCloud Director for NFV.
- **Container Support**—BCF integration with Docker containers and industry-leading container orchestration platforms (Kubernetes, Mesosphere DC/OS, and Red Hat OpenShift) provides networks automation, deep visibility, and rapid troubleshooting in multi-container and multi-orchestration platform environments.
- **Cloud Management Platform Integration**—BCF controller natively supports VMware vSphere, OpenStack, and multiple container platforms through a single API. Plug-ins for each platform simplify management from traditional box-by-box to automation and self-service models.

Dell EMC Open Networking Switches

Big Cloud Fabric leverages Dell EMC's Open Networking Ethernet switches to provide a full portfolio of switch configurations from a single vendor, simplifying acquisition, installation, and support. Dell EMC Open Networking switches are designed for the enterprise with hot swappable and redundant power supplies and system fans. The various port offerings for each switch are detailed in Table 1.

Table 1. Dell EMC Open Network Ethernet Switches

Switch	Port Configuration	Switching I/O Bandwidth (full-duplex)	
S4048-ON	48x10GbE + 6x40GbE	1.44Tbps	
S4048T-ON	48x10GbT + 6x40GbE	1.44Tbps	
S6100-ON	64x40GbE	2.56Tbps	
S6010-ON	32x40GbE	2.56Tbps	
Z9100-ON	32x100GbE	6.4Tbps	

ESG Lab Tested

ESG Lab performed hands-on testing of Big Switch Networks next-generation data center networking with a goal of validating the agility and simplicity of this new paradigm of network fabric that takes an application-centric view. Testing also focused on scalability and automation, and how that translates to OpEx savings.

Provisioning and Basic Operations

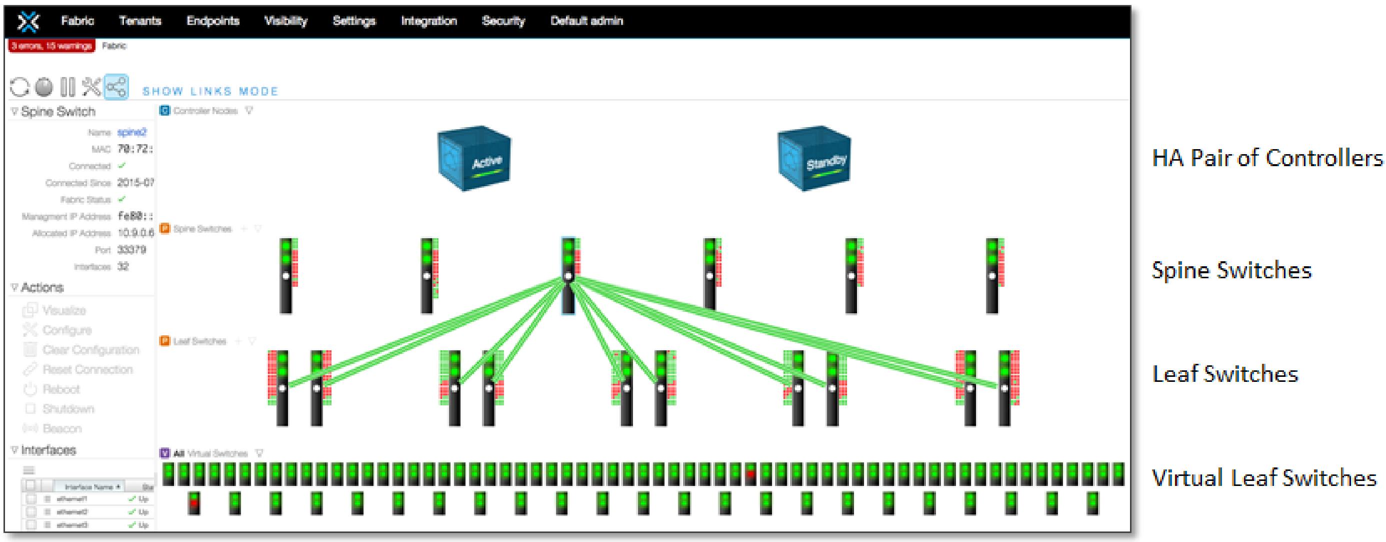
ESG Lab started by using the Zero Touch Fabric (ZTF) feature to configure Big Cloud Fabric on a preinstalled test bench with topology set according to Big Switch Networks' best practices for high availability—redundant active/standby BCF controller appliances, and redundant parallel pairs of leaf switches. The test bench used for evaluating the P+V Fabric edition included an OpenStack environment, including an OS controller, compute nodes with KVM-based hypervisors, and Switch Light Virtual-based virtual switches, running on endpoint servers connected to the fabric.

The first step was to add each Dell EMC Open Networking Ethernet switch to the configuration, and designate the switch role as either leaf or spine node. We logged in to the BCF controller GUI using a standard web browser, and navigated to the ***Fabric*** view (see Figure 2). Each switch in the test bench was added to the configuration by clicking on the ***+***, then entering the MAC address and switch name, and selecting the switch role.

When each switch was powered on, the switch's built-in Open Network Install Environment (ONIE) boot loader searched the network for boot servers. The BCF controller, acting as a boot server, responded and pushed out Big Switch Networks' own Switch Light operating system. Once booted, each switch used link layer discovery protocol (LLDP) to identify links to other switches in the fabric. The entire installation and configuration process completed in less than 15 minutes, and most of that time was consumed by the boot and load process of the individual switches.

The ***Fabric*** view, shown in Figure 2, displayed the fabric topology. The active/standby controller pair was shown at the top, with green bars indicating healthy status. Below the controller was a row for the spine switches and a row for the leaf switches. The health of each switch was displayed by red or green icons. Also shown was the state of each network interface, green for connected, and red for disconnected or error. All of the virtual switches running in the OpenStack environment which are auto discovered using plug-in LLDP, were also displayed.

Figure 2. ESG Lab Test Bench as Shown by Big Switch Fabric controller GUI



The GUI was simple to understand and easy to navigate while providing very powerful features. Selecting a switch provided detailed information about the switch on the left hand side, and displayed the connections between leaf and spine. Hints and additional information were available for all graphical elements by hovering the mouse over the element. Tables and sections could be collapsed or expanded, enabling administrators to quickly find pertinent and critical information.

Next, ESG lab reviewed the current status of the fabric by clicking on the Big Switch Networks logo at the top left to get to the homepage. Figure 3 shows the current state dashboard. At the top left, in red, was a hyperlinked red box showing any fabric errors and warnings requiring administrator attention. The top left of the screen displayed the controller attributes and the inventory of components, including switches, tenants, segments, and endpoints. The top right section provided a quick graphical snapshot of the state of the controller, showing memory, CPU, and disk utilization, along with a switch to flip the controller between active and standby mode.

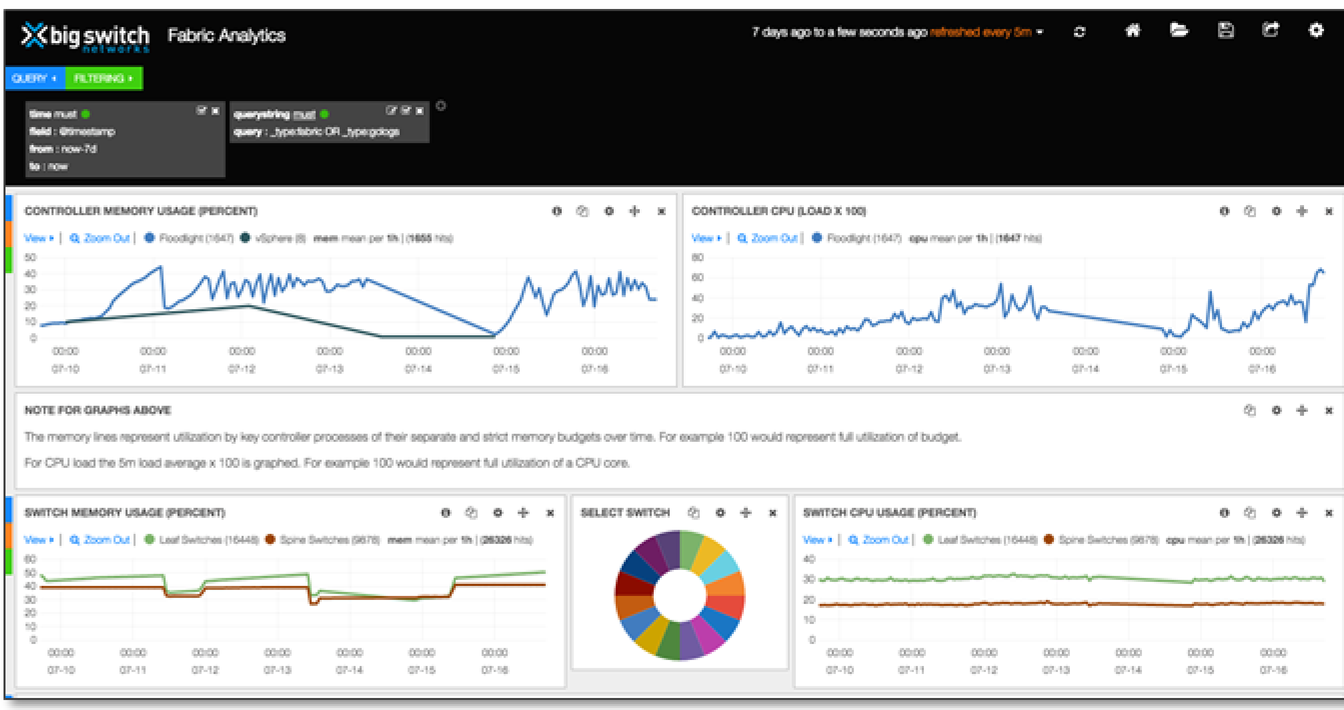
Figure 3. BCF Controller GUI—Current Fabric Status

The rest of the dashboard was split into graphical elements showing the state of various components of the fabric. At the top were two heat maps showing the memory and CPU load for each switch in the fabric, with heavier loads indicated by taller columns and redder colors, while lighter loads were shown with smaller columns and greener colors. Below the switch information, the dashboard displayed information related to network activity, including the top talkers and top usage for tenants, segments, and interfaces. This information enables administrators to quickly focus on areas and components that may need more attention.

In addition to the real-time state of the system shown by the dashboard, the BCF controller includes a detailed analytics module for analysis of historical data and configuration changes. The module includes a number of prebuilt dashboards, and provides the ability for users to create their own dashboards. The prebuilt dashboards provide time-series views of the data, and are categorized by physical or logical data. Physical data includes the fabric, fabric components, and the controller. Logical information includes tenants, segments, endpoints, and virtual integration with VMware and OpenStack.

ESG Lab started the analytics module by navigating to ***Visibility***, then ***Analytics***, which by default displayed the historical state of the fabric, as shown in Figure 4. The information provided in this screen was similar to the current state dashboard, with data graphed over time. Each graph was a module that could be separately configured or moved to any location on the screen.

Figure 4. Big Cloud Fabric Analytics



Displayed at the top were graphs showing the controller memory and CPU utilization over time. Each graph was independent, and time scales could be changed with the simple click or drag of the mouse. At the bottom of the screen were the memory and CPU utilization aggregated over all switches in the fabric. An individual switch could be selected and its data graphed by clicking on a segment in the center color wheel. Just as with the current state dashboard, information on the network activity was also displayed.

The GUI provides a simple click-to-create interface to add new graphs (including columns, lines, sparklines, histograms, maps, etc.) and dashboards. Simple to complex queries and filters can be created to control which data is analyzed.

Why This Matters

IT managers, along with senior executives and line-of-business stakeholders, continue to look for ways to improve resource efficiency and ROI. The challenge is exacerbated by the adoption of virtualization and software-defined data center technologies, and a shortage of professionals with the proper skill sets. With these challenges in mind, a growing number of organizations are moving from a traditional three-tier network architecture to a two-tier leaf-spine network. ESG research indicates that less one in five (17%) organizations leverages a two-tier leaf-spine design today, but an additional 50% of organizations expect to transition away from their existing three-tier architectures going forward.[2]

ESG Lab verified that Big Cloud Fabric required just a few keystrokes and less than 15 minutes to go from bare hardware to a fully functioning leaf-spine network fabric. The GUI and in-depth analytics were intuitive and easy to use, and enabled us to rapidly understand the logical and physical network topology, and to quickly tune the system for optimum performance. By aggregating all networking hardware into a single management domain, Big Switch Networks simplifies the burden on IT infrastructure and staff.

Deploying Applications—Logical Tenants, Segments, and Routers

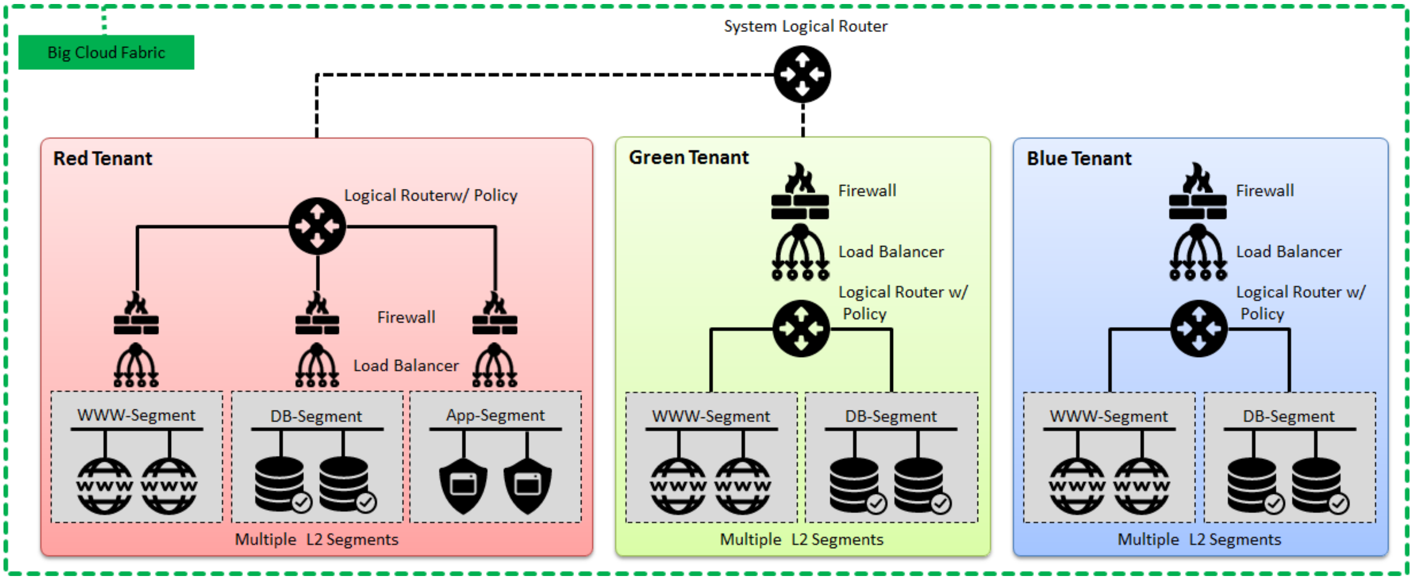
Once the Big Cloud Fabric was installed, the next step was to configure the fabric to enable endpoint (physical server or virtual machine) communication. The traditional 3-tier data center network architecture (shown on the left in Figure 1) requires considerable effort and time to translate high-level logical network designs into box-by-box configurations. Network configuration is slow, prone to errors, and difficult to audit. Dynamic changes, such as automated VM migrations to different physical servers, cause changes to the logical network, requiring more manual box-by-box configuration updates. Scaling the fabric requires even more configuration changes, and introduces the possibility of a fabric shutdown due to human error.

Big Cloud Fabric uses a logical abstraction based on the concept of tenants, logical segments, and logical routers to simplify network fabric configuration and maintenance. A tenant has administrative control over a logical L2, L3, and network policy design that connects endpoints under the tenant's control, as shown in Figure 5. A segment is logical L2 domain, usually implemented as a VLAN. The BCF controller automatically translates the logical network design for each tenant into the appropriate configuration across all leaf, spine, and virtual switches in the fabric. This frees administrators from the tedious and error-prone manual translation and box-by-box configuration process. Simultaneously, all tenants and tenant segments exist on all relevant physical and virtual leaf switches, ensuring logical endpoint connectivity regardless of physical location in the network fabric.

Figure 5 shows the logical configuration of three tenants. The red tenant is a typical three-tier web application with web servers, database servers, and application servers. The tenant includes three segments, one for each tier. Endpoints on the same segment, equivalent to an L2 VLAN, can communicate with each other. Communication between segments uses L3 routing through a tenant-specific logical router. Communication between the red and the green tenant also uses L3 routing, using the predefined BCF system logical router.

[2] Source: ESG Research Report, *Trends in Data Center Networking*, February 2016.

Figure 5. Logical Representation of Tenants, Segments, and Endpoints

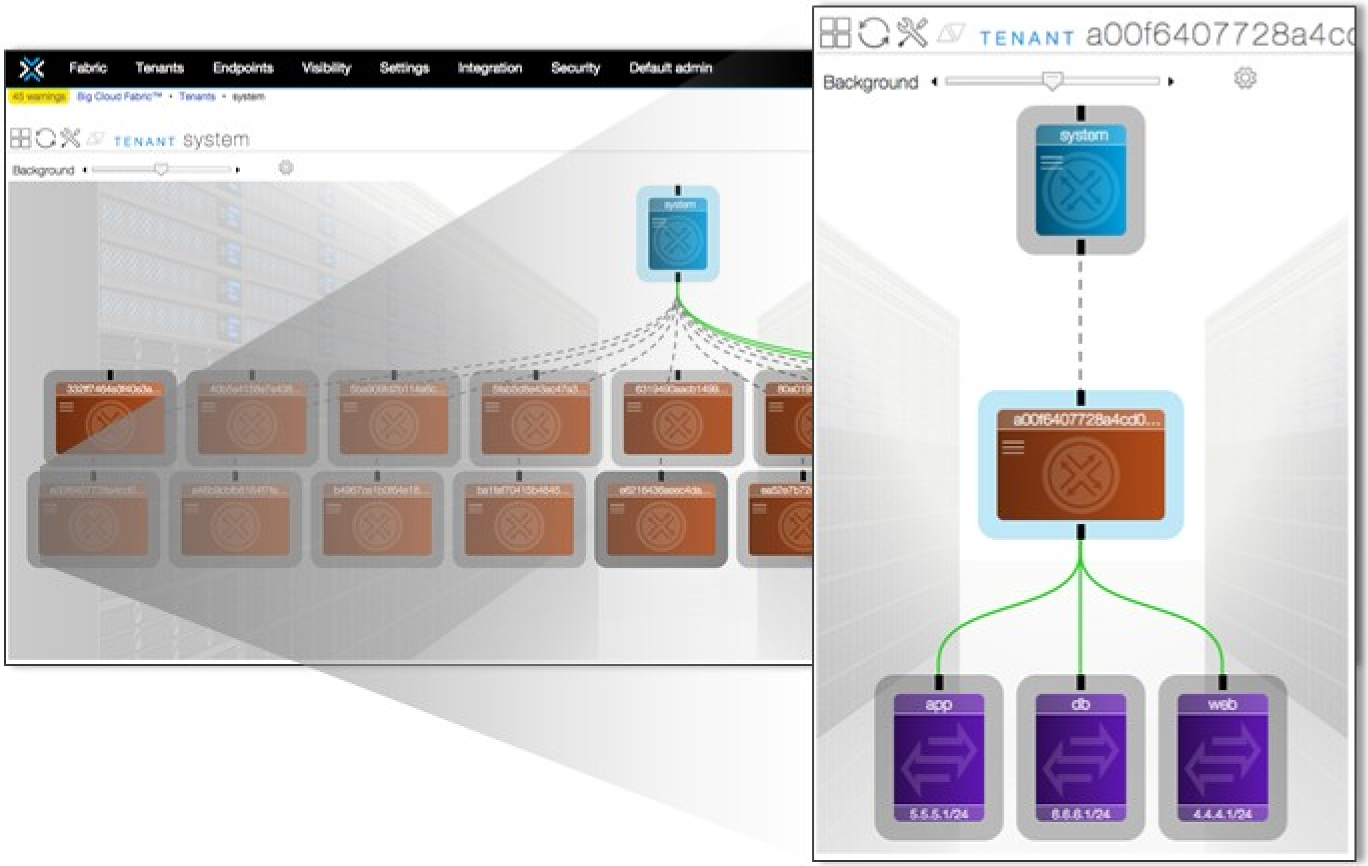


First, ESG navigated to ***Tenants*** to display the list of existing tenants. Next, clicking the ***+***, ESG created the red tenant by specifying the name. The GUI displayed the details for the red tenant. Creating the three red tenant segments—WWW-Segment, DB-Segment, App-Segment—was accomplished by clicking the ***+*** on the logical segment list, and providing each segment name. Next, using the same simple interface, we created a logical router interface for each segment, specifying the IP address information for the segment. Finally, we added routes to the red tenant logical router to route between each segment. The same process was used for the green and blue tenants, and in a matter of minutes, we had configured the entire fabric. Finally, we added routes to the system logical router to connect the red and green tenants.

The BCF controller automatically interprets the logical concepts of tenants and segments to create the appropriate optimized configuration for each switch in the fabric. The configuration ensures that the logical constructs are available across the entire fabric, regardless of the physical implementation. Thus, endpoints can be physically attached to any leaf router, and will automatically connect to the appropriate tenant segment without manual configuration by the administrator.

Next, ESG Lab navigated to ***Tenants***, then ***System***, and the GUI generated a graphical view of all of the tenants instantiated in the Big Cloud Fabric, as shown in Figure 6, which included each tenant's connection to the system logical router if the tenant was so configured. Clicking on a tenant displayed the details for that tenant, which was a three-tiered architecture similar to the red tenant. Each segment, shown in purple, displayed the IP address range for the segment, and connections to the tenant logical router.

Figure 6. Graphical View of Tenants and Tenant Details



The simplicity of the BCF controller GUI was demonstrated by the automatic laying out and routing of the tenant and segment mapping. These graphics enabled ESG Lab to simply and quickly gain understanding of the logical configuration of the fabric and the tenants without having to read and interpret command line configuration statements, or navigate through each configuration page in the GUI.

Why This Matters

Network architects face significant challenges when implementing logical network designs. The manual process takes significant time, effort, and expertise, and each physical switch must be individually configured.

ESG Lab verified that BCF automatically translates the logical abstraction of tenants, logical segments, and logical routers into the correct optimized configuration for all affected leaf and spine switches in the network fabric. Configuration changes were automatically applied, eliminating the switch-by-switch, time-consuming, and error-prone process of the traditional network fabric environment.

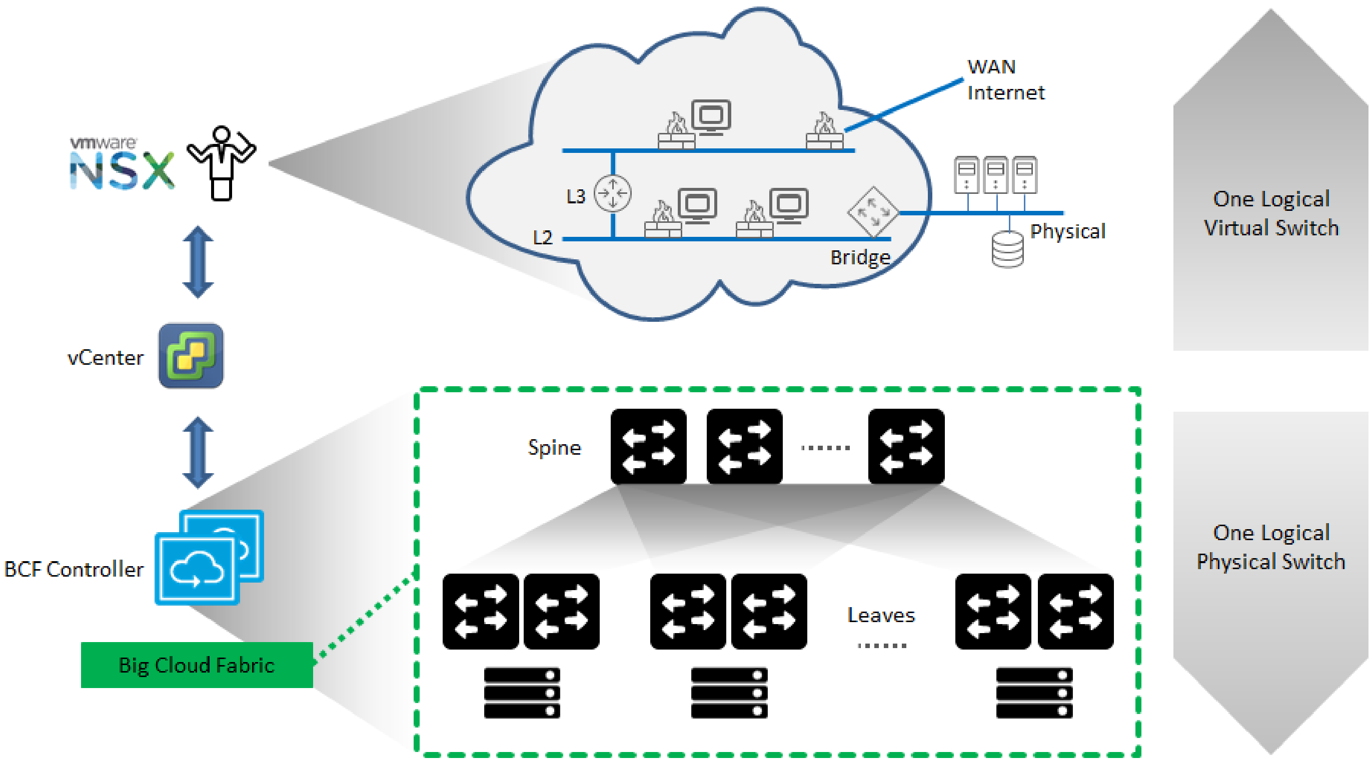
Cloud Management Systems Integration

The simplicity of the BCF GUI became even more apparent with Big Cloud Fabric's integration with orchestration software. The P-Fabric edition supports VMware vSphere and NSX, and OpenStack. The P+V Fabric edition was designed for OpenStack cloud environments and is fully integrated with OpenStack, supporting Neutron ML2 driver and Neutron L3 plug-in integration using Big Switch Networks' Switch Light Virtual for KVM-based hypervisors.

With the BCF P+V Fabric edition, L2 isolation and L3 routing are implemented within the BCF. NAT (floating IP and PAT) is performed on the virtual switch, providing NAT resiliency. BCF provides full visibility across the physical and virtual fabrics for network statistics, analytics, and end-to-end troubleshooting.

The P-Fabric edition of BCF includes a VMware vSphere extension to integrate and automate fabric configuration, maintenance, and troubleshooting. The fabric can be further extended and integrated with VMware NSX, as shown in Figure 7. The result is a single network fabric with one logical physical switch (BCF) and one logical virtual switch (VMware NSX).

Figure 7. BCF Integration with VMware vCenter and NSX



The fully integrated solution provides fabric automation and visibility, including:

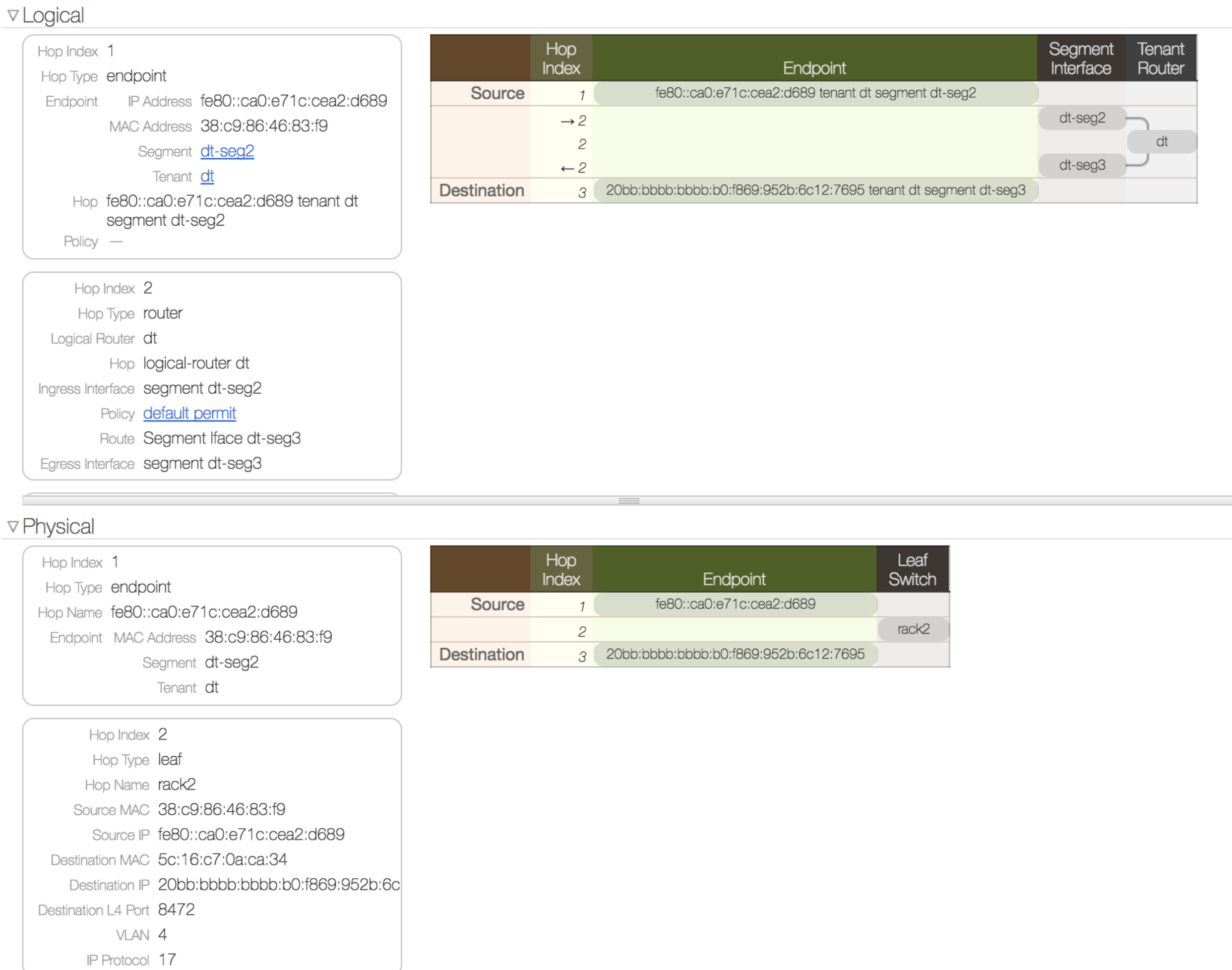
- Auto host detection and LAG formation.
- Auto network creation for VTEP Port-Group & VTEP Discovery.
- Troubleshooting—VTEP-to-VTEP connectivity.
- Visibility through fabric analytics (VM-name, VXLAN ID, and Logical Switch).

Troubleshooting and Resilience

When communications fail, either because of physical component failures or misconfigurations, diagnosing the root cause of the failure can be difficult—significant complexity is hidden behind the integration with orchestration software, and the tenant and segment abstractions. Big Cloud Fabric includes diagnostic tools to aid the system administrator with debugging.

ESG Lab used the built-in test path tool to trace the communication path between two endpoints in a BCF P+V Fabric, a technique used to diagnose a communication failure. First, ESG Lab used the GUI to navigate to ***Visibility***, then to ***TestPath*** to start the test path tool. Next, we selected source and destination endpoints from dropdown lists, and started the path tracing. The path trace result is shown in Figure 8.

Figure 8. BCF Diagnostics: Test Path



In addition to introducing resiliency through the automatic configuration of all physical switches, drastically reducing the potential for human error, Big Switch Networks' best practices include using redundant switches and controllers. With this redundant topology, each leaf node is comprised of two interconnected switches. Each endpoint is connected to both leaf node switches, with the endpoint interfaces configured as a link aggregation group (LAG). Should one switch in the leaf node pair fail, endpoints are accessible through the other switch.

The BCF controller was also designed to run as an active/standby pair on a cluster of two controllers. Should the active server fail, the standby can take over. Because the controller is only responsible for configuring the switches, and is not involved in moving packets through the network, the fabric can run headless. Thus, the network will continue to function even in the rare case that both controllers suffer catastrophic failures.

ESG Lab reviewed Big Switch Networks' demonstration of the resiliency of the Big Cloud Fabric. The resiliency test configuration of BCF included 32 leaf switches, six spine switches, and a two-server controller cluster. The environment contained 42,000 simulated VMs running the Hadoop Terasort benchmark on Mirantis OpenStack.

To stress the environment, Big Switch Networks used the Chaos Monkey test methodology to force random components to fail while measuring the benchmark performance. Chaos Monkey tests were configured to fail the BCF controllers every 30 seconds. Random switches were forced to fail every eight seconds, and random links in the fabric were failed every four seconds. Despite more than 640 component failures during each seven-minute benchmark test, no discernable impact on performance was observed.

Upgrading the BCF Fabric

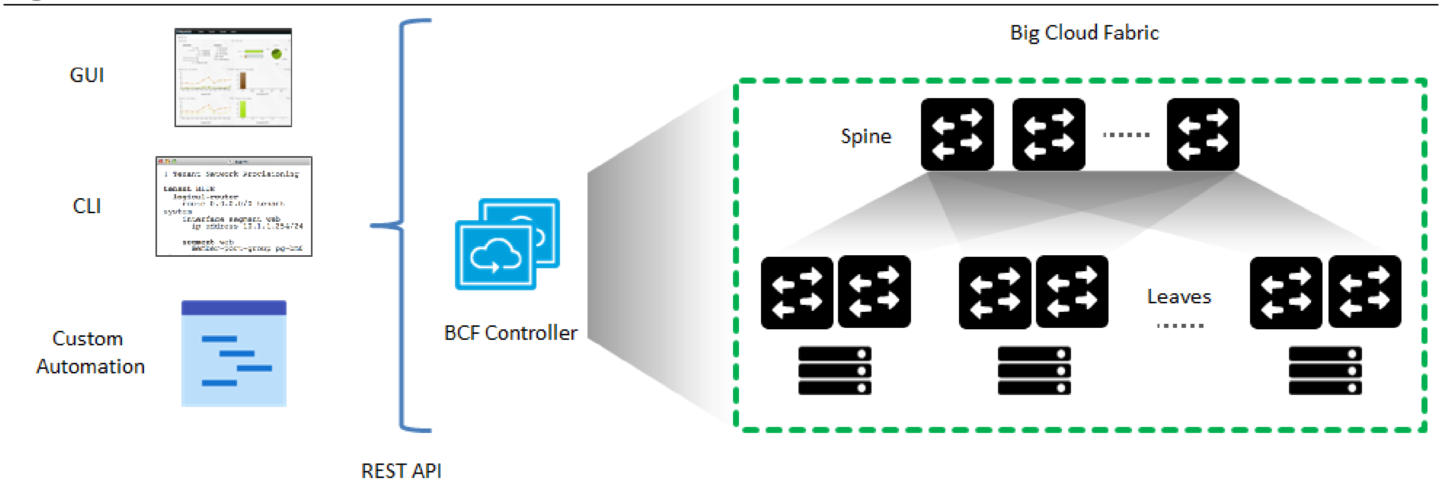
The redundancy of the fabric not only contributes to fabric resiliency, but it also simplifies and hardens the software upgrade process. In the traditional network environment, upgrading software is a fragile manual process, and upgrades are performed box-by-box—manually or using custom scripts. The administrator must first save the existing box configuration, then download the new software to the box, apply the software upgrade, and finally reapply the configuration. This process must be repeated separately for each box in the environment.

In contrast, Big Switch implemented a Zero Touch software upgrade. Once the administrator initiates the software upgrade, the process is entirely automatic and coordinated between the controllers. First, both of the controllers automatically and intelligently partition the network in half, ensuring that all endpoints are able to communicate even when one of the partitions is shut down. Next, the standby controller is automatically upgraded. Then the controller roles are flipped, and the upgraded standby becomes the active controller. Next, the controller forces a reboot of each switch in one partition. The switch reboot forces the switch to load a new version of the Switch Light OS from the controller, and to automatically receive an updated switch configuration from the controller. After the first partition has been completely upgraded, the switches in the second partition are rebooted, for software upgrade. Then finally, the now standby down-rev controller is upgraded. During the entire upgrade process, the network fabric continues to work, and there is no application downtime.

Programming and Customization

The Big Cloud Fabric controller provides a standard REST API for system management. All Big Cloud Fabric actions—installation, configuration, and upgrades—whether executed by the GUI, through the command line interface (CLI), or through automated scripts, are sent to the controller over a standard REST API, as shown in Figure 9.

Figure 9. BCF Automation



Administrators can leverage both the GUI and the CLI interface to accelerate development of automation scripts. From the CLI or the GUI, administrators can enable a debug mode, which displays each REST API call made as the result of a user action. This enables rapid script development by manually executing a sequence of commands, and using the REST API calls as the basis of the script. To further ease scripting and automation, Big Switch provides a set of Python libraries, which ESG Lab used to create scripts for common BCF management operations.

Why This Matters

Virtualization and cloud management platforms rely heavily on automation, reducing the administrative workload and enabling the modern self-service IT infrastructure. Automation becomes more critical as networking and data center environments grow in size and complexity. Yet IT staff are challenged to manage the networking infrastructure with the same level of simplicity and efficiency as they manage the rest of their data center environments, including servers and storage.

Big Switch Networks architected BCF for the modern enterprise data center. BCF integrates directly with VMware vSphere and NSX, and OpenStack, enabling administrators to extend the network fabric from the physical into the virtual and simplifying configuration and management. Further integration is achieved through automation—all BCF management functions can be accessed through the REST API.

Big Switch Networks demonstrated the resiliency of BCF using the Chaos Monkey methodology, where there was no impact to performance after introducing more than 640 component failures during the course of a seven-minute test. Leveraging redundancy, the entire network fabric software can be upgraded via a single command without interrupting normal operations. With resiliency, automation, and integration into cloud management platforms, BCF reduces the burden on IT infrastructure and staff.

The Bigger Truth

Network infrastructures are becoming more of a challenge to manage thanks to the ubiquity of server, application, and desktop virtualization. Providing agile, elastic network services to all applications and users is a huge challenge due to the dynamic nature of today's highly virtualized data center. This problem is exacerbated as networks expand to accommodate private and public cloud infrastructures.

The modern data center, where many systems are running on virtualized infrastructure, present multiple networking challenges to IT organizations, including deployments and ongoing management. Using traditional networking devices and tools in modern data center topologies can be quite challenging, since those tools were designed for physical, relatively static infrastructure. Traditional data center network topologies built with chassis-based core switches and independent top of rack distribution switches can make every aspect of data center networking, from deployment to ongoing management, needlessly complex and difficult. A network designed with hyperscale principles in mind, which deploys and performs with the same flexibility and ease of management as server and application virtualization, will be required as public and private cloud solutions mature and become ubiquitous.

Big Switch Networks has developed a next-generation data center networking solution that delivers on that promise. With Big Cloud Fabric, enterprises and cloud service providers can rapidly create software-defined, multi-tenant networks without the management overhead that exists with traditional routers and switches. These multi-tenant networks are built using Dell EMC Open Networking Ethernet infrastructure, significantly simplifying the network's physical topology and reducing CapEx.

In hands-on testing, ESG Lab was able to deploy a complete Big Cloud Fabric in 15 minutes. The GUI was intuitive and easy to use, providing quick access to current and historical states. ESG Lab also used the GUI to rapidly configure three tenants with their segments and logical routers. The BCF controller automatically translated the logical network topology of each tenant and applied the appropriate optimized configurations to all switches in the fabric, removing likely human error and accelerating the configuration process. This ease and speed of management has the potential to significantly decrease OpEx.

ESG Lab reviewed a Chaos Monkey demonstration of the BCF network fabric resiliency, where more than 640 failures were introduced into a fully configured fabric over a seven-minute test with no discernable change in performance. Using BCF diagnostic tools, we traced the logical and physical network paths through the fabric, aiding in root cause analysis of communication failures.

BCF is managed through a REST API, and includes a Python library for rapid script development. Using the GUI and CLI, ESG Lab executed a sequence of commands, then used the REST version of those commands and the BCF Python library to create automated management scripts. This ability to swiftly develop management scripts for the BCF is critical for the modern enterprise data center, where management and scalability are achieved through automation.

If your organization is struggling with the cost and complexity associated with building and managing a traditional physical network to meet the needs of a next-generation software-defined data center (e.g., VMware and OpenStack environments, vSAN-enabled hyperconverged infrastructure, and Docker containers), then ESG Lab recommends that you seriously consider the benefits of a next-generation data center networking strategy powered by Big Switch Networks and Dell EMC Open Networking Ethernet infrastructure.

www.esg-global.com contact@esg-global.com P.508.482.0188