BROCADE

WHITE PAPER

Software-Defined Networking in the Campus Network

Centralized applications to control the proper functioning of networks have been available for years. These applications cobbled together various mechanisms to control networking devices. Each networking vendor required a custom plugin for the application to work, and—in some cases—multiple plugins for a vendor were required, due to differences within vendors' product lines. New software releases from a networking vendor generally broke the application, requiring updates to the application. These applications were expensive to purchase, due to the heavy cost of development and maintenance.

SDN resolves this issue by defining interfaces between the centralized controller and the networking devices. Regardless of any software change on either side, as long as the updates remain true to the defined interface specification, they are transparent. This enables faster development of reliable, long-lived applications, agnostic to the devices on the network. The impact is an emergence of a new set of cost-effective SDN applications that fine-tune network resources to meet the demands of 21st-century users and their applications.

On-Demand Campus Network Resources Provisioning with SDN

Traditional network environments require application-specific policies such as security and access control, Virtual Local-Area Network (VLAN) traffic isolation and Quality of Service (QoS) to be provisioned across the network one switch at a time. This consumes a significant amount of resources and results in a static network that cannot be easily updated as business requirements evolve or new applications need to be deployed.

In contrast, SDN-enabled networks can dynamically allocate network resources in real time to meet the needs of running applications. Custom-built or prepackaged SDN applications running on the SDN controller can use input from many sources—including predefined application-specific security and QoS requirements, physical network statistics, user activity, security threat analysis, and so on—to allocate and protect network resources, set access control rules, and prioritize traffic in a fully dynamic fashion.

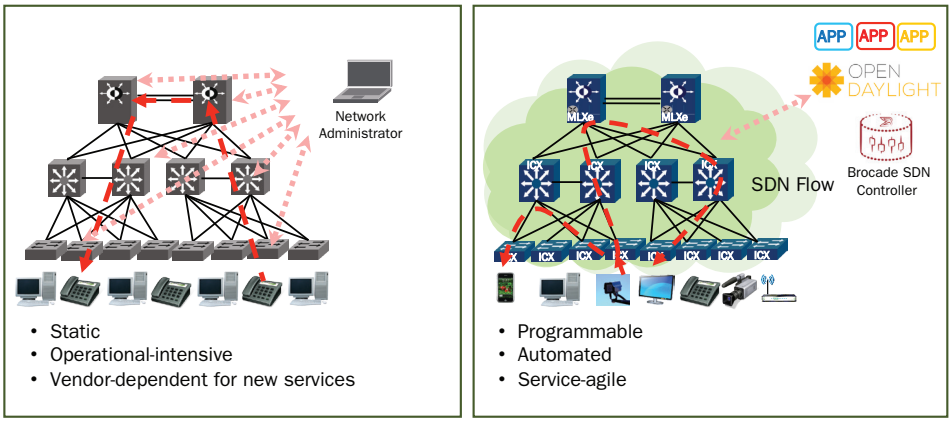

Figure 1. Traditional campus network versus SDN campus network.

Campus Networks in the 21st Century

The flow of information in the campus network is the lifeblood of an enterprise. It needs to transit undisturbed, while at the same time be monitored to ensure security and integrity. User applications have evolved from e-mail, printing, and file transfers to time-sensitive video and audio communications, real-time imaging, and Big Data massive-scale transfers.

Legacy architectures and networking protocols have reached a breaking point as they try to meet agility and flexibility requirements. New architectures and new protocols are needed to ensure the secure, free flow of information on the campus.

The Brocade® HyperEdge® Architecture (discussed in separate white papers) enables a new networking topology to meet the physical demands of delivering data. SDN has become a key enabling technology of the HyperEdge Architecture. OpenFlow, running

on Brocade switches, can either affect all traffic on a given link or work in tandem with traditional protocols that utilize the Brocade hybrid per-flow mode. This enables manipulation of specific flows on a link, while letting other flows utilize the normal packet processing pipeline.

SDN enables you to tailor network traffic in a new way. The SDN application can take input from many sources: physical network statistics, users logging on or off the network, high-priority user applications started and stopped, anticipated traffic due to historical trends, security threat analysis, and so on. This information can then be used to make decisions on how best to respond. The SDN controller, based on the network map it generates, makes the decision on where in the network to push these changes. This can be as simple as an increase in the priority of a specific application flow to full-blown network access control with role-based resource allocation. All of these operate with each layer independent to each other, a defined Application Programming Interface (API) between the layers, and a predefined messaging set. This allows for a robust set of functionalities that scale with the capabilities that each layer reports to one another. Unlike in the past, an update in one layer does not break the functioning of another layer.

Brocade platforms are standardizing on OpenFlow version 1.3—with the associated increased functionality, high availability, and security capabilities over version 1.0—for a robust enterprise solution. Some vendors have decided to implement OpenFlow using CPU-bounded software processing and forwarding of traffic flows. In order to ensure the highest performance level possible, Brocade implements the flow processing all in hardware.

The revision level of OpenFlow (version 1.3.00 vs. 1.0.00) defines the complete set of capabilities that each component (application, controller, and switch) is potentially capable of supporting. The discovery capability that is built into the OpenFlow protocol allows for a piecemeal upgrade of any component without breaking compatibility. This enables any Brocade device, whether it is a Brocade ICX® Series switch, a Brocade MLXe Series router, or a Brocade VDX® Series switch, to interoperate with the Brocade SDN Controller (a distribution of the OpenDaylight Project), or any other SDN controller running the same version of OpenFlow. In addition, as functionality supported by this particular version of OpenFlow is enabled on a device, the new functionality seamlessly integrates with the Brocade SDN Controller and the applications running on the controller.

Based on real-world experiences, Brocade pioneered and enhanced the implementation of OpenFlow with the hybrid per-flow mode. This enables you to mix OpenFlow and standard packet processing pipelines, not only on a per-port basis, but also on a per-flow basis. Thus, the Brocade solution selectively unlocks the power of SDN on the campus for specific user or application traffic, while continuing to use the standard packet pipeline for switching and routing the rest of the traffic on the network. This hybrid-mode OpenFlow capability is available on Brocade ICX switches for the access and aggregation layer, and on Brocade MLXe routers for the core, all of which work together to deliver an end-to-end SDN solution.

The following sections of this paper cover a number of potential use cases of SDN in the campus. This discussion does not present a comprehensive list of all possible uses, but it provides an overview of the potential opportunities that SDN opens in the campus. The SDN applications may or may not currently be available or under development. They may be supplied by third-party vendors, not necessarily by Brocade.

Application-Based Resource Allocation

Traditionally, an application could easily be identified by the IP port number in a packet header. These could be statically configured, with a select group of applications given higher precedence over the network traffic of other applications. A Voice over IP (VoIP) phone would generate Session Initiation Protocol (SIP) traffic, and that SIP traffic could be given high priority through the network.

More often, network administrators are moving away from multiple networked devices on a desktop to one device: primarily a laptop, but possibly a docked tablet or a soft phone. At the same time, more and more applications are moving from dedicated apps to integration into HTML using HTTP as the transport of choice. Those two trends, when taken together, make the old way of a static Access Control List (ACL) entry unusable. The packet IP port information is the same for low- and high-priority traffic, all hidden with HTTP port 80 traffic. Also, the source MAC address is the same, since there is no way to differentiate a VoIP phone from a laptop, and the soft phone traffic and Web surfing appear the same. One option in this case is to trust the Type of Service (ToS) and Differentiated Services Code Point (DSCP) value of the packet. However, that assumes that the applications on the device can be trusted to apply the appropriate value, which could allow for malware to flood a network with bogus high-priority traffic. A new mechanism is needed to identify which specific flows should be treated with high priority, with all others defaulting to low priority. Using SDN with hybrid ports on the access switch can solve this problem.

Microsoft Lync is an example. Consider a simplified description of the process that occurs in making a call. A user with a headphone connected to a laptop brings up Lync with a Web interface, which asks to make a call to a fellow employee from the contact list. The Lync call controller converts this contact information to an IP address. The Lync call controller then sends this IP address to the Lync client running on the laptop. A call is started between the contact's IP address and the laptop's IP address. There is nothing specific in the packet to indicate it should have higher priority than other traffic. This is where SDN comes into play.

In an SDN environment, as the Lync call controller is sending the IP address to the Lync client in the laptop, the Lync controller can be configured to also send to an SDN application, whose function is to talk to the SDN controller and have the priority set to specified values for specific IP pairs in a network. A Lync call, for instance, would be set to a high priority. The SDN application communicates to the SDN controller that the priority level for a specific IP pair needs to be set to high and passed through non-congested links.

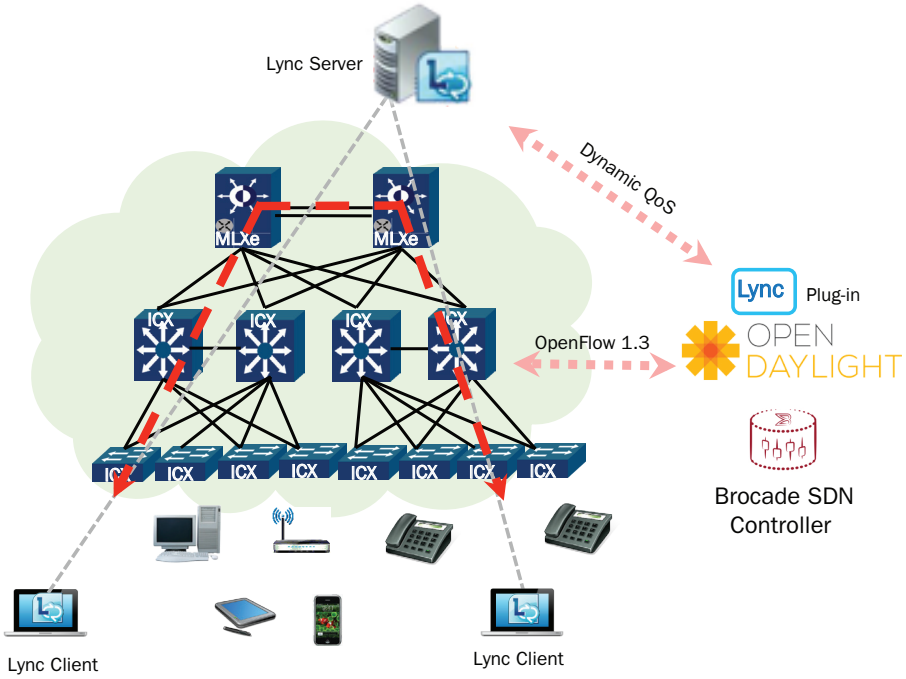

Figure 2. SDN-based adaptive and automated QoS.

The Brocade SDN Controller takes this information, and—based on the topology it has created from the information received from all the OpenFlow-enabled devices on the network—determines the optimal OpenFlow path. This path would start from the location that the packet enters through to the location where it exits the network, in both directions, and would determine the optimal path for the call packets to flow through the network. This may require directing packets out links that are different than the traditional pipeline would have selected. This flow matching information, along with the required actions, are pushed to each of the OpenFlow-enabled switches. Assuming the call is within the same organization, these entries are on the Brocade ICX 6610 Switch (an access switch), which both users are connected to, as well as on the intervening Brocade ICX 6610 and Brocade MLXe devices in the network. When the call ends, the OpenFlow entry ages out due to inactivity, freeing up resources for other calls or high-priority application flows.

An interesting enhancement to this monitors some application flows and not others, in this case VoIP calls. In addition to the Lync call manager sending the flow information to the SDN application, a second source could send a list of source and destination IP pairs to monitor. When the Lync information is received, the priority path through the network is set up (if the IP pair is on the watch list). In addition, a copy packet action can be added, along with the set priority action, to have a copy of the call sent to a security device. There is no need to know how the mirroring function works on a switch or to collect more information than needed. You simply utilize the OpenFlow protocol to implement the task.

This new SDN capability takes advantage of the knowledge of flow setup by applications running on top of the network. These supply the information needed to identify flows that need to be prioritized or given extra bandwidth through the network. As long as all layers are on the same version of OpenFlow, enhancing devices in the network do not require a rewrite of applications.

Role-Based Resource Allocation

It is often helpful to configure what users can or cannot do on a network, based on the role they play within an organization. You can do this type of configuration by using the IEEE 802.1X protocol. The access switch acts as the authenticator, relaying information to a Remote Authentication Dial-In User Service (RADIUS) server. The RADIUS server in turn authenticates the username and password that was sent. If the user is given authorized access, then the RADIUS server, using Vendor-Specific Attribute (VSA) fields in the RADIUS response, sends the information to the switch to configure the resources to allocate to the user. This tightly couples the higher-level authentication application (the RADIUS server) with the network layer device. If the capabilities of the access device change, the RADIUS server entry needs to be updated. This can be overlooked, potentially causing end users to not get authenticated. Even worse, users could be authenticated, but appropriate security constraints on that user might not be supplied. This is the case for any Network Access Control (NAC) tool that requires the back-end application to have the knowledge of the edge networking device and be configured with that device's specific supported features.

SDN and OpenFlow offer a hardware- and software-independent abstraction model to access and manipulate resources. The process works by bypassing product-specific functions and using only the authentication yes/no functions between the user and the NAC application. When the authentication process is complete, a message is sent to the role-based resource allocation SDN application. It contains the MAC address of the user, the port of entry in the network, and the role of the user. The application then compares this role with the previously configured capabilities list (that is, which users this user can communicate with on the network, which VLAN should be assigned to that user for the network, how much bandwidth the user can have assigned to its traffic, which IP addresses are off limits, and so on). These capabilities are converted to a network resource message that is sent to the Brocade SDN Controller. The Brocade SDN Controller then identifies the appropriate networking device to which to send the OpenFlow table with the appropriate resources allocated (that is, a priority setting for traffic, bandwidth for flows), along with limitations to drop flows to restricted addresses. This is then set on the Brocade ICX 6610 access switch port as the NAC server finishes authenticating the user on that port.

Such a process enables you to enhance the edge device without touching the back-end systems. As long as the edge device supports the same revision of OpenFlow as the Brocade SDN Controller, an update is completely transparent to the role-based SDN application and the back end. Once the user has been allocated the resources, standard packet processing pipelines can be used for switching and routing packets through the network. This leverages the ability for Brocade to utilize an OpenFlow table to allow or deny packets coming into the network and to mark the packet for specific resources (then to drop into the normal packet pipeline to make forwarding decisions). Or, for select users, the normal pipeline is bypassed, and traffic is directed out specific ports.

An example of this second type of environment would be technical users who have a high-performance server on their desktop. Such users have been authenticated as trusted and now are allowed to have all traffic to and from their server treated as secure. This traffic is allowed to transit through the campus network and out a high-performance Wide-Area Network (WAN) link on a Brocade MLXe router, without the need to transit through a firewall. All other users who do not authenticate in this secure state are sent using the standard pipeline through a firewall connected to the Brocade MLXe router. In this case, as the Brocade SDN Controller sends the OpenFlow table to the Brocade ICX access switch, it also sends OpenFlow tables to intervening Brocade ICX switches and Brocade MLX® switches. This creates a guaranteed bandwidth pipe to the Brocade MLXe router that is acting as the WAN router, where an entry is set to send all traffic from the user out the port that bypasses the firewall.

Classic NAC systems exercise control only on access layer switches, leaving the rest of the network untouched. In contrast, a Brocade SDN Controller maintains a holistic mapping of the entire network and the resources attached to it, enabling SDN applications to quickly apply access and traffic policies throughout the network. By taking advantage of Brocade ICX switches at the edge and access layer and Brocade MLXe routers at the core, a consistent set of functionality beyond OpenFlow enables you to segregate priority user traffic from "regular" user traffic. This enables you to take advantage of the normal pipeline and the associated standard mechanisms, or for a full OpenFlow-only mode on a per-user flow basis. This functionality proves to be a truly powerful optimization for utilizing network resources.

Network Access Control

The previous use case, "Role-Based Resource Allocation," demonstrated the use of a classic NAC system for user authentication, combined with the use of SDN to enable user-specific access policies. The next logical step is to implement a complete NAC system with SDN. This provides the flexibility to offer multiple authentication methods to support a variety of devices.

The OpenFlow standard enables you to send a packet from the Brocade SDN Controller out a specific OpenFlow-enabled port on a switch. The standard also enables you to create an OpenFlow table entry to have any packet received on a port sent to the Brocade SDN Controller. These two features are the building blocks to create a strong NAC solution. The NAC application can try various authentication mechanisms by sending out packets to mimic protocols such as IEEE 802.1X, MAC authentication, or Web authentication, and can utilize the protocol supported by the device connecting to the port.

For instance, you might have a set of devices that need to access a secured network but that do not have the ability to respond to NAC user/password queries. These might be print servers, access points, security card readers, and so forth. The MAC address of each device can be used as the authentication mechanism, with the vendor and device type gleaned from the MAC address and compared to valid devices that are allowed on specific ports or a range of ports on specific switches (this would be preconfigured). In such a case, preconfigured rules would be sent as OpenFlow table entries to that switch for that port. This allows the device access to the network, but limits the type of traffic that is allowed access to the network. For instance, there is no reason for a card reader to browse the Web or do port scans.

If the MAC address is not trusted, then try to authenticate the device by sending out a crafted 802.1X "session initiate" request out the port, using the packet insert function. If the device responds back with the appropriate 802.1X response packet, then the packet is captured and sent to the Brocade SDN Controller over a secure OpenFlow link, and then it is sent to the NAC SDN application. The NAC application mimics a RADIUS server, completes the handshake, and authenticates the user by having the Brocade SDN Controller send the appropriate crafted packets. At that point, the role-based resource application routine engages. Users who have devices that support 802.1X but are not currently in the database can then be queried by the NAC application injecting standard packets to "Web authenticate" the user. In other words, it mimics a DHCP server to give a temporary IP address, yet it only traps Web traffic from the user, mimics a Web server, and tunnels Web page information to set up a registration process for the user and add the user to the 802.1X database. Then, a 802.1X query is initiated and, if successful, lets the user access those resources that are allowed.

Some devices have active users on them but do not use the 802.1X protocol. These devices fail the checking of the source MAC against a list of allowed devices. Such devices do not respond to multiple 802.1X queries, thus the NAC application moves to a Web authentication. This authentication is similar to the one used for the 802.1X-capable but non-registered devices, but in this case is used each time the device needs to access the network. The NAC application gives the user the option of logging in or registering. If the user registers, the user then follows the same path as the 802.1X-capable users. The user supplies specific registration information, selects a username and password, and has those stored where the NAC application has access. Then the user is redirected to the login or register page and enters the username and password. That information is all tunneled through OpenFlow packets and—once authenticated—the OpenFlow table is configured with the appropriate table entries to allow the user onto the network with appropriate restrictions and resource allocations.

An alternate to having all of the Web page functionality tunneled through OpenFlow, is to temporarily redirect all traffic from the device to a registration Web server. That server then performs all of the Web-based registration functions, and sends its credentials and roles to the NAC application. For an 802.1X user, the port status changes to have the user authenticate using the mimicked 802.1X functionality described earlier. If the user is Web-based, then the NAC application receives the positive authentication, along with the role of the user, and sends the appropriate OpenFlow entries to the Brocade ICX 6610 to allow access to the network with appropriate restrictions and network resource allocation.

Communities of Interest

Communities of Interest (CoIs) consist of groups of individuals working together who need to share common resources. These might be teams of researchers located in disparate parts of a campus, cross-functional teams working on a project, contracted service vendors, students in a class, or any number of short-term collaborative groupings that need to work together and share dedicated network resources. Multitenancy applications also fall under this category: they are distinct groups of users that are securely segregated from other groups. Such groups share a common architecture, yet protections exist to ensure no group is starved for resources by the other groups.

Classically, this is accomplished by using various technologies such as VLANs connecting to Virtual Routing and Forwarding (VRF) instances with QoS access lists and rate limiters. Each time a new group is added to the network, it takes time to design how to overlay the group needs onto the existing network, then to access each affected device on the network and apply the appropriate configuration. An error at any step in the process leads to frustrated users and hours of troubleshooting, unnecessarily affecting the productivity of the group and wasting time and money.

An SDN solution mitigates human error and eliminates the vagaries of multivendor, multiproduct proprietary dependencies. Regardless of whether the group requirement is a static long-term deployment or a dynamic, short-term, "login, collaborate, and logout" situation, the Brocade SDN Controller maintains the state of the physical network topology, the devices connected to it, and the resources currently allocated. Thus it is able to seamlessly spin-up and spin-down the resources on demand.

As an example, consider a group of developers that are working on a six-month project to create the newest flagship product for their company. These users need to have guaranteed bandwidth from their development workstations. Other teams stagger their shifts, using the same servers (but different logins) and belonging to different teams.

This is typical for a cost-conscious company looking to maximize utilization of the engineering workstations. In such a case, each user could belong to a different CoI development team.

The process might work like this: As a user logs into the workstation, the same credentials are used to log into the network using the SDN NAC application described above. As users are identified, not only are their specific roles identified, but the CoI application determines if each user belongs to a specific active CoI. The role-based application sets the security constraints for each user and sends them to the Brocade SDN Controller. The CoI application determines the current set of users in the CoI who are logged in, and adds the new users to that pool, sending this information along with the desired bandwidth between the devices (as well as any aggregate bandwidth to reserve in aggregation/core devices in the network).

The Brocade SDN Controller takes the requests from the applications and determines which network devices need to have their OpenFlow tables modified, whether unique entries at the point of access or aggregated/subnetted entries at aggregation layers in the network mapping to multiple users in the CoI. The controller then pushes all of these table entries to the appropriate devices. All of this occurs in the time it takes for a user to log into the workstation and load the first file, and the process is invisible to the user.

The reverse process occurs as the user logs off the device. The CoI and role-based applications tell the Brocade SDN Controller to spin-down the resources for the user, and the Brocade SDN Controller determines which devices need to be changed. The controller then pushes the appropriate table entries on the aggregation layer and releases the resources at the access switch. There is no need to worry about ACL entries, Command-Line Interface (CLI) constraints, or how many users already received bandwidth on that link. The Brocade SDN Controller does everything for the user with a physical and logical mapping of resources and the OpenFlow API.

The Brocade Solution

Brocade is leading the way in moving SDN from simple test environments to real-world provisioning. The first Brocade products were the Brocade NetIron® CES, CER, and MLX platforms for core and WAN edge deployments. These included the visionary addition of hybrid per-flow mode that combined the resiliency of nearly two decades of hardware and software development for day-to-day traffic management with the agility of OpenFlow to dynamically fine-tune specific application and user flows. The initial release was an extended version of OpenFlow version 1.0. Current and future products are implementing OpenFlow version 1.3 with its increased functionality, high availability, and security.

Brocade NetIron products have allowed Brocade to optimize the device to Brocade SDN Controller performance and to develop the hybrid per-flow mode of operation into a robust enterprise-grade capability. These are now migrating as capabilities into the Brocade ICX family of campus products and the Brocade VDX family of data center Ethernet fabric products.

Beyond the Brocade MLX Series routers as a core and the WAN platform, the Brocade ICX 6610 will join the Brocade SDN ecosystem. This offers a solid OpenFlow-capable access switch with Power over Ethernet (PoE), but also a strong aggregation or small core switch and a robust data center front-end server top-of-rack switch. Brocade is committed to SDN and OpenFlow and will continue to expand the OpenFlow functionality into the Brocade ICX family, for a scaling enterprise campus topology. The Brocade VDX Series also offers flow to optimal fabric path mapping. The result is an end-to-end, dynamic network topology for SDN applications that until now could not have been orchestrated.

Brocade was a charter member of the OpenDaylight Project, founded in April 2013. The OpenDaylight controller, operated by the Linux Foundation, has quickly become the leading open-source SDN controller in the industry, benefiting from the combined support and expertise of most major networking providers, leading SDN startups, IT professional services firms, and individual developers and users.

The Brocade SDN Controller is a fully tested, extensible commercial distribution of the OpenDaylight controller, the leading open-source SDN controller. It Is the first commercial controller built directly from OpenDaylight code, without any proprietary extensions or platform dependencies.

The Brocade SDN Controller provides:

- A smooth on-ramp to SDN adoption with an easy-to-use GUI, installation tools, and expert developer support
- The broadest platform for use in multivendor environments—with the ability to control physical and virtual networking devices from all major vendors
- Single-source technical support for Brocade SDN Controller domains
- Education and professional services to help organizations develop and deliver their own business logic, use cases, and custom network services

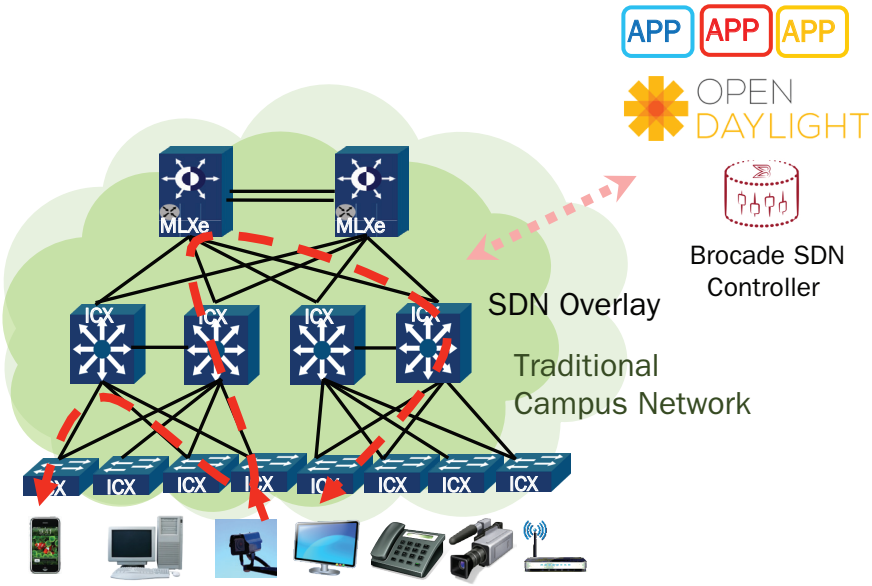

Figure 3. Brocade SDN campus solution.

Summary

Brocade recognized early on the power of SDN in transforming the network to operate in a 21st-century manner for 21st-century applications. The change is similar to the transformative nature of moving from DOS to Windows: moving from knowing far too much about how the hardware worked to utilizing software without an awareness of—or need to know about—the underlying hardware. In this transformation, Brocade understands the need to coexist and allow for a migration from the old to the new. The innovation of the hybrid per-flow mode is one of many innovations implemented by Brocade. This technology enables a controlled, gradual transition to a SDN ecosystem, with products that interoperate seamlessly with the existing network topology, while allowing for deploying SDN into the network on your terms and at your pace. The initial Brocade offering is a broad set of OpenFlow-enabled networking devices to address the campus onramp access layer through to the data center server. Brocade is also actively engaged in the standards bodies. Brocade champions the enterprise user viewpoint within the OpenDaylight Project, to ensure a robust, enterprise grade, cost-effective solution. Brocade also supports SDN application developer needs, to ensure a complete set of enterprise solutions. Brocade is committed to SDN at all layers—with networking devices, a robust open-source SDN controller, and enterprise SDN applications from the leading providers—for a complete solution for the enterprise business demands of today and tomorrow.

About Brocade

Brocade networking solutions help organizations achieve their critical business initiatives as they transition to a world where applications and information reside anywhere. Today, Brocade is extending its proven data center expertise across the entire network with open, virtual, and efficient solutions built for consolidation, virtualization, and cloud computing. Learn more at www.brocade.com.

Corporate Headquarters
San Jose, CA USA
T: +1-408-333-8000
info@brocade.com

European Headquarters
Geneva, Switzerland
T: +41-22-799-56-40
emea-info@brocade.com

Asia Pacific Headquarters
Singapore
T: +65-6538-4700
apac-info@brocade.com

© 2015 Brocade Communications Systems, Inc. All Rights Reserved. 09/15 GA-WP-1835=03

ADX, Brocade, Brocade Assurance, the B-wing symbol, DCX, Fabric OS, HyperEdge, ICX, MLX, MyBrocade, OpenScript, The Effortless Network, VCS, VDX, Vplane, and Vyatta are registered trademarks, and Fabric Vision and vADX are trademarks of Brocade Communications Systems, Inc., in the United States and/or in other countries. Other brands, products, or service names mentioned may be trademarks of others.

Notice: This document is for informational purposes only and does not set forth any warranty, expressed or implied, concerning any equipment, equipment features, or service offered or to be offered by Brocade. Brocade reserves the right to make changes to this document at any time, without notice, and assumes no responsibility for its use. This information document describes features that may not be currently available. Contact a Brocade sales office for information on feature and product availability. Export of technical data contained in this document may require an export license from the United States government.

BROCADE